PROPEL: AN APPROACH SUPPORTING USER GUIDANCE IN DEVELOPING PRECISE AND UNDERSTANDABLE PROPERTY SPECIFICATIONS

A Dissertation Presented
by
RACHEL L. COBLEIGH

Submitted to the Graduate School of the University of Massachusetts Amherst in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

September 2008

Computer Science
PROPEL: AN APPROACH SUPPORTING USER GUIDANCE IN DEVELOPING PRECISE AND UNDERSTANDABLE PROPERTY SPECIFICATIONS

A Dissertation Presented
by
RACHEL L. COBLEIGH

Approved as to style and content by:

______________________________
Lori A. Clarke, Co-chair

______________________________
George S. Avrunin, Co-chair

______________________________
Leon J. Osterweil, Member

______________________________
Elizabeth A. Henneman, Member

Andrew G. Barto, Department Chair
Computer Science
To Jamie, my beloved and friend. Thanks for every design review.
“Be anxious about nothing, but in all things, with prayer and supplication, with thanksgiving, make your request known to God, and the peace of God, which surpasses all understanding, will guard your hearts and minds through Christ Jesus.”

*Philippians 4:6-8*
ACKNOWLEDGMENTS

First and foremost, I want to thank Lori Clarke and George Avrunin for holding me to exacting standards for all these years: this work is as much their achievement as it is mine, and I will undoubtedly apply the lessons they taught me about critical thinking and perseverance to many other projects in the future. I think I have finally learned how to put my asbestos suit on!

I also want to thank Beth Henneman for her expertise in blood transfusion and patient identification, her attention to detail, her enthusiasm, her encouragement, and the way she infused a sense of fun into everything! She patiently taught me how to ask the right questions, and she provided an invaluable perspective that added a good dash of practice to my theories. Similarly, Lee Osterweil has continually challenged me to think outside the box and reconsider my assumptions. I want to thank him for his encouragement and his long-term vision throughout this entire process.

In addition to Beth, I am also indebted to a number of other medical domain experts for their contributions to our case study evaluation. At Baystate Medical Center, Dr. Phil Henneman has lent his expertise to our efforts to capture Emergency Department properties and processes. At the D’Amour Center for Cancer Care at Baystate, Dr. Lucy Cassells, Dave Brown, Gina Parisi, Ann Garbecki, Gary Bessette, and Dr. Wilson Mertens have lent their expertise to our efforts to capture chemotherapy properties and processes. I am also grateful for the time and expertise that Marie Henebury, MSGt Nancy Walker, TSgt Renea Ivey, James Berger, and LCDR Felix Alfonso contributed towards the Defense Blood Standard System properties.

My daily life as a grad student was greatly influenced by my colleagues in LASER. Sandy Wise kept everything humming and seemed to know a little something about nearly everything; I appreciate all the practical things he taught me. Heather Conboy has been a great source of clear thinking, good questions, implementation advice, and friendship. Barb Lerner offered career advice every time I asked. The implementation of PROPEL was a group effort: I am grateful for the significant contributions of Matt O’Connell, Vitaliy Lvin, Dave Miller, Ricky Chang, Valerie Gartland, and Huong Phan. I could not have done it alone! For their friendship and the opportunity to discuss ideas of all kinds, I want to thank all my fellow grad students, especially Matt Billmers, Joel Sieh, M.S. Raunak, Shangzhu Wang, Bin Chen, Jianbin Tan, Amr Elssamadisy, Bobby Simidchieva, Stefan Christov, and Aaron Cass.

Many people have inspired me to start and finish this degree. First, I want to thank my parents, Doug and Sara Smith, for their encouragement and the high value they place on education and personal sacrifice.
They have been a great example of God’s love to me. I also want to thank my uncle, Dr. Greg Smith, for encouraging me, correcting me, and holding me up to a high standard. In addition, I’m grateful for my sisters, Jessica Michalski, Ruthie Smith, and Priscilla Smith, who have prayed and teased and encouraged me not to give up. Similarly, thanks must go to my roommates, Michelle Reyes and Laylaa Ali, for putting up with my strange hours and odd obsessions. Many thanks to my church families at APA, College Church, and Elmwood Chapel, for every time somebody asked me how my degree was coming along: it was a constant source of motivation to keep up my momentum. I also want to thank Bob and Marge Day, who provided me with a wonderful place to live for four years, and Bill and Judy Glucksman, who let me stay with them when I didn’t have a home near UMass. For inspiring me to pursue a Ph.D. in the first place, I want to thank Tony DiRuzza, my 10th-grade Biology teacher, who encouraged me to apply for entrance to Mass Academy, and Pauline Lamarche, who taught me how to program in C when I got there.

I cannot sufficiently express my gratitude to my husband, Jamie Cobleigh, for the countless ways in which he has taught me, sacrificed for me, encouraged me, challenged me, inspired me, cracked the whip to keep me focused on finishing, and celebrated every victory with me. There were a few times when I felt like giving up and he asked the right questions to keep me going. I love him dearly.

Finally, and most importantly, I’m utterly grateful to God for giving me life, a mind, a love for learning, and the opportunity to pursue this degree. I’ve raged and cried and leapt for joy over this effort, and none of it would be possible without You. You give this work its value and meaning. Thanks for concealing these matters and giving me the honor of searching them out. I look forward to whatever comes next!
ABSTRACT

PROPEL: AN APPROACH SUPPORTING USER GUIDANCE IN DEVELOPING PRECISE AND UNDERSTANDABLE PROPERTY SPECIFICATIONS

SEPTEMBER 2008

RACHEL L. COBLEIGH
B.Sc., UNIVERSITY OF MASSACHUSETTS AMHERST
M.Sc., UNIVERSITY OF MASSACHUSETTS AMHERST
Ph.D., UNIVERSITY OF MASSACHUSETTS AMHERST

Directed by: Professor Lori A. Clarke and Professor George S. Avrunin

Property specifications are often used in requirements engineering to concisely describe a single aspect of system behavior. Although each property has a narrow focus, it can still be difficult to specify a property correctly. There are often subtle, but important, details in desired system behavior that can easily be overlooked, and there is little guidance available for how to avoid such mistakes. In addition to capturing these details correctly, property specifications should be (a) precise enough to support automated analyses that can be used to check that actual system behavior is consistent with the specifications, and (b) understandable enough to be readily comprehended by all system developers. Property specifications can be written in a mathematical formalism, which provides precision, but such formalisms are often difficult to understand. Thus, in practice, property specifications tend to be written in natural language. Property specifications written with such informality are often ambiguous, however, and usually cannot be used in many types of automated analyses.

To address these challenges, our approach supports elicitation and specification of properties by providing templates that build on commonly-occurring property patterns. These templates offer guidance by explicitly indicating the variations that must be considered, thereby ensuring that important subtle details are not overlooked. To support the use of this approach, we developed PROPEL, for “PROPerty ELucidator.” PROPEL provides three alternative views that users can work with to specify properties: graphical finite-state automata, which offer precision; “disciplined” natural language, which offers understandability; and question trees, which offer guidance for selecting a template.
To evaluate this approach, we used PROPEL to specify properties in five case studies in the medical domain. The case studies showed that our approach was effective at specifying the vast majority of the properties we encountered. We also undertook a small empirical study that showed that the disciplined natural language view of the properties was usually understood. These results indicate that our approach to property elicitation and specification is a promising one.
# TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>ACKNOWLEDGMENTS</th>
<th>vi</th>
</tr>
</thead>
<tbody>
<tr>
<td>ABSTRACT</td>
<td>viii</td>
</tr>
<tr>
<td>LIST OF TABLES</td>
<td>xi</td>
</tr>
<tr>
<td>LIST OF FIGURES</td>
<td>xii</td>
</tr>
</tbody>
</table>

## CHAPTER

1. INTRODUCTION .................................................................................. 1

2. RELATED WORK .................................................................................. 5

2.1 Overview of Requirements Engineering ....................................... 5

2.2 Requirements Elicitation ............................................................... 7

2.2.1 User Research ........................................................................... 8

2.2.2 Natural-Language Approaches .................................................. 10

2.2.2.1 Natural-Language Processing ................................................ 10

2.2.2.2 Question-Based Guidance .................................................... 13

2.2.3 Patterns and Templates ............................................................... 13

2.2.4 Specification Mining ................................................................. 17

2.3 Requirements Modeling ................................................................. 19

2.3.1 Natural-Language Approaches .................................................. 19

2.3.1.1 Boilerplates ........................................................................... 19

2.3.1.2 Recommendations and Heuristics ......................................... 20

2.3.2 Functional Approaches ............................................................... 21

2.3.3 Goal-Oriented Approaches ......................................................... 22

2.3.4 Scenario-Based Approaches ....................................................... 23

2.4 Formal Property Specification ....................................................... 24

2.4.1 History-Based Specifications ..................................................... 26

2.4.2 State-Based Specifications ......................................................... 27

2.4.3 Transition-Based Specifications ................................................ 28

2.4.4 Functional Specifications ......................................................... 29

2.4.5 Operational Specifications ....................................................... 30
2.5 Requirements Organization ............................................................. 30
  2.5.1 Techniques for Finding Structure ............................................. 31
  2.5.2 Techniques for Structuring Requirements Documents ...................... 32
  2.5.3 Strategies for Expressing Individual Requirements’ Relationships .......... 33
2.6 Requirements Analysis and Validation ............................................. 34
2.7 Requirements Management Processes .............................................. 36
2.8 Summary ....................................................................................... 38

3. A PROPERTY ELUCIDATION APPROACH ........................................ 40
  3.1 Specifying a Behavior ................................................................. 43
    3.1.1 Behavior Templates .............................................................. 44
    3.1.2 Behavior Question Tree ......................................................... 45
      3.1.2.1 QT Notation .................................................................... 45
      3.1.2.2 An Example of Instantiating a Property’s Behavior in the BQT .......... 47
    3.1.3 Behavior Finite-State Automata Templates .................................... 54
      3.1.3.1 FSA Template Notation ...................................................... 54
      3.1.3.2 An Example of Instantiating a Property’s Behavior FSA Template ...... 55
    3.1.4 Behavior Disciplined Natural Language Templates ............................ 58
      3.1.4.1 DNL Template Notation ...................................................... 59
      3.1.4.2 An Example of Instantiating a Property’s Behavior DNL Template ...... 60
    3.1.5 Using All Three Views to Instantiate a Property’s Behavior ................. 63
    3.1.6 Behavior Summary ............................................................... 64
  3.2 Specifying a Scope ........................................................................ 65
    3.2.1 Scope Templates ................................................................. 65
    3.2.2 Scope Question Tree .............................................................. 68
      3.2.2.1 An Example of Instantiating a Property’s Scope in the SQT ............... 69
    3.2.3 Scope Finite-State Automata Templates ........................................ 74
      3.2.3.1 An Example of Instantiating a Property’s Scope FSA Template ............ 75
    3.2.4 Scope Disciplined Natural Language Templates ................................. 78
      3.2.4.1 An Example of Instantiating a Property’s Scope DNL Template .......... 79
    3.2.5 Scope Timeline ...................................................................... 82
      3.2.5.1 An Example of Instantiating a Property’s Scope in the Timeline .......... 83
    3.2.6 Using All Four Views to Instantiate a Property’s Scope ....................... 85
    3.2.7 Scope Summary ..................................................................... 86
  3.3 Summary ....................................................................................... 87
4. THE PROPEL TOOL ................................................................. 91
   4.1 Design Goals ............................................................... 91
   4.2 Architecture ............................................................. 92
   4.3 Alphabet Views .......................................................... 95
   4.4 Support for Multiple Properties ...................................... 97
      4.4.1 Project Tree View .................................................. 97
      4.4.2 Summary Views .................................................... 99
   4.5 Summary .................................................................. 101

5. CASE STUDY EVALUATION ..................................................... 102
   5.1 Overview ................................................................ 102
   5.2 Methodology ............................................................. 103
   5.3 Quantitative Results ...................................................... 107
      5.3.1 Overview ............................................................. 107
      5.3.2 Limitations on PROPEL’s expressibility ...................... 110
         5.3.2.1 Concepts that were not expressible ....................... 110
         5.3.2.2 Issues that can probably be handled better .......... 111
         5.3.2.3 Parameterization .............................................. 114
      5.3.3 Informal-to-formal mappings .................................... 115
         5.3.3.1 Multiple resources ........................................... 116
         5.3.3.2 Single-variable state information ......................... 117
         5.3.3.3 Ordered AND decompositions ............................ 117
         5.3.3.4 Restrictions on scope delimiters ......................... 119
      5.3.4 Observations about the distribution of properties .......... 120
         5.3.4.1 Differences in scope frequencies ......................... 120
         5.3.4.2 Differences in behavior frequencies .................... 122
         5.3.4.3 Scope frequencies are related to behavior frequencies 125
      5.3.5 Observations about option-setting frequencies ............... 125
      5.3.6 Summary of the quantitative results ......................... 128
   5.4 Qualitative Observations ............................................... 129
      5.4.1 Initial learning stage for both computer scientists and domain experts 129
      5.4.2 Issues involved in identifying an abstract informal specification of a property 130
         5.4.2.1 Three different perspectives influenced abstract goals and levels of abstraction 131
         5.4.2.2 Domain boundaries influenced a property specification’s level of abstraction 136
      5.4.3 Issues involved in stating an informal property specification clearly ........ 138
         5.4.3.1 Addressing terminology problems ....................... 138
         5.4.3.2 Techniques for improving the clarity of informal property specifications 142
         5.4.3.3 Clearly stating a set of properties requires still more work ........... 145
5.4.4 Issues involved in formalizing a property .............................................. 145
  5.4.4.1 Challenges and benefits in formalization ........................................... 146
  5.4.4.2 Lessons learned about the use of PROPEL ........................................ 148
  5.4.4.3 A formal specification can have more or less detail, depending upon its intended use ................................................................. 153

5.4.5 Issues involved in organizing a set of properties ...................................... 155
  5.4.5.1 Property groupings ................................................................. 155
  5.4.5.2 Organizational structure can change .............................................. 161

5.4.6 Summary of qualitative observations ....................................................... 162

5.5 Threats to Validity ....................................................................................... 163
5.6 Summary .................................................................................................... 165

6. DNL TRANSLATION STUDY EVALUATION ................................................... 167
  6.1 Methodology .............................................................................................. 168
  6.2 Observations .............................................................................................. 170
  6.3 Summary .................................................................................................... 172

7. CONCLUSIONS ............................................................................................. 173
  7.1 Observations .............................................................................................. 174
  7.2 Future Work ............................................................................................... 175
    7.2.1 User-Interface Issues ........................................................................... 176
      7.2.1.1 Property-decomposition guidance ................................................. 176
      7.2.1.2 Option-resolution guidance ......................................................... 177
      7.2.1.3 Scope-focused property views ..................................................... 179
      7.2.1.4 Alphabet Views ........................................................................... 180
      7.2.1.5 Project Tree View ....................................................................... 181
      7.2.1.6 Summary Views ......................................................................... 181
    7.2.2 Expressibility Issues ............................................................................ 182
      7.2.2.1 Exploring new behavior templates ............................................. 182
      7.2.2.2 Loosening the restrictions on the scope templates ...................... 185
      7.2.2.3 Exploring event patterns and compositions .................................. 186
      7.2.2.4 Exploring other property-specification paradigms and languages ... 187
    7.2.3 Evaluation Ideas .................................................................................. 188
    7.2.4 Summary .............................................................................................. 189

APPENDICES

A. PROPERTY VIEW DETAILS ...................................................................... 190
  A.1 Question Trees ....................................................................................... 191
    A.1.1 Behavior Question Tree ................................................................. 191
# LIST OF TABLES

<table>
<thead>
<tr>
<th>Table</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1 Summary of Selected Formal Specification Languages</td>
<td>25</td>
</tr>
<tr>
<td>3.1 The Four Behavior Templates</td>
<td>44</td>
</tr>
<tr>
<td>3.2 Summary of the Behavior Templates’ Options</td>
<td>44</td>
</tr>
<tr>
<td>3.3 Comparison of Property and Scope Views in Terms of Design Goals</td>
<td>87</td>
</tr>
<tr>
<td>5.1 Property Expressibility Summary</td>
<td>108</td>
</tr>
<tr>
<td>5.2 Case Studies’ Property Elucidation Summary</td>
<td>109</td>
</tr>
<tr>
<td>5.3 Frequencies of the Problematic Issues</td>
<td>111</td>
</tr>
<tr>
<td>5.4 Types of Informal-to-Formal Property Specification Mappings</td>
<td>116</td>
</tr>
<tr>
<td>5.5 Formalization Status Marking Convention</td>
<td>142</td>
</tr>
<tr>
<td>6.1 Property Complexity Categories</td>
<td>168</td>
</tr>
<tr>
<td>6.2 DNL Study Experimental Results</td>
<td>171</td>
</tr>
<tr>
<td>7.1 Expanded Comparison of Property and Scope Views in Terms of Design Goals</td>
<td>179</td>
</tr>
<tr>
<td>A.1 Options that Affect State 3’s Transitions in the Response and Precedence FSA Templates</td>
<td>201</td>
</tr>
<tr>
<td>A.2 Response and Precedence FSA Template Interaction Constraints</td>
<td>203</td>
</tr>
<tr>
<td>D.1 DNL Translation Study Raw Data Summary</td>
<td>550</td>
</tr>
<tr>
<td>D.2 DNL Translation Study Raw Data</td>
<td>550</td>
</tr>
</tbody>
</table>
# LIST OF FIGURES

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.1</td>
<td>Behavior Template Organization in the BQT</td>
<td>46</td>
</tr>
<tr>
<td>3.2</td>
<td>Choosing a Behavior Template in the Behavior Question Tree</td>
<td>48</td>
</tr>
<tr>
<td>3.3</td>
<td>Resolving the Nullity Option in the Behavior Question Tree</td>
<td>49</td>
</tr>
<tr>
<td>3.4</td>
<td>Resolving the Pre-arity Option in the Behavior Question Tree</td>
<td>50</td>
</tr>
<tr>
<td>3.5</td>
<td>Resolving Three Options Together in the Behavior Question Tree</td>
<td>51</td>
</tr>
<tr>
<td>3.6</td>
<td>Example Fully-Resolved Behavior in the BQT</td>
<td>53</td>
</tr>
<tr>
<td>3.7</td>
<td>Example Partially-Resolved FSA Template</td>
<td>55</td>
</tr>
<tr>
<td>3.8</td>
<td>Resolving the Nullity Option in the FSA Template</td>
<td>56</td>
</tr>
<tr>
<td>3.9</td>
<td>Resolving Two Options Together in the FSA Template</td>
<td>57</td>
</tr>
<tr>
<td>3.10</td>
<td>Resolving Three Options Together in the FSA Template</td>
<td>57</td>
</tr>
<tr>
<td>3.11</td>
<td>Example Behavior FSA</td>
<td>58</td>
</tr>
<tr>
<td>3.12</td>
<td>Example Partially-Resolved DNL Template</td>
<td>60</td>
</tr>
<tr>
<td>3.13</td>
<td>Resolving the Nullity Option in the DNL Template</td>
<td>61</td>
</tr>
<tr>
<td>3.14</td>
<td>Resolving Two Options Together in the DNL Template</td>
<td>61</td>
</tr>
<tr>
<td>3.15</td>
<td>Resolving Three Options Together in the DNL Template</td>
<td>62</td>
</tr>
<tr>
<td>3.16</td>
<td>Example Behavior DNL Specification</td>
<td>63</td>
</tr>
<tr>
<td>3.17</td>
<td>Examples of Scopes Based on the Four Scope Templates</td>
<td>67</td>
</tr>
<tr>
<td>3.18</td>
<td>Two Ways To Interpret a Missing End Delimiter</td>
<td>68</td>
</tr>
<tr>
<td>3.19</td>
<td>Scope Template Organization in the SQT</td>
<td>69</td>
</tr>
<tr>
<td>3.20</td>
<td>Choosing a Scope Template in the Scope Question Tree</td>
<td>70</td>
</tr>
<tr>
<td>3.21</td>
<td>Resolving an Option in the Scope Question Tree</td>
<td>71</td>
</tr>
<tr>
<td>3.22</td>
<td>Resolving Another Option in the Scope Question Tree</td>
<td>72</td>
</tr>
</tbody>
</table>
CHAPTER 1
INTRODUCTION

In a world that is increasingly dependent on complex systems whose failures could result in human casualties or significant economic loss, it is vital that the people who build and maintain these systems clearly understand how these systems are expected to behave. The complexity of such a system means that many people are usually involved in its development and no one person can comprehend it in its entirety. Requirements are thus an essential tool for prescribing how various aspects of the system should behave. Requirements can be used as the basis for the design, implementation, validation, and maintenance of the system. Since requirements are usually the first statement of desired system behavior and of assumptions about the system’s context of use, they impact all the other steps in the system development life cycle. If errors in the requirements are not found until late in the life cycle, those errors can be difficult and expensive to correct, because their impact is so far-reaching [35, 179]. Careful requirements engineering is thus indispensable to understanding, building, and maintaining a complex system.

A key contribution of requirements engineering is ensuring that the needs and desires of system stakeholders are clearly communicated to those who are building and maintaining the system. System stakeholders (or stakeholders) include both system users, who are the end users or who have an end-user view of the system, and system developers, such as requirements engineers, system designers, implementers, quality engineers, and documentation writers, who need to have a deep understanding of the system as they build and maintain it. Requirements engineers are responsible for developing and maintaining the requirements, and doing so can be a complex task because system stakeholders can change their perspectives over time. Requirements engineers thus have a number of different responsibilities that span the entire system development life cycle. Their responsibilities include identifying who the key system users are, capturing assumptions about the system’s context of use, eliciting the requirements from all the system stakeholders, specifying the requirements in a form that enables effective communication among the system developers, resolving the conflicts between the stakeholders’ differing expectations and perspectives, validating the requirements with all of the stakeholders, and managing changes in the stakeholders’ expectations and thus changes in the requirements.

A requirements specification encodes requirements, which can include both behavioral constraints and non-behavioral constraints (e.g., justifications, priorities, or resource limitations), in some notation. A re-
requirement can prescribe fairly vague system-wide qualities, such as “the system must be user-friendly and fast,” or an extremely specific constraint on one particular aspect of system behavior, such as “when a patient’s temperature rises above a safe level, the patient monitoring system must send a ‘suspected transfusion reaction’ message to the nurse.” We refer to a requirement that focuses on a particular aspect of system behavior as a property of the system. In this work, we focus on supporting elicitation and specification activities for property specifications that are amenable to automated analyses, such as automated formal verification, because these analyses are powerful tools that can help system developers clearly understand system behavior. Although communicating with system users is an essential part of developing and validating properties, system users are often not the intended audience for the property specifications. For this reason, the property specifications that we focus on in this work are aimed primarily at system developers.

Ideally, these property specifications should be both precise enough to support automated analyses and also understandable enough to be readily comprehended by all the system developers. Property specifications that are hard to understand or that are imprecise can lead to an incorrect design, a faulty implementation, or erroneous validation. There has historically been a trade-off between precision and understandability in property specifications, and most property-specification approaches focus on supporting just one of these qualities. There has been a great deal of work done on developing mathematically-precise property-specification formalisms, such as temporal logic or finite-state automata. The strength of these formalisms is that they provide precise property specifications that can be used as the basis for automated analyses, such as automated formal verification [57], which check that the behavior of a system is consistent with its specified properties, or automated test-case generation [257]. The weakness of these formalisms, however, is that they are often difficult for system developers to understand [142, 172]. Because of this shortcoming, requirements engineers tend to avoid the more mathematical property-specification formalisms, and they instead write property specifications in natural language, perhaps sprinkled with some tabular or graphical notations to try to add precision. Although these kinds of specifications seem to be more understandable to system developers, properties written with such informality often contain ambiguous or inconsistent descriptions of the system. Thus, they are of limited value when doing system development. To address this trade-off in property specification, we have developed an approach that produces properties that are represented in a way that supports both understandability and precision.

In addition to producing property specifications that have both of these qualities, an effective approach must also be able to help elicit property specifications from system stakeholders and to concisely express the system behavior that those stakeholders desire. The issue of how to precisely and correctly express the desired system behavior is a serious problem, because even though property specifications are written with a focus on one particular aspect of a system’s behavior, it is still surprisingly difficult to write them correctly.
Property specifications are often almost correct, but fail to capture some subtle, but important, details of the desired system behavior that are easily overlooked. Often these details are not revealed until validation of the system begins, late in the system development life cycle. In such situations, system developers spend a considerable amount of time trying to ensure that the system adheres to a property specification, only to later determine that the property has been specified incorrectly. To address this problem, the property-specification approach that we have developed is also intended to provide requirements engineers with elicitation guidance.

There are few approaches that try to offer both understandability and precision, and fewer still that also try to incorporate some form of guidance for requirements engineers. In this work, we propose a property-specification approach [60, 239, 240] that is designed to address these issues. Our approach provides a set of templates for commonly-occurring types of properties. These templates explicitly indicate the variations that must be considered, thereby ensuring that important subtle details are not overlooked. Our approach represents these templates using multiple property views, which were chosen to support precision, understandability, and elicitation guidance. We have developed PROPEL, for “PROPerty ELucidator,” which is a tool that supports the use of this approach. PROPEL provides three different property views that PROPEL’s users, or specifiers, can work with to create their properties:

- a graphical, extended Finite-State Automata (FSA) template, which offers precision;
- a Disciplined Natural Language (DNL) description, which offers understandability; and
- a hierarchical Question Tree (QT), which offers guidance for selecting a property template.

Used together, these three property views are meant to mitigate the tension between precision and understandability in property specifications and to provide specifiers with guidance for eliciting subtle property details. Specifiers have flexibility in choosing whether to work with one view at a time or with any combination of views simultaneously. These views are automatically kept synchronized with each other: as specifiers make a change in one of the views, the other views correspondingly change. It is our expectation that this approach will help specifiers to better understand, and thus correctly express, their desired properties. The property specifications that are produced using our approach do not require extensive expertise in specification formalisms to be understood, and they can be used as the basis for verifying system behaviors, testing the acceptance of event sequences, validating the consistency of a set of properties, and other types of analyses.

We performed an evaluation of our approach to determine how effective it is at helping to create property specifications that are both precise and understandable. To conduct this evaluation, we used PROPEL to specify properties in five case studies in the medical domain: a chemotherapy process, a blood transfusion process, an emergency department process, a process for verifying patient identity, and a process for managing a national blood repository. For each of these case studies, we elicited properties by interviewing domain
experts, who were the system users, and occasionally the specifiers, for these processes, and by examining source documents that the domain experts gave us. Although there are reasons to be cautious about coming to strong conclusions about our approach, there are good indications that this approach does provide effective support for helping specifiers, including those who do not have expertise in requirements engineering, to develop precise and understandable property specifications that capture desired system behavior. We elicited a total of 145 properties and were able to use PROPEL to specify 95% of them, a surprisingly positive result. Although most of the domain experts did not work directly with PROPEL, we observed that the subtle details that PROPEL required the domain experts to resolve helped to clarify many issues that they had not examined closely before their participation in these case studies. Overall, the domain experts responded very positively to the property elucidation activities and the use of our approach. In addition to these positive findings, these case studies also illustrated several limitations of our approach, and some possible ways to address these shortcomings are discussed in Chapter 7.

Following this introduction, the next chapter of this thesis presents a discussion of related work and then Chapter 3 provides a detailed description of our approach. It gives background information on property patterns and then describes the property templates and the three property views in which they are represented, along with an illustrative example from one of the case studies in our evaluation. Chapter 4 describes PROPEL’s architecture and two forms of support that the tool provides for managing multiple property specifications. Chapter 5 discusses the five case studies used in the evaluation of our approach, and it presents the results and some qualitative observations on this experience. Similarly, Chapter 6 discusses a small study that was conducted to evaluate the understandability and precision of the DNL property view. Finally, Chapter 7 presents our conclusions and discusses possibilities for future work.
CHAPTER 2
RELATED WORK

2.1 Overview of Requirements Engineering

Requirements engineering (RE) is a branch of systems engineering, where a system is a process that might involve hardware, software, or people. RE is primarily concerned with three areas of system development:

- Discovering how a system should behave with respect to its context of use (i.e., its environment) and the problems that it is intended to address.

- Representing those discoveries in a form that can be used for communication among system developers, validation of possible designs or implementations of the system, and guidance for how to build it.

- Managing the impact of stakeholders’ changing expectations of system behavior.

Some of the earliest work in RE drew attention to the fact that inadequate, inconsistent, ambiguous, or incomplete requirements can lead to significant cost and quality problems [26, 35, 179]. Despite several decades of research in RE, understanding how to effectively define a system’s requirements is still recognized as an on-going challenge, and progress still needs to be made in increasing awareness of RE as a complex and necessary task that cannot be taken for granted. In fact, as new technologies enable ever more complex systems to be developed, it is increasingly apparent that effective requirements development and maintenance is a vital part of the system development and maintenance process [30].

Zave [270] describes RE in terms of the relationship between the goals that stakeholders have for the system and the precise specifications of those goals, and also in terms of how those goals and specifications evolve over time. This description of RE is a good high-level summary because it concisely highlights three major aspects of requirements development and maintenance. First, this description highlights the motivation for carefully defining system requirements and keeping the development of the system tied to real-world goals. Second, this description highlights precision as an essential basis for how to leverage the requirements in meaningful ways. Third, Zave’s description of RE also points out an issue beyond just the discovery and representation of system requirements: effective RE also must be able to manage how requirements and systems evolve over time. This evolution is not just concerned with the change between versions of a single system, but also with how various aspects of the system definition might be reused in other systems.
A system and its requirements evolve over time as its stakeholders’ needs evolve, and it is important to remember that RE is essentially a human-centered process that is focused on resolving uncertainty and handling conflicting expectations. Cheng and Atlee [48] argue that RE differs from the rest of systems engineering in the sense that it is primarily RE that resides in the problem-definition space. Although RE approaches can often help with validating potential solutions, many RE approaches are first used for defining the problem that a system should address. The problem-definition space presents three challenges that are distinct from those in the solution-definition space:

• The problem-definition space is often ill-defined and contradictory at first. Many RE approaches are thus focused on defining terminology, building ontologies, and negotiating among system stakeholders’ varying perspectives.

• The problem-definition space is less constrained than the solution-definition space. Requirements engineers are thus often involved with a wide variety of concerns beyond just the intended system’s interface with its environment. It is often necessary to gain a deeper understanding of the environment (e.g., hardware, software, the physical world, and human behavior, both individual and organizational) before requirements specification can even begin.

• Because requirements must often bridge between non-technical stakeholders, who do not have expertise in formal languages, and system designers, implementers, and testers, who need precise specifications of the intended system, RE artifacts (e.g., property specifications, requirements models, requirements documents) must achieve both understandability and precision. RE artifacts thus must bridge between the informal and the formal, with all of the attendant pitfalls.

To address these three challenges, RE involves a complex set of activities that span the entire system development life cycle. These activities include the following:

• identifying who the key system users are;

• capturing stakeholders’ assumptions about the system’s environment;

• eliciting the requirements from all the system stakeholders;

• specifying the requirements in a form that enables effective communication among the system developers;

• resolving the conflicts between the system stakeholders’ differing expectations and perspectives;

• validating the requirements with all the system stakeholders; and
• managing changes in the system stakeholders’ expectations and thus changes in the requirements.

The following discussion explores each of these RE activities and some of the proposed approaches to handling them, and in doing so, aims to present an organizational framework for the research that has been done in RE. This framework builds on the work described in several previous surveys [1, 48, 210, 271] and puts particular emphasis on work that is more closely related to our research. The RE activities are discussed here in a particular order, but in practice RE processes are not necessarily linear in nature nor are the activities clearly delineated from one another. Depending on the system in view, some activities might not occur at all, or the activities might be interleaved or iterated through in a wide variety of different orderings. It can be difficult to draw a clear line between the various RE activities because each activity contributes to other activities. For example, many approaches discussed in this chapter can be thought of as contributing to multiple activities. By and large, however, these approaches are referenced where they make their dominant contribution. In addition, since our case studies (see Chapter 5) focused on the medical domain, we also reference where other RE approaches have been applied in healthcare. These references are placed in the context of the primary RE activities involved.

Section 2.2 describes approaches that are primarily aimed at eliciting information about who the key system users are, what the system stakeholders’ assumptions are about the system’s environment, and what the system stakeholders want the system behavior to be. Section 2.3 then describes approaches that are primarily aimed at modeling this information in an informal or semi-formal notation, and Section 2.4 describes approaches that are aimed at specifying this information in a formal notation. Based on such RE artifacts being produced, Section 2.5 describes approaches that aim to help with organizing collections of them. Section 2.6 then describes approaches that analyze those RE artifacts to improve the quality of requirements specifications and knowledge about requirements’ relationships, and to aid stakeholders with validating the requirements. Following that, Section 2.7 briefly touches on approaches that take a wider view of requirements development as it changes over time and across globally-distributed groups of system developers. Finally, Section 2.8 finishes with a discussion of how our property-elucidation approach fits into this organizational framework.

2.2 Requirements Elicitation

The elicitation of system requirements is often considered the first step in the RE process and in the overall system development life cycle. Requirements elicitation activities are concerned with gaining an understanding of the desired system goals and the assumptions about its environment from the system stakeholders. The research on requirements elicitation varies widely in the formality of approaches suggested, often based on when in the requirements elicitation process the approaches are designed to be used. Many
informal elicitation approaches offer suggestions about the types of questions to ask to begin building a high-level ontology of the system’s domain and to begin discovering where the boundaries between the system and its environment are. The discovery of where these boundaries lie, at least at a high level, is centrally important to determining the direction of the rest of the RE process. Thus, it is important that the requirements engineers identify the key system users and work to bring all the system stakeholders to an agreement about where the system’s boundaries are. Using this initial domain ontology and these boundaries, requirements engineers can do informal or semi-formal modeling, formal specification, or analysis, often in an iterative fashion, continually refining this knowledge. As the knowledge becomes more refined, formal elicitation approaches may be appropriate to use alongside the informal ones. For example, the effort to specify a requirement more formally can in itself prompt further elicitation, since the level of detail required to specify a requirement formally might necessitate more information than was gathered informally before that point.

There has been a great deal of work done in how to bridge the divide between informality and formality in requirements elicitation, ranging from approaches that try to handle the formalization process automatically, such as natural-language processing, to approaches that aim to provide some guidance through the use of formal patterns or templates. In addition, there are a number of specification mining approaches that try to avoid informality in requirements elicitation altogether and instead aim to extract requirements from other system design or implementation artifacts that are already specified in a formal notation.

The following section begins by describing several informal elicitation approaches. We then describe approaches that use natural language to aid with the elicitation process, and these approaches range from formal natural-language processing techniques to informal question-based guidance for improving the quality of requirements written in natural language. Finally, we explore two types of approaches that aim to bridge the divide between informality and formality in requirements elicitation: specification mining and the use of formal patterns or templates.

### 2.2.1 User Research

There are a number of largely informal but structured ways of obtaining information from system users, and these approaches are often referred to as “user research” or “field methods.” A variety of approaches have been developed to work within different types of constraints, depending upon how many system users are available, the nature of their availability, the amount of time that can be used for the elicitation activities, the depth of information required, and the degree to which prototypes of the desired system are available.

In situations where a great depth of information is needed about the environment that the system is intended to work within, contextual inquiry, also known as ethnographic study or field observation [140,261], has been found effective. It is extremely time-consuming, however, and the number of system users that can
participate is severely limited by this constraint. We briefly employed a light-weight version of this approach in our Chemotherapy case study (see Chapter 5) when we were given an opportunity to tour a pharmacy and were able to observe and ask questions in that environment. An alternative to contextual inquiry that can yield a similar level of depth is log books, or introspection [207]. In this approach, the system users themselves observe and record the details in their environment, for a length of time that is similar to that used in contextual inquiry. Requirements engineers then debrief the system users after the fact, as a replacement for actually being present in the environment themselves, and rely on the log book for more detailed information. This approach has been found effective in situations where requirements engineers are not permitted to be present, such as in certain restricted medical environments.

For situations where less time is available or there is a need for eliciting information from a larger number of system users, surveys/questionnaires [222, 233], focus groups [115], or interviews [262] are alternative elicitation methods. Of these, interviews seem to be the most common method employed [73]. They were the primary means of requirements elicitation activities in our work [56, 135, 136], and in the work of many others who have applied RE approaches in the medical domain [98, 99, 105, 153]. One structured interview approach that has been used in healthcare and in industry is the Joint Application Development (JAD) approach [262]. JAD is focused on a small interview group where each participant has an assigned role, and these roles involve requirements engineers and system stakeholders in an informally-structured discussion of the desired system’s goals and boundaries. One benefit of this structured interview approach is its scalability: Garde and Knaup [105] successfully used JAD in an 8-year, 70-institution, nationwide RE effort to build a unified software system for pediatric oncology in Germany. Another structured interview approach is Antón’s Goal-Based Requirements Analysis Method (GBRAM) [12], which provides a framework for how to use interviews to elicit and refine goal, agent, and constraint information about a desired system.

Some elicitation approaches also try to incorporate other parts of the software development or RE process in the requirements elicitation process. For example, some approaches suggest doing elicitation in parallel with modeling, analysis, or validation, and they use scenarios and use cases that describe how system users expect to interact with the system, inspections [218] or other analyses of existing organizational artifacts, or requirements animation [1, 111]. One example of this hybrid approach is evolutionary prototyping [49, 74, 79, 158, 241], which is the practice of creating a prototype of the desired system and then continually updating this prototype as necessary throughout the system development life cycle as the system stakeholders change their expectations and thus the system requirements. The initial prototypes are likely to be thrown away, but later versions might eventually be used to build the final product after the domain and system boundaries are well understood. Related to this integrated style of requirements elicitation is the Rapid Application Development (RAD) approach [196], which adds evolutionary prototyping to the JAD interview structure. In
the RAD approach, requirements engineers and software developers compress the requirements efforts and the system design, build, and test phases into a series of short, iterative development cycles.

2.2.2 Natural-Language Approaches

Natural languages are widely used in practice to informally describe system requirements. Mich et al. [200] recently conducted a study that suggests that up to 95% of the requirements documents used in industry are written in normal (79%) or structured (16%) natural language. Since natural language (NL) is such a common representation for specifying requirements, there are many approaches to requirements elicitation that rely on the use of NL. These approaches range from automated natural-language processing (NLP) techniques to informal question-based guidance for how to improve the quality of requirements written in NL.

2.2.2.1 Natural-Language Processing

NLP encompasses a set of symbolic and linguistic analysis tools that can be used to support the activities of a human requirements engineer, but given the current NLP state-of-the-art, work in this area stops short of being able to provide automatic and accurate semantic analysis [231, 234]. Although there is a wide breadth of work in NLP, only a subset of the research is concerned with applying it to RE. These approaches are focused on extracting formal specifications from system descriptions that are written in arbitrary (though often a restricted subset of) NL. In fact, many NLP approaches severely restrict the subset of NL that they can process, and they rely on the use of a controlled subset of NL (CNL). It is called a controlled subset because there are rules imposed on the writers: a simple syntax, a limited number of words, and a limited number of clearly defined meanings for these words. In general, in a CNL each word has only one meaning and thus can only have one part of speech. The objective is to obtain a brief, and above all, unambiguous text, though the extent to which this is achieved varies. For example, the Attempto Controlled English (ACE) project [101, 102] focuses on doing automatic parsing of requirements specifications written in a CNL, to produce translations into first-order logic and Prolog. ACE restricts NL by forbidding certain kinds of verbs (including some that we use in our DNL) and adjectives, and by limiting the acceptable phrases only to declarative sentences. ACE addresses the problem of ambiguity in an NL description by choosing a pre-defined interpretation based on its syntax. This approach thus forces requirements engineers to rewrite the description until they manage to get their desired interpretation to be what the ACE parser sees. To aid with this, ACE provides annotated NL templates for non-expert users. The main shortcoming of this approach is that in its aim to admit arbitrary (though controlled) NL, it is forced to limit itself to first-order logic and it does not try to handle temporal/ordering property specifications.
In addition to the ACE CNL, a few other generalized CNLs have been proposed [117, 155, 236], but they have not been widely adopted in practice, largely because they seem unnatural to read or write [40]. Several domain-specific CNLs have also been developed, most notably the aerospace specification standard, the Aerospace and Defence Industries Association of Europe Simplified Technical English [19]. Such CNLs are expensive to create and maintain, however, since historically they have been developed manually, by human lexicographers. In addition, they also tend to be inflexible, due to the difficulty of changing them manually. To address this, Boyd et al. [39, 40] have developed an algorithm to identify an “optimal” lexicon dynamically, by computing trade-offs between three qualities: readability, expressibility, and unambiguity. This approach is not meant to replace CNLs, but instead to complement them by enabling lexicographers to create their own domain-specific CNLs within the boundaries of automatic support for enforcing an optimal trade-off between the three qualities mentioned above. The authors assert that their approach will help with creating CNLs that are more natural to read or write because they are customized for the particular domain and can automatically change as the domain evolves.

In addition to NLP approaches that rely on a CNL, there are also approaches that use a much less restricted grammar. For example, the Cico/Circe approach developed by Ambriola and Gervasi [9, 10] focuses on automatically parsing a restricted subset of arbitrary NL into propositional logic, using a “fuzzy language” algorithm with heuristic optimizations. Like ACE, the CARL system, which implements the Cico/Circe approach, also provides suggested NL phrases for common relationships between propositions. Although the Cico/Circe approach does not impose rules on the writers of requirements specifications, this approach has similar limitations to those of ACE: since it restricts the admitted NL grammar even less than ACE does, it must limit itself to translating the NL specifications into propositional logic. Propositional logic can be used to describe some high-level requirements, but not details about how a system should behave at the temporal/ordering level. In addition to translating the NL specifications into propositional logic, the CARL system can also do consistency analysis with respect to the set of requirements specifications. The NL statements are parsed into first-order logic but are analyzed in terms of propositional logic: if both the true and false valuation of a primitive can be derived from the logical statements, then that constitutes an inconsistency [107].

Similar to our small empirical study (see Chapter 6), the authors of the CARL work did a human-subjects experiment [106] with a survey to find out (a) how close the automatic translation is to what would be created by people who are familiar with propositional logic, and (b) whether the inferences found by the formal reasoning done on the translation results are close to what domain experts would infer from the corresponding NL sentences. They had a subject pool of 15 people who were drawn from a population of academics and software development professionals (who all had either a Master of Science or Doctorate in
Computer Science). The experiment was composed of two parts: first, the subjects were asked to match NL statements with propositional logic statements, and second, to judge whether or not certain inferences were valid, based on a series of NL statements. The experimental results indicated that CARL’s translations and reasoning agreed with the human subjects’ responses in about 80% of the cases.

In addition to the two major RE-focused NL-processing approaches given above, there are a host of other research groups that have done similar kinds of processing from different perspectives. The earliest approaches [55, 113] focused on extracting key nouns and verbs from NL text to build a simple model of object relationships. Later approaches incorporated part-of-speech and sentence-construction analysis to extract more detailed object models. Liu et al. [188, 202], Bryant and Lee [46, 178], and Juristo et al. [154] have all proposed approaches that translate object-oriented requirements written in a restricted subset of NL into code skeletons in various programming languages. Mich [199] has proposed a technique for translating requirements written in a restricted subset of NL into object-oriented versions, and Overmyer et al. [215] provide a suite of basic linguistic analysis tools to aid in creating an object-oriented model from a set of requirements. Still other approaches try to do limited semantic analyses. Kof [161] proposed an approach for extracting a domain-specific ontology from a set of requirements. Michael et al. [201] have done work on providing NLP support for doing certain kinds of consistency-, conflict-, and gap-analysis of policy-based system design specifications that have been translated into a formal representation. There is also a body of work [5, 7, 91, 139, 245] that has been done to translate requirements written in several different restricted subsets of NL into various kinds of logics, including some temporal logics (e.g., Linear Temporal Logic (LTL) [220], Computation Tree Logic (CTL) [88]). In addition, some work has been done to use NLP to aid with the generation of NL instructions in healthcare. Webber et al. [255] developed the TraumAID system for generating NL instructions for multiple-trauma situations, and their system incorporates formal temporal constraints to guide the ordering of the instructions. A more in-depth survey on the use of NL generation in healthcare can be found in [51].

The strength of these NLP-based approaches is that they offer automated processing for the initial creation of a formal representation of a wide range of requirements specifications. The weakness of many of these approaches, however, is that because they are trying to process requirements specifications written in a restricted, but arbitrary, subset of NL, they are limited to being able to extract only propositional and/or first-order logic from the NL specifications. The few approaches that can handle a subset of temporal logic tend to require the NL specifications to be written in such a restricted subset of NL that the applicability of the approach is limited. Although the DNL offered by our approach is similarly limited, all of these NLP projects are much more ambitious than our approach is with respect to the use of NL: PROPEL does not attempt to extract any information from arbitrary NL specifications. Rather, we pre-define a “disciplined”
NL representation that is designed to express the particular formal concepts (i.e. the FSAs) that underlie our property specification approach, and we provide a mapping between this “disciplined” NL representation and the FSA representation.

### 2.2.2.2 Question-Based Guidance

In addition to approaches that restrict the NL phrases that can be used, there are approaches to improving the quality of NL specifications by providing guidance via questions. These questions are intended to highlight possible ambiguities in the requirements. Reubenstein and Waters [227] did early work in this area when they created an interactive Lisp-based system that takes an NL description of objects and operations that can be performed on them and asks a series of questions to refine the model, including efforts to identify ambiguities or conflicts in the terminology used. By contrast, Kamsties et al. [156] suggest an informal, manual approach. Their approach has users pretend to formalize their requirement by asking the questions that would be necessary to formalize the requirement based on a Unified Modeling Language (UML) [37] metamodel, but stops short of actually having users formalize the requirement. They describe ways of identifying several types of ambiguities and they have developed a series of domain-specific questions for their case-study domain to highlight those ambiguities. Similarly, Wasson [254] built an ontology for the terminology used in requirements specifications for a medical device, and describes several well-formedness questions for improving the quality of the ontology and thus the terminology used in the requirements. From an entirely different perspective, Baum et al. [24] suggest defining a “design space” notion that can be characterized by any number of dimensions. They have developed lists of dimensions concerning system design that they can ask system developers about in a template-like fashion and can use tool support to establish “correlations” between dimensions, which can later allow some semi-formal conflict analysis. Requirements can be design artifacts to which this approach is applied. After answering questions about a system, some kinds of implementation-level requirements can be derived from the answers.

### 2.2.3 Patterns and Templates

Other approaches to requirements elicitation offer guidance for how to specify a requirement, by providing patterns or templates. As the field of RE has developed, the potential for requirements reuse has prompted an increased interest in the patterns that arise in requirements specifications across diverse domains. There has been a growing body of work that has tried to identify aspects of requirements that can be reused, and this has given birth to investigation into both informal and formal requirements patterns, and the creation of templates to make using those patterns easier. Section 2.3.1.1 gives examples of some informal NL boilerplates, and there have also been a number of semi-formal requirement templates suggested. Some of the early work
in RE was focused on “structured specifications,” which use a standardized vocabulary or syntax and are thus formatted to be amenable to automated parsing tools. These structured specification approaches enable users to specify various types of relationships between different modules in the system design, so that things like flow graphs can be automatically generated (e.g., by specifying the input/output relationships). These approaches recognized common patterns in the relationships and often provided templates that could help to elicit what is necessary to specify instances of those patterns. For example, the Requirements Statement Language (RSL) [25], as part of the Software Requirements Engineering Methodology (SREM) [4], offers two interchangeable views: the graphical flow graph view (an R-Net) and the RSL structured specification for describing process flow, and the tool support enabled automatic translation between them. The approach made RSL templates available for use in defining certain types of elements (nouns), relationships (verbs), or attributes (adjectives). Similarly, but with a somewhat richer taxonomy of concepts, the PSL/PSA work [247] also offers structured specifications. This approach provides predefined templates for specifying the concepts, and automated support for generating flow diagrams and a data dictionary, for use in doing certain types of consistency and completeness checks. In addition to these structured specification approaches, Heitmeyer et al. [131] also suggest templates for how to specify the SCR table functions, using a combination of suggested formats for the tables and a predefined structure that includes various fields that are to be filled in with descriptive NL text.

There has also been some work that tries to incorporate more formality in the patterns used to support requirements elicitation. For the purposes of making theorem proving [137, 216] easier for certain types of real-time properties, Archer et al. [18] have leveraged common patterns in manually-created real-time proofs to create a timed-automata template. Users can fill in certain parts of the timed-automata template to customize pre-defined proofs for their needs. To make the proofs more understandable, an NL description of the proof is offered alongside the formal definition. In contrast to this theorem-proving focus, Konrad et al. [163, 166, 167] have developed a system called Hydra that tries to guide requirements elicitation via requirements patterns for embedded systems, to support formal verification. These requirements patterns include UML structural and behavioral diagrams that have formal semantics associated with them, relationships with recommended Design Patterns for implementation purposes, and for verification purposes, an associated set of recommended formal requirements based on the Dwyer et al. [81] property pattern approach.

In addition, van Lamsweerde et al. [72, 181, 194] suggested a method for searching through a library of previously-created requirements frameworks to construct detailed requirements from goals, and the correctness of the goal-refinement patterns for doing so are verified using formal logic. The basic idea behind this work is in exploiting the repetition of some of the basic patterns of framework designs to locate and fit in the needed pieces of a new, partially-completed framework. Requirements frameworks are made up of
well-defined specific numbers and types of inputs, outputs, pre-conditions, and post-conditions. The work describes a process of pruning out possible matching frameworks first through structural mapping and then narrowing that set down further through semantic matching. van Lamsweerde et al. suggest that finding a good match in the existing frameworks may also lead to the discovery of missing components in a developing framework that might need to be added to the design.

There are a few formal pattern- and template-based approaches that have been particularly influential in the development of the approach behind PROPEL [60, 240]. PROPEL is based largely on Dwyer et al.’s property specification pattern approach [80–82], which identifies commonly-occurring types of property specifications and attempts to provide users of finite-state verification tools with high-level, formalism-independent abstractions for dealing with those types. The property pattern approach describes the property types in terms of a small set of scopes and behaviors, a subset of which is the basis for the scopes and behaviors that we use in PROPEL. For each property pattern, the property pattern approach provides mappings into the input formalisms for some finite-state verification tools, examples of known uses, and relationships to other patterns. The mappings to various specification formalisms involve a number of choices. For instance, in state-based formalisms, Dwyer et al. chose to take the interval in which the property is to be evaluated to be closed on the left and open on the right. Thus, the scope consists of all states beginning with the starting state and up to but not including the ending state. They chose closed-left open-right scopes because these were relatively easy to encode in formal specifications and were the most commonly encountered in the real properties that they had collected. They recognized, however, that other variations of the scopes might be required, such as open-left open-right scopes, and their web site [80] includes notes on how to modify the mappings to obtain such variations. These notes also discuss such issues as combinations of the patterns and which instantiations of parameters in the patterns are safe in which formalisms.

Someone who wishes to modify a property specification pattern, however, must have significant expertise with the particular specification formalisms utilized by the finite-state verification tool being applied. The property patterns themselves do not highlight the choices made and the notes do not attempt to point out all plausible modifications. It is assumed that the requirements engineer who wants to verify a particular property can identify the ways in which it might differ from the particular forms in the property pattern approach and, with some assistance from the notes, make the necessary modifications. Since the target audience for the property pattern approach is users of finite-state verification tools, and expertise with the specification formalisms is a prerequisite for effective use of such tools, this is not an unreasonable requirement.

In PROPEL, however, we are concerned not only with supporting property-specification experts, but also with eliciting precise and rigorous requirements from system stakeholders who are unlikely to be fluent in temporal logics or other specification formalisms. We are thus especially interested in identifying the possible
variations and determining which of these are intended, and in representing properties and their possible variations in a more understandable way, via both DNL and FSA templates. Our focus is on pointing out the various ways in which a high-level requirement might be interpreted and on helping the specifier elucidate the property by making informed choices between these interpretations. We have thus extended the property specification patterns by including these variations in the patterns and by providing multiple views of the property specification, including both an NL-based view, DNL, and a formal view, the FSA.

This idea of combining an NL-based view with a formal view was mentioned early in RE, by Brooks [45], and this combination has been used in several approaches. In addition to PROPEL, there is other work that has also extended the property pattern approach, using both NL and a formal view. While our approach extended the Dwyer et al. work by introducing more attention to the subtle details within the property specification patterns, the Konrad and Cheng [164, 165] work extends it by moving the concept of commonly-occurring types of property specifications into the real-time domain. Like the Dwyer et al. work, Konrad and Cheng offer translations of their real-time specification patterns into a number of real-time formal logics. Based on this work, Konrad and Cheng have also developed a tool that lets users choose from NL descriptions for every scope and behavior in the Dwyer et al. property pattern approach and every real-time behavior from [165], and the tool will produce the associated temporal logic mappings. The NL descriptions in the Konrad and Cheng work and in the Dwyer et al. work are not necessarily a good basis for developing precise property specifications, however: Salamah et al. [232] did a study where they found ambiguities and conflicts in how the NL descriptions could be interpreted when they tried to create their own LTL mappings based solely on the NL.

Like the Konrad and Cheng work described above, Mondragon et al. [204, 205] extend the property specification patterns in Dwyer et al. [81]. In this case, the property specification patterns are extended via Composite Propositions (CPs), which are logical formulae that compose multiple proposition primitives in a predefined way (e.g., conjunction). Mondragon et al. offer NL and formal-logic descriptions of these CPs and their own versions of NL descriptions of the Dwyer et al. scopes and behaviors. While these NL descriptions map to the underlying formal logics that Mondragon et al. support, they are not intended to be used as property specifications: they are mainly just expansions on the original brief descriptions of the scopes and behaviors that were given by Dwyer et al., for the convenience of the users of their Prospec tool. In contrast to this extension to the Dwyer et al. work, our QT, FSA template, and DNL template property views are aimed at clearly expressing a number of significant variations in the subtle details in the property patterns that we support and their fully-resolved forms can be used as property specifications.
2.2.4 Specification Mining

Unlike the approaches described so far, which try to incorporate some type of informal or semi-formal elicitation techniques, there are a number of approaches that try to reverse-engineer requirements based on formal system specifications or code already written to implement the system. There are a variety of different techniques that can be applied to mining specifications, including dynamic analysis of execution traces, static analysis of formal specifications or code, data mining in a code base, and semi-automatic approaches that require more human involvement. For example, Ammons et al. [11] proposed a machine-learning approach that observes execution traces and iteratively refines an FSA that describes both control and data dependencies, based on what the authors call “frequent interaction patterns.” Similarly, Cook et al. [61] also aim to use machine learning to refine an FSA, except that they do not assume perfect execution traces. The weakness of these approaches, however, is that in attempting to learn an arbitrary FSA, they are tackling an NP-hard problem [112] and are thus not scalable. There have been less ambitious machine-learning approaches to specification mining, however. For example, Ernst et al. [89] have proposed Daikon, which is a tool that uses dynamic inference to extract simple “likely” invariants based on an examination of a set of execution traces that contain information about variable settings. Daikon is a statistical machine-learning approach that assumes perfect execution traces and thus cannot guarantee soundness by itself, although the inferred invariants have been used as input to a static analyzer [209], as a means of verifying their correctness with respect to the actual code.

Although Daikon avoids the scalability problems by learning simple invariants instead of arbitrary FSAs, there have been a couple approaches that avoid the scalability problems by limiting the identifiable FSAs to a subset of the properties that PROPEL supports. For example, Yang and Evans [263, 264] have proposed a dynamic approach that analyzes execution traces to determine whether a few variations on the Global Response property [81] hold, given a set of execution traces. In the same vein, Chang et al. [52] build on our property pattern templates to create a set of inference templates, which achieve the same purpose as Yang and Evans did, although in a more efficient and scalable way, at least in terms of the number of different types of properties that can be identified. Like the machine-learning approaches described in the previous paragraph, Chang et al.’s approach assumes perfect execution traces, but Yang et al. have recently extended their approach to handle imperfect execution traces [265], with the goal of being able to support more realistic code bases.

One of the major weaknesses of dynamically mining requirement specifications from execution traces is the inability to guarantee soundness, and one way to address this issue is to instead (or also) use a static analysis approach to extract requirements from code or formal system design specifications. Some static analysis approaches examine object-oriented code directly and just find simple invariants [97] or interface
assertions (i.e., non-null checks and simple bounds checks) [94], in the hopes that at least semi-automating the process of extracting the “boring but necessary” assertions will free up system developers to write the more difficult assertions. Other approaches, also based on examining code directly, aim to find function-pair temporal constraints (e.g., “a call to function X must not be followed by a call to function Y”), using just static analysis [8] or a combination of dynamic and static analysis [258]. Still other approaches use static analysis of code to generate state-based specifications of a high-level design of the system [38]. Taking the opposite approach, Gurfinkel et al. [119] have developed a temporal-logic query checker that tries to learn missing/implied properties by using model checking [57] to interact with a formal specification of a system’s high-level design. In the example given by Gurfinkel et al., the queries are propositional formulas that are specified in a multi-valued extension of CTL, where one or more of the propositions are replaced by placeholders, and the high-level design of the system is given as a Software Cost Reduction (SCR) specification. The query checker finds the strongest solution(s) to the query and returns those as the discovered properties of the system, thus providing the opportunity to examine how they compare to the desired properties of the system without needing to build and execute the system first.

In addition to dynamic- and static-analysis techniques for mining specifications, there have been a few approaches that leverage data-mining techniques to identify patterns that might indicate that certain temporal constraints must hold in object-oriented interfaces. For example, Li and Zhou [186] use data-mining techniques to automatically extract simple function-pair temporal constraints by examining an entire code base and seeing the order in which functions are called. Similarly, Livshits and Zimmerman [189] use data-mining techniques to automatically extract likely error patterns—and thus the temporal constraints to avoid them—by examining the version histories of a code base and seeing how function calls are reordered when bugs are fixed. This approach also uses a dynamic analysis technique to confirm possible temporal constraints that are identified during the data-mining activities. In general, data-mining techniques require an enormous amount of data to reliably identify constraints, however, so they are unlikely to perform well on infrequently-called functions.

Sequence-Based Software Specification [223] is another approach to specification mining based on examining execution traces, but this one relies heavily on human interaction. This technique is basically the enumeration of the entire space of system event-sequences and then the elimination of the undesirable sequences one by one until a black-box specification is arrived at, where the elimination is done by the requirements engineer or a domain expert. Then, based on “sequence abstractions” that are designed to reduce the enumeration space, concrete events are replaced by more abstract events that represent disjunctions of the concrete events, and certain sequences are recognized as equivalent for the purposes of the system design. Coming up with rules that cover all the desirable sequences is helped in part by the abstractions, although no particular
formalism for expressing the rules is advocated. This approach can be partly automated: given an alphabet of events, it will generate all sequences and assist with organizing abstractions. This is basically the brute-force approach to figuring out what the desired system behavior is with respect to a particular alphabet of events. The benefit of this approach is that it provides some level of completeness, if the requirements engineer or the domain expert is willing to slog through every possible sequence to mark it as “desirable”, “undesirable”, or “don’t know”. This approach does not scale well, however, because as the event sequences grow beyond “toy” sizes, there is effectively an infinite number of event sequences to mark, even with the use of the sequence abstractions.

2.3 Requirements Modeling

Requirements modeling is focused on activities that aim to informally or semi-formally describe what humans know or believe about non-mathematical aspects of a system and its environment, such as a “common-sense” understanding of the natural world or the political dynamics of a social organization [116]. These activities are usually focused on building a rough first pass at an ontology of the system’s domain or a sketch of a flow of system users’ interaction with the system, often by using NL or an informal graphical notation. Requirements modeling approaches can be oriented around different types of concepts, such as relationships between system stakeholders, goals, or scenarios. The approaches in this area thus tend to be informal or semi-formal early-stage exploratory techniques. Informality provides a low barrier for stakeholder adoption, since it tends to be easier to create and update informal or semi-formal specifications than formal ones. For these reasons, modeling activities are often interleaved with elicitation activities, and the models are iteratively refined as their development suggests new areas for elicitation and vice versa.

2.3.1 Natural-Language Approaches

Describing requirements in NL is probably the most common [200] informal modeling approach used in practice, and there has been work in using boilerplates or limited linguistic analyses to make recommendations for how to improve the quality of requirements stated in NL. In addition to these approaches, most of the NL-based elicitation approaches described in Section 2.2.2.1 can also aid with improving the quality of NL requirements.

2.3.1.1 Boilerplates

There have been a number of approaches that have suggested templates, or boilerplates, as a means of improving the quality of NL specifications. Two of these approaches, ACE and Cico/Circe, were mentioned in Section 2.2.2.1. At the level of single words, the ACE lexical editor offers annotated NL templates to
guide non-expert users. These are templates for lexical entries in the ACE database and they are meant to add information that enables semantic analysis. For example, if a non-expert user wants to add a noun to the lexicon, ACE prompts the user for number, gender, and type information about that term. At the level of sentence fragments, the Cico/Circe tool includes suggested phrases for expressing relationships between artifacts. There have also been several approaches that suggest sentence-level templates. Hull et al. [144] provide several “the system shall”-type boilerplates, such as the one for a performance requirement:

The <system> shall be able to <function> <object> not less than <performance> times per <units>.

Similarly, Denger et al. have proposed Natural Language Patterns [76] that aim to reduce ambiguity in NL specifications by offering a sort of quasi-BNF, where certain nonterminals could be arbitrary NL phrases that describe some particular class of concept (e.g., “phrase that contains an event”). For example, one of their event patterns is:

<When | If> (conjunction) noun phrase (ACTOR | RECEIVER) verb (ACTION | COMMUNICATION) noun phrase (ACTUATOR | OBJECT)

This work also makes a brief stab at expressing some real-time requirements, using a recommended set of keywords to tie ideas together in common ways (though it should be noted that these are many of the same common ways, like “if...then” connectives, that some of the NLP-based approaches refer to). The Denger et al. work is not rigorous: it focuses on creating conventions and structure. In the approach that Denger et al. advocate, they manually search for the language pattern that would fit the given requirement and they use the language pattern to rewrite the requirement with the aim of making it less ambiguous and annotating it with information that can aid with building an ontology or an object model. In comparison to the general templates described above, Heninger [134] developed a number of domain-specific NL templates for the aerospace domain.

2.3.1.2 Recommendations and Heuristics

Moving beyond boilerplate restrictions to handle more arbitrary NL, there are a number of approaches that simply offer recommendations and heuristics for how to mitigate ambiguities and conflicts in NL specifications, based on limited types of linguistic analyses. Chantree et al. [53] use human-trained heuristics to determine which ‘and’ and ‘or’ connectives in NL specifications are most likely to be confusingly ambiguous, and which are innocuously ambiguous because they are easily resolved by humans’ understanding of the context. This work tries to provide semi-automated guidance to requirements engineers, to help them focus on just the confusingly ambiguous connectives and thereby reduce the effort spent on this issue. There are
also several approaches with a somewhat broader focus. Fabbrini et al. [90] suggest several linguistics-based quality dimensions against which to evaluate requirement specifications. Similarly, Breaux et al. [44] examined HIPAA documents, identified patterns in the legal language, and made recommendations for how to improve its precision. For example, they suggest replacing the use of certain ambiguous words (e.g., “may”) with less ambiguous ones (e.g., “might” or “is allowed to”). We found similar guidelines to be useful in our case study evaluation, where we looked for ways to reduce the number of words used (see Section 5.4.3.2.2) and to add clarifying notes to the text where necessary (see Section 5.4.3.2.3).

2.3.2 Functional Approaches

There are several informal or semi-formal functional modeling approaches whose aim is to provide a general framework for sketching concepts and relationships between them. For example, the Structured Analysis and Design Technique (SADT) [229] provides a basic boxes-and-arrows notation for reasoning about data and control flow. Similarly, Loucopoulos and Champion [190] recommend using concept maps and informal conceptual graphs to aid with modeling the results of requirements elicitation activities. One of the most popular informal/semi-formal modeling approaches in industry is the Unified Modeling Language (UML). UML is a primarily graphical notation designed to offer the maximum amount of flexibility in the kinds of information that it can express, and it includes diagrams for modeling system users and their environment (e.g., use case diagrams), the structure of a system (e.g., class diagrams), the control and data flow of a system (e.g., state machine diagrams and activity diagrams), and the desired behaviors of a system (e.g., sequence diagrams and scenario diagrams). This desire to offer maximum flexibility in what UML can express causes the language to suffer from a lack of clear semantics, however [109].

A more targeted functional modeling approach that has received attention recently is the i* work [268], which is aimed at modeling intentions. It supports two types of diagrams, dependency and rationale, and has a notion of human agents, who can have different roles. This work supports various types of dependencies on other agents, such as goals, tasks, and resources. One of the aims of this work is to enable system stakeholders to answer “why” questions about the system and its environment. Descriptions of the system are represented using a semi-formal graph-based model to capture dependency relationships between agents that operationalize goals and produce resources that are used by other agents. The i* work has been extended in a number of ways, such as adding support for doing profitability analysis with e3value modeling [114], and adding support for Belief-Desire-Intention [43] reasoning via the Tropos [50] approach.
2.3.3 Goal-Oriented Approaches

One area in modeling that has received particular focus is goal modeling. The high-level objectives that a system is intended to meet are referred to as goals. Goals can vary from high-level, non-functional requirements to specific details about the system’s services that can be operationalized, that is, mapped to particular system functions and perhaps even represented in some kind of precise formalism. Identifying goals is an important part of understanding the system’s boundaries and can help to bring implicit and competing system stakeholder expectations to light. Eliciting and modeling abstract goals can also help system stakeholders to focus on understanding the system domain and boundaries, rather than trying to come up with solutions to perceived problems.

There has been a great deal of work in developing goal-oriented requirements elicitation and modeling approaches, and it can largely be categorized by the extent to which each approach supports formalizing the goal models. One group of approaches has tended to avoid incorporating explicit operationalization links and formal notations. The Goal-Based Requirements Analysis Method (GBRAM) [12, 15] is the most prominent example of this work and it spans a wide variety of RE activities, from initial elicitation to some analysis. GBRAM is intended to help requirements engineers identify goals and actors and begin a semi-formal organization of them. This approach also elicits information about constraints on the goals and actors and makes suggestions for how to do goal refinement and an initial analysis of goals’ benefits and costs. GBRAM is based on the Inquiry Cycle [222], which is a set of guidelines and recurring questions designed for interviews with system stakeholders. GBRAM’s approach produces requirements documents that are expressed in NL.

Another group of goal-oriented approaches to RE also support a semi-formal modeling layer, but they add a layer for explicitly associating actions with goals and formally specifying how those actions will operationalize the goals. The most prominent example of work that offers a goal-oriented connection between the informal and formal specification activities is the Knowledge Acquisition in autOmated Specification (KAOS) approach [32, 71, 174, 182]. The KAOS approach is two-tiered: there is a semi-formal layer for specifying goals and their relationships and there is a formal layer for specifying the expected behaviors of the actions that will operationalize the goals. KAOS provides constructs for representing goals, actors, constraints on them both, actor roles and capabilities, a notion of actor assignments and heuristics for determining which actors are “best” suited for being assigned to operationalize particular goals. KAOS also provides a framework of goal categories and support for formalization of goal constraints. Some of the goal-refinement work done in KAOS overlaps with later work done in GBRAM, and KAOS contributes an additional taxonomy of goal-refinement patterns that provide some guidance for what kinds of refinements are possible,
given the goal’s category. For a more detailed exploration of goal-oriented modeling approaches, see van Lamsweerde’s survey [173].

2.3.4 Scenario-Based Approaches

Scenarios are used to describe a specific case of a general type of interaction with the desired system, and they are usually expressed in terms of a sequence of actions or events. Although the distinction between scenarios and use cases is not always clear in the literature, there is some consensus that a use case in the UML sense encapsulates a family of related scenarios. Because nearly all the research in this area is concerned with analyzing individual scenarios, however, we will not explore use cases in any depth here. A scenario can capture an example instantiation of a related set of requirements, assumptions about their relationships, and assumptions about the system’s environment. A scenario is often described informally in NL or in some kind of semi-formal graphical notation, such as UML sequence diagrams. The informality and story-like nature of these notations can make using scenarios desirable because it allows them to be an effective means of communicating with system stakeholders who are domain experts but who are not familiar with formal specification approaches.

The research efforts in scenario-based requirements modeling vary widely in their actual use of scenarios. Some approaches are designed to provide a means of creating scenarios, and other approaches assume the existence of scenarios and use them for other purposes. Manual approaches to scenario creation suggest guidelines for how to structure and fill in the details in the scenarios. Examples of manual approaches include the Inquiry Cycle [222] and storyboards [47,127]. Storyboards provide a narrative template for describing the data and control flow in system users’ interaction with the intended system. Storyboard notation generally breaks the flow into discrete chunks and uses one graphical “comic-strip” panel and/or one NL paragraph per chunk. Storyboards have been used in a number of domains, including healthcare. For example, Frean [99] used them to describe system stakeholder expectations for communication between elder-care facilities, physician offices, and hospitals. In addition to the manual approaches for scenario creation, there are other approaches that provide some mechanism for generating them semi-automatically. For example, CREWS-SAVRE [246] takes in descriptions of resource models, object-linking (e.g., hierarchy) models, descriptions of actors and the actions they can take, etc., and generates the set of possible scenarios that the models allow. These scenarios can then be used to refine the input models or to act as expressions of system requirements.

In contrast to scenario-creation approaches, there are a number of approaches that assume the existence of scenarios and then use them to guide goal modeling. There has been work done by van Lamsweerde et al. [176] that extends the KAOS approach to be able to infer goals based on a learning algorithm that takes scenarios and generates a set of goals specified in the KAOS temporal logic, such that the positive scenarios
are included and the negative scenarios are excluded. Antón et al. [14] have extended the GBRAM approach to support identifying hidden or implied scenarios. This extension is based on negating a goal, identifying what obstacle conditions would satisfy the negated goal, and then discovering what possible scenarios would give rise to those conditions. The focus in the Antón work, however, is not so much to elicit new scenarios, but rather to move up to a more abstract level once the appropriate scenarios are identified and use the information to elicit new goals.

Goals are not the only requirements artifacts that can be derived from scenarios: there has also been work in deriving new scenarios from existing ones. Uchitel et al. [250] have done work with negative scenarios (instead of the more common focus on working primarily with positive scenarios) to find the places where requirements have been missed. In a way similar to how the Antón work is based on negating a goal and identifying what obstacle conditions would satisfy that negation, Allenby and Kelly [6] proposed a way to try to discover missing requirements by systematically looking at each available scenario and asking how aspects of it could fail. Letier et al. [180] have also looked at various ways in which a positive scenario can be negated by faulty event sequences and they use the results of their analysis to discover new positive and negative scenarios.

2.4 Formal Property Specification

Although informal and semi-formal requirements models are a valuable part of the RE process, validating them and using them to verify a desired system tends to be a primarily manual process, so there has been a great deal of work done to explore automating the process. An essential part of this work is focused on exploring formal property specification languages. A formal property specification is an encoding in some formal notation of a specific characteristic that a desired system should conform to. A specification language is considered formal if it has well-defined rules for describing its grammar (the syntax), well-defined rules for interpreting expressions in an unambiguous and meaningful way (the semantics), and an underlying proof theory that allows information to be inferred from a specification written in it. One important use of formal property specifications is as input to formal verification tools that can automatically determine whether a system conforms to its specified requirements. These verification activities are supported by a range of approaches, from theorem proving [137,216] to model checking [57] to test-case generation [257]. Although formal verification approaches offer more automated support than the informal and subjective validation approaches do (see Section 2.6), they have some drawbacks due to the difficulty of getting system stakeholders to understand formal property specifications [142].
<table>
<thead>
<tr>
<th>LANGUAGE</th>
<th>SPECIFICATION PARADIGM</th>
<th>PRESENTATION STYLE</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>History-Based</td>
<td>State-Based</td>
</tr>
<tr>
<td>Alloy</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Alloy + relational calculus</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>B</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>BLAST queries (higher-level)</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>BLAST queries (lower-level)</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>BSL</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>CASL</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>CTL</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>DEP</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Document-Driven Inspection</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Duration calculus</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Fluent-LTL</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>FormalCheck queries</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Formal Tropos</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>FSA</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>GIL</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>GIST</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>INCA</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>INFOLOG</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>TFL</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>KAOS temporal logic layer</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>LOTOS</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>LTL, J-scopes extension</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>LTS, SELTS/TTM</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>MEDL</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>MTS</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>PAISLey</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Petri Nets</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Process algebras</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Promela</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>QRE</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>RSML</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>RT-FIL</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>SCR</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>SETL</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>SLIC</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>SpecTRM-RL</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Statecharts</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>StTeFSPIL</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Timeline Editor</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>TLA</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Trace Assertion Method</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>UML use cases (formal)</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>VDM, VDM++</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>VTS</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Z, Z++</td>
<td>✓</td>
<td>✓</td>
</tr>
</tbody>
</table>

Motivated by a desire to leverage the understandability of the informal or semi-formal modeling approaches mentioned in Section 2.3, some of these approaches have been extended to include support for transitioning into formal property specification activities, by providing both informal modeling constructs and formal property specification languages. For example, Tropos has a formal layer [103] and there has been work to add formal semantics to UML diagrams. Li et al. [185] proposed a formalization of the UML use case diagrams where they use a relational algebra to specify the use cases in terms of pre- and post-conditions, and the same algebra to specify a set of state-invariant properties that hold in those use cases. Li
et al. then used these formal semantics to determine whether new use cases conflict with the state-invariant properties that have been specified. Konrad and Cheng proposed an approach that adds formal semantics to UML and then performs automated, real-time analysis based on translations of UML diagrams into LTL and Promela [166]. Taking a different approach, Kilov and Rumpe [158, 159] have tried to coordinate efforts to bring more formalism into UML when it is used for business modeling, including proposing a set of Business Process Specification Patterns that are specified using the authors’ version of formal UML semantics. These patterns are composable: for example, the “assessment” pattern is made up of the “information collection” pattern, followed by the “decision” pattern. Kilov and Rumpe also suggest organizational patterns, such as “multiple subtyping hierarchies,” where the same group of people can be organized into different hierarchies, in terms of job positions or educational levels, etc., based on the desired perspective.

Although the approaches mentioned above have all been proposed recently, the branch of RE concerned with formal property specification includes some of the earliest requirements-focused work and it represents much of the research done in RE, at least in terms of the sheer number of different approaches suggested. These languages have different strengths and weaknesses and can be used for different types of automated analyses. In the following sections, we discuss a sample of the formal property specification languages that have been proposed. There are a number of different ways to categorize formal property specification languages, but each way presents some problems. As shown in Table 2.1, the organization that we use here classifies them in terms of the five paradigms that van Lamsweerde [172] has identified: history-based, state-based, event-based, functional, and operational. Although this organization can provide a sufficient context for partitioning most of these languages, some of them can fit into more than one paradigm, depending on how various aspects of their notation are used. In addition to using van Lamsweerde’s paradigms, we also note the predominant presentation style(s), textual, graphical, or tabular, that are used with each of the selected formal specification languages.

### 2.4.1 History-Based Specifications

The first type of formal property specification that we explore can be viewed as “history-based”, or temporal. Formal property specification languages that fit into this paradigm are designed to describe “the maximal set of admissible system histories (or ‘behaviors’) over time” [172]. The properties usually come in the form of logical assertions about objects in the desired system and make use of temporal operators that refer to past, current, and future system states. These assertions are interpreted based on different views of how time is structured (discrete, dense, or continuous) and on whether they refer to single points in time or to bounded intervals that span from one point in time to another. Examples of this type of formal property specification language include LTL [220] and its “J-scope” extension [126], CTL [88], Temporal Logic of Actions
(TLA) [169], INFOLOG [95], Interval Temporal Logic (ITL) [206], Formal Tropos [103], and Graphical Interval Logic (GIL) [77].

Some work has been done to bridge between various aspects of history-based property specification formalisms. The Bandera Specification Language (BSL) [63,64] is an extensible property specification language that is designed to function as an abstraction of many history-based formalisms and can map into several of them, based on what is actually being used (e.g., LTL or CTL) to do model checking. There has also been work done to formally specify properties that are concerned with both events and states. Giannakopolou and Magee have introduced Fluent-LTL [108], in which they add an event-based structure to LTL and aim to encode both state and event by defining boolean state variables in terms of occurrences of their associated events (the “fluents”). In some ways this work can be viewed as the opposite approach to the one that Paun and Chechik [54, 224] have taken, in which they aim to describe both state and event by defining events in terms of state changes.

There has also been a particular focus on the development of real-time, history-based property specification languages, many of which extend the temporal logics previously mentioned and can refer to the same kinds of time structures. These real-time languages are generally composed of a set of the same types of temporal operators as are in the temporal logic languages, with additional constructs for reasoning about real time. Examples of real-time, history-based formal property specification languages include the duration calculus [122, 226], the KAOS real-time temporal logic layer [71], Real-Time Future Interval Logic (RT-FIL) [225], the FormalCheck query language [266], and the Meta Event Definition Language (MEDL) [160]. Further examples of history-based formal property specification languages can be found in Bellini et al.’s [27] survey of real-time temporal logics.

### 2.4.2 State-Based Specifications

Another type of formal property specification can be categorized as “state-based”. Formal property specification languages that fit this paradigm usually define invariants for system objects and pre- and post-conditions for the methods or operations that can be performed by the system, in terms of system state at some arbitrary snapshot. In these languages, a pre-condition would specify the weakest necessary condition for an operation to be applied to a particular object and a post-condition would specify the strongest possible effect that could be exhibited by the output states after that operation is applied to that object.\(^1\) There are a number of examples of this state-based type of formal property specification language, including Z [2,221,244], B [3], Alloy [146], VDM [151], and object-oriented extensions, VDM++ and Z++ [177]. Further discussion and

\(^1\)Although the state-based paradigm definition includes pre- and post-conditions, it should be noted that such conditions can also be used as guards on transitions. There is thus some overlap between this state-based paradigm and the transition-based paradigm.
comparisons between the expressive powers of some of the various state-based formal property specification languages can be found in Duan’s survey [78].

2.4.3 Transition-Based Specifications

A third type of formal property specification can be categorized as “transition-based”. Formal property specification languages that fit this paradigm are designed to describe the admissible transitions between system states. Properties specified in this type of language can be described by a set of state machine transition functions. Each transition function takes a system state and a trigger event as input and produces a corresponding output state. The trigger event forces the transition to occur, although some languages in this paradigm also use pre- or post-conditions to guard whether the transition can occur. One prominent example of a transition-based property specification approach that can have guards on its transitions is Software Cost Reduction (SCR) [130,132]. System properties in SCR are expressed in terms of monitored variables, which are inputs to the system and define the system states (called “modes”), trigger events that cause transitions to new system states and that occur when monitored variables change their values, and controlled variables that represent system outputs. SCR provides a tabular representation for the properties that is designed to help with the organization of the variables and trigger events. SCR is more than just a property specification approach, however. It also includes formal analysis capabilities to automatically check for certain kinds of internal syntactic consistency and completeness in a set of property specifications.

Some transition-based property-specification approaches leverage a combination of graphical and tabular notations. For example, Leveson et al. [129,183] have developed the Requirements State Machine Language (RSML), a real-time, transition-based, property specification language. In addition to having transitions, RSML has trigger events and conditional guards that keep the transitions from occurring until the appropriate conditions are in place. The graphical representation of RSML was originally based on the transition-based, semi-formal specification language of Harel’s Statecharts [124]. Because of this history, RSML’s graphical notation also provides some support for abstraction, allowing the grouping of states into superstates, for example. A formal semantics for Statecharts has been proposed [123] and supported by the STATEMATE tool [125], and RSML similarly offers a formal semantics for its graphical notation and provides tool support. Another strength of RSML is that its tabular representation of AND/OR logic for describing the conditional guards was easier for the system developers who used this approach to understand than a more text-based mathematical notation [128]. One weakness of RSML compared to Statecharts, however, is that there is no termination guarantee on the RSML step semantics. Leveson et al. also introduced a new event-based formal property specification language, SpecTRM-RL [184], which is based on the same underlying RSM specification. The authors had observed that requirements engineers were not limiting themselves to creating
only black-box property specifications, as was the original intention of RSML, so this new languages makes some additional restrictions on the kinds of properties that can be expressed.

Taking a different approach to a transition-based formalism, PROPEL relies on FSAs, which do not make use of conditional guards on the transitions or a tabular presentation style. To make its property-specification approach more understandable to requirements engineers, PROPEL provides a graphical representation of the FSAs that it supports. There are a number of other transition-based languages that offer graphical representations. Smith et al. [238] have created the Timeline Editor, which can describe optional events, required events, absence events ("fail events"), and something approaching a combined state- and event-based formalism. The graphical Timeline Editor converts property specifications into Büchi automata. Braberman et al. [41] have developed Visual Timed Event Scenarios (VTS), which is a graphical state-machine notation for specifying real-time, event-based negative scenarios. These graphically-defined scenarios can be used to generate timed automata to support real-time model checking.

There have been a number of other transition-based formal property specification languages and approaches suggested from several different perspectives. Similar to FSAs, Olender and Osterweil [213] proposed Quantified Regular Expressions (QREs) in their Cecil/CESAR work. PROPEL makes use of a subset of the behavior and scope taxonomy proposed by Dwyer et al. [81], and this notion of scopes is similar to the anchors that are used in the Cecil/CESAR language for expressing constraints on event sequences. Declarative Event Patterns (DEPs) [252] are a means of specifying acceptable sequences of events and handling unacceptable sequences via an aspect-oriented programming paradigm. The INCA query language [62] offers support for specifying properties in terms of event-delimited intervals from an integer-programming perspective. The BLAST query language’s lower-level tier [33] is based on automata and can be used to specify properties about execution traces. SLIC [21] is designed to allow specification of temporal properties about how APIs written in C can be called (function calls are the events); this is an attempt to formalize the interface requirements that were previously captured in NL in code comments. Taking an approach that more often specifies a system model rather than properties about that model, the Simple Programming Language (SPL) [193] and Promela [143] both provide a means of defining a transition system that can be used as input to their respective model checkers, STeP and SPIN.

2.4.4 Functional Specifications

A third type of formal property specification can be categorized as “functional,” sometimes also known as “relational” or “algebraic.” Formal property specification languages that fit this paradigm are designed to describe a system as “a structured collection of mathematical functions” [172]. While Parnas did not invent the notion of a functional property specification, he was the first to bring it into a software-engineering
There are a number of approaches in this functional specification paradigm. Parnas's Document Driven Inspection method [218, 237] allows requirements engineers to formally specify system behaviors functionally and represent these property specifications in a tabular form [148] that is designed to be amenable to manual inspection. Similarly, the Trace Assertion Method [65, 149, 253] allows properties about event traces to be specified functionally, and presented in a tabular format. The BLAST query language's higher-level tier [33] provides support for specifying properties in terms of relational queries, and the SET language (SETL) [235] enables properties to be specified functionally, based on set theory. Building on a language that existed previously, Frias et al. [100] have recently done work in adding a relational calculus to Alloy so that reasoning can be done in this context in addition to what can be done in Alloy's original state-based formalism. In addition, the Common Algebraic Specification Language (CASL) [145, 195] is a general-purpose functional property specification language that has been the focus of a standardization effort and has been extended into several other formal property specification paradigms. Further discussion and comparisons between the expressive powers of some of the various functional property specification languages can be found in Duan's survey [78].

### 2.4.5 Operational Specifications

The final type of formal specification that is discussed here can be categorized as “operational”. Formal specification languages that fit this paradigm are designed to describe the system as “a structured collection of processes that can be executed by some more or less abstract machine” [172]. Some early languages, including LOTOS [36], PAISLey [269], GIST [22, 92], Petri nets [168], and process algebras [138, 203], fit into this formal specification paradigm. Some recent work has also been done in this area, such Labeled Transition Systems (LTS) [191] and the real-time extension, SELTS/TTM [272]. In addition, Uchitel et al. [249] create an operational specification of system behavior based on using scenarios and property specifications as input. This specification is represented using Modal Transition Systems (MTSs), which are capable of representing and distinguishing between possible and required behavior.

### 2.5 Requirements Organization

Organizing requirements is not generally considered a sub-area of RE, per se, but it is a cross-cutting RE activity that complements the other activities and should not be ignored. Requirements organization activities help to provide structure for what is discovered in the other activities. Organization activities aim to provide categories (e.g., “functional” vs. “non-functional”) or frameworks (e.g., requirements specification document-level structures, AND/OR decompositions of sub-requirements), within which to understand relationships between individual requirements and between requirements and other artifacts in the system.
development life cycle. There is not a significant amount of work specifically in this area; most of the work is part of research whose main aims are to address issues in other RE activities. Thus, various ways of organizing a set of requirements, such as from a goal-oriented perspective or from a perspective focused on building an object model, are mentioned in the elicitation and modeling sections. If an ontology is created through the elicitation and modeling activities, it can be applied to aid with organizing the set of requirements. For example, if the ontology indicates that a certain human role in a system is responsible for a particular set of tasks, it might be reasonable to collect all the requirements that involve those tasks into one section of a requirements document, or to tag all those requirements with the role’s name in a requirements management system. Such an organization might make it easier for the domain experts who are familiar with that role to validate that set of related requirements. In general, organization activities are done throughout the course of doing the other RE activities, whenever the set of requirements reaches a level of complexity that requires that structure be added to enable the system stakeholders to use the requirements successfully. The work in organizing requirements generally falls into three categories: techniques for finding structure, techniques for structuring requirements documents, and strategies for expressing relationships between requirements.

2.5.1 Techniques for Finding Structure

There has not been much work that is explicitly focused on how to find structure in a set of requirements, although some NLP approaches [234] can be cast in this light, since they can be used to group requirements that use similar terminology. Such NLP approaches stop short of being able to suggest structure beyond a rough clustering of the requirements, however. To define a richer structure, it is often necessary for requirements engineers and system stakeholders to manually examine the set of requirements. Brassard [42] presents two techniques that can be used to elicit an organization from system stakeholders: affinity diagramming and card sorting. These techniques have uses outside of RE, but they can be applied to organizing requirements. Both techniques involve writing the individual requirements on cards and then having system stakeholders iteratively group the cards. In affinity diagramming, system stakeholders label the groups with categories and then assign priorities or other attributes to requirements in those categories. This technique could be considered a manual version of the shallow semantic tagging that Sawyer et al. [234] do in their statistical NLP approach, which employs a combination of lexical techniques (i.e., word frequency profiling, collocation analysis, and part-of-speech analysis) to guide automatic tagging of individual requirements. Although this automated approach can aid requirements engineers, the manual approaches that involve system stakeholders tend to be more robust with respect to identifying related requirements that do not share the same terminology. In card sorting, system stakeholders label the groups with categories and then use a simple graphical flow-chart notation to describe relationships between the groups, iteratively improving the catego-
rization and finding conflicts or missing requirements. A somewhat more focused approach to finding an organization for a set of requirements is the non-functional requirement (NFR) classification and accompanying question-based guidance proposed by Glinz [110]. This work identifies three different types of NFRs, performance-related, quality-related (i.e., the “-ilities”), and environmental constraints, and provides a series of questions that can be used to group existing NFRs using this taxonomy. Such a classification can help to show where non-functional requirements might be missing or where trade-offs might need to be considered.

### 2.5.2 Techniques for Structuring Requirements Documents

Another focus of work in organizing requirements has been on techniques for structuring entire requirements documents. These documents are primarily written in NL, and perhaps also incorporate some graphical or tabular notations. Although this area has not received much attention in the research literature, it is of vital importance in practice, since a vast majority of requirements are specified using such documents [200]. Some early work in RE focused on various structured templates for requirements documents, called Software Requirement Specifications (SRSs) [75, 147], and many variations on the SRS concept have been suggested. For example, Heitmeyer [132] has recommended a document template for organizing a collection of SCR requirement specifications. Such document-structuring techniques are essentially helpful tips for making requirements documents more easily understandable, navigable, and maintainable. We applied several techniques in the informal requirements documents that we created during our case study evaluation [135]. See Section 5.4.3.2.1 for more details on our use of these techniques. Some of them are also advocated by Hull et al. [144], such as the use of a unique, decomposable numbering system, the effort to group similar requirements together in the document layout, tracking the review and satisfaction status of each requirement, and using hierarchical sections with summarizing or clarifying text. Other document-structuring techniques that Hull et al. advocate are classifying each requirement by type (i.e., functional vs. non-functional) and priority, and including references to other artifacts in the system development life cycle, such as rationale or source information, functional designs, or test plans, to support traceability. Other document-structuring techniques have also been suggested. For example, Heninger [134] did a case study in organizing a collection of requirements written in NL. Heninger represented the requirements in a tabular format [148] and observed that carefully structuring the documents in this way was essential to understanding the requirements. This approach to organizing requirements is not relevant to just NL requirements, however. For example, Bartussek and Parnas [23] also observed that using a tabular format to organize their algebraic trace-assertion specifications made navigation and analysis of those specifications easier.
2.5.3 Strategies for Expressing Individual Requirements’ Relationships

In addition to the document-level structuring techniques described above, there has also been work [116] on strategies for how to organize individual requirements and their relationships. The five strategies highlighted are: generalization / specialization, classification, parameterization, viewpoints, and exceptional vs. nominal circumstances. Often these strategies are made visible by document structure, but they are not as much about document structure as they are about the requirements relationships. Of these five strategies, we used generalization / specialization, classification, parameterization, and to a limited extent, viewpoints, in our case study evaluation. Although we specified both exceptional and nominal circumstances in the processes that we elicited during these case studies, we did not organize the properties in terms of exceptional vs. nominal circumstances, because all of the properties were required to hold in both types of circumstances. See Section 5.4 for more details on requirements relationships that occurred in our evaluation.

Generalization / specialization strategies have been used in a number of approaches. For example, the KAOS [71] semi-formal goal-modeling layer uses AND- and OR-refinements to describe the relationship between a goal and its sub-goals. Similarly, Antón [13] provides an ordered-AND sub-goal relationship and a notion of goal refinement that involves adding constraints to what it means to achieve that goal. In addition, there are a number of approaches, such as [50], that support a generalization / specialization strategy via the use of an OO-based framework for organizing the requirements.

Another organizational strategy that has been used is to assign classifications to the requirements. For example, reference models [118,120,219] separate requirements based on whether they are system stakeholder desires, a specification of the system, or assumptions that are made about the environment. NLP can also be used to automatically classify requirements, such as the work done by Cybulski [66]. This work is based on an information-retrieval system that identifies keywords in NL requirements text, classifies the requirements based on the discovered keywords, and then supports queries on the repository of classifications.

One organizational strategy that can be used with very similar requirements, besides just grouping them, is parameterization. This strategy shows which aspects of the requirements are able to differ and which must stay fixed. Weber and Weisbrod [256] were involved with managing a large number of requirements in the automotive domain, where the requirements were organized by product families. Weber and Weisbrod used parameterized requirements for the different members of the product family, to aid with requirements reuse.

A significant organizational strategy that has been used to identify and resolve conflicts among differing system stakeholders’ requirements is based on stakeholder viewpoints [96,243]. The premise behind this approach is that a system is usually not used by just one type of stakeholder, and thus its requirements should not be based on just one perspective. This approach provides a way of organizing requirements in terms of the system stakeholders who generate them, and a way of supporting conflict and consistency analysis. This
approach can also be helpful in providing structure for requirements elicitation and validation, especially when there are many different system stakeholders.

In addition to the strategies described above, one strategy that can help limit complexity during the RE process is to organize the requirements based on whether they are relevant to the nominal or exceptional system behavior. Delaying the elicitation and modeling of exceptional system behavior and incorporating it later in a structured way has been a part of several approaches, mostly notably those involving negative, or obstacle, scenarios [13, 246, 249].

2.6 Requirements Analysis and Validation

Requirements analysis and validation activities are often interwoven, because there is a complementary focus on evaluating and improving the quality of requirements and requirements relationships. Activities that could qualify as requirements analysis or validation are often interwoven throughout all the other activities in a requirements management process (see Section 2.7). For example, requirements elicitation is often interwoven with requirements analysis and validation in RE processes [210] and some approaches that support all three activities have thus been mentioned in the previous sections. The goals of requirements analysis include detecting and resolving requirements ambiguities, conflicts, or inconsistencies; discovering missing or implied requirements; or identifying attributes of the requirements, such as their functional/non-functional classification, volatility, priority, source, or justification. Based on the information that requirements analysis provides, validation techniques aid with getting system stakeholders to agree on what the requirements of the system are, despite possibly-conflicting perspectives. Research in requirements validation tends to focus on improving the quality of information available to system stakeholders, to get more effective feedback about current system designs. Requirements analysis and validation approaches range from manual, informal inspection techniques to formal techniques with automated support and well-defined semantics.

One key type of information that requirements analysis and validation approaches create are traceability links between individual requirements [67] and between requirements and the relevant components of a system design. Berenbach and Wolf [28] have reported on their experiences with Sysiphus, a tool that allows several different types of formal specifications to be unified via a single database, to support stronger traceability links. In addition to using traceability links for source and justification information, some approaches use the links to determine the effects of possible changes to the requirements [49, 58]. For example, Cleland-Huang et al. [58] support requirements traceability by keeping track of trade-off links between different non-functional requirements and establishing links between those requirements and the functional components that implement them, which are represented by UML diagrams. Requirements engineers can use
these links to evaluate the impact of proposed changes to the non-functional requirements on the functional model: by using probabilities in the links model, they can generate the set of functional components that will be impacted by the change. In a similar technique that uses code instead of UML, the peeking methodology suggested by Little [187] explicitly associates code segments with both functional and non-functional requirements and offers tool support to keep track of the relationships. This work is also an example of approaches that are concerned with connecting the requirements management process with the overall systems engineering process.

Since many requirements in practice are elicited and specified using NL, there are a number of manual and NLP-based approaches for analyzing requirements. For example, manual inspections [208] have been used to methodically examine a set of requirements and look for their relationships and attributes. Kamsties et al. [156] use inspections to detect five types of requirements relationships: overlapping, conflicting, specifying, constraining, and predicated upon. Kamsties et al. give multiple interpretations of the relationship(s) for each pair of requirements in a tabular format, to highlight the ambiguities present in the set of requirements. Breaux et al. [44] manually examined Health Insurance Portability and Accountability Act (HIPAA) documents and identified patterns in the legal language. Based on these patterns, they manually constructed a formal description of the HIPAA documents and used it to make recommendations for how to improve the documents’ precision by removing ambiguities, detecting conflicts, and stating implicit and missing information explicitly. In addition to these manual approaches, NLP-based approaches have also been used to aid with requirements analysis. For example, Rupp and Goetz [230] use NLP to detect ambiguities, conflicts, and incomplete requirements, when the requirements are written in a CNL. Similarly, Osborne and MacNish [214] use NLP to detect sentence-structure parsing ambiguities in requirements and give users the ability to choose from a set of possible parses. Each possibility is represented using a CNL re-statement and an accompanying first-order logic formula. See Section 2.2.2.1 for more work on using NLP to improve the quality of requirements specified in NL.

In addition to the manual and NLP-based approaches to detecting and resolving requirements conflicts, there have also been several approaches where the aim is to coordinate different system stakeholder perspectives, since these have been found to be a common source of conflicts. For example, Boehm et al.’s WinWin approach to requirements negotiation [34] provides a framework for describing the “win” conditions for each stakeholder involved in the negotiation, identifying conflicts between those conditions, and enumerating and refining the options available for how to resolve those conflicts. Taking a more automated approach, the viewpoints work [84, 96, 211, 212] builds on requirements traceability links by adding semantic information to support automated reasoning (some of it goal-based [171, 175, 228]) about requirements conflicts and consistency. Consistency relationships between different viewpoints are expressed via operations that can be
performed on the data model of the system and there is support for automated analysis of propagation of requirements change over time. This approach is strongly involved in the requirements process management work as well. In addition, Antón and Potts [17] propose an approach to assigning priority and resolving requirements conflicts based on viewing the evolution of requirements (and whole systems) in terms of the system users’ perspectives: that is, in terms of the services that the systems provide and the burdens and benefits that the users experience based on each version of the services. Complementary to these approaches, which are focused on elucidating and organizing the space of system stakeholder perspectives, there have been a variety of other approaches for getting insight into each individual stakeholder’s perspective, including simulations [133, 248], animations [153, 192, 251], and prototypes [105].

Moving beyond these informal and semi-formal techniques for detecting and resolving requirements problems, several approaches have suggested using formal static- or dynamic-analysis techniques. A common approach is to use model checking (e.g., [141]) to do consistency analysis. For example, the work that Letier et al. [180] have done is based on a formal language for expressing scenarios, via message sequence charts (MSCs). They have developed a technique that can derive “input-output implied scenarios” from the standard kinds of scenarios that are often used early in the RE process, which contain implicit assumptions about how different components interact. This new kind of scenario prevents the different components from placing any restrictions on other components’ behaviors. Such a scenario allows new (and possibly unforeseen) interleavings in the events generated by the various components to be found through an analysis of the original standard scenario. This approach can be used for finding undesirable event interleavings or undocumented assumptions if, for example, an interleaving is not actually possible because of assumptions about the system that are not captured by following event traces in an MSC. Similarly, Jeffords and Heitmeyer [150] developed an algorithm that can derive state invariants from SCR requirements specifications, to support stakeholder validation of those specifications. In addition to these static approaches, Feather et al. [93] have also done work to support dynamic detection of execution traces by instrumenting a system with FLEA property specifications and monitoring it with the AMOS system to see where the properties deviate from properties specified in the KAOS temporal-logic layer [71].

2.7 Requirements Management Processes

RE is often viewed as an effort that only takes place at the beginning of the system development life cycle, but changing requirements and changing expectations during system development mean that RE is actually interwoven throughout the entirety of the system development life cycle. Requirements management
processes thus try provide a coherent set of steps that progress from an initial idea to a final specification [48], and research in this area aims to address three types of challenges:

- Providing guidance for integrating all the RE activities: elicitation, modeling, formal specification, analysis, and validation.
- Evolving requirements over the lifetime of a system.
- Managing requirements and system stakeholders that are in a distributed (i.e., globalized) setting.

The nature of the requirements management process in any given system development will differ based on the system's domain, the goals of the system stakeholders, the existing system and requirements specification artifacts, and the skills of the requirements engineers involved. The research in requirements management processes thus focuses, in part, on treating requirements as configurable artifacts that evolve throughout the system development life cycle and on managing them using the same configuration-management practices as are used with other artifacts in the system development life cycle. For example, Nuseibeh et al. [212] introduced a process model in their viewpoints-focused work to provide some basic guidance for requirements engineers in choosing which types of RE techniques (i.e., formal, semi-formal, or informal) to use at which points in the requirements management process. Similarly, Berenbach [29] has compared experiences in managing requirements that are specified in UML and NL and he has come up with recommendations for how to avoid common pitfalls. The research in requirements management processes also tries to coordinate the RE activities of elicitation, modeling, specification, analysis, and validation by configuring them as needed for different types of domains, system stakeholders, and constraints. This work often overlaps with requirements analysis (see Section 2.6), because determining how to best do the requirements process configuration depends on determining the nature (e.g., priority, formal/informal representation, volatility) of the requirements being managed and on leveraging traceability links between individual requirements and between requirements and other artifacts in the system development process.

Requirements are not the only thing being managed in requirements processes: the humans involved in the process are also resources to be managed. Kauppinen et al. [157], Wiegers [259], and Sommerville and Ransom [242] have all done work in introducing RE processes to industrial software development companies and they give heuristics and recommendations for how to do so effectively. There is also a growing interest in how to manage requirements in a global systems development context. For example, Daniels et al. [70] have developed the Enterprise Analyzer, a combination of “tools, facilities, facilitation, and processes” for doing remote/distributed group requirements elicitation and refinement. Daniels et al. have evaluated their approach in two case studies with the US Army, where they defined the functional requirements of two
modules of a large integrated information system, with the involvement of people in multiple locations. Taking a higher-level view, Berenbach [31] examined how different organizational structures with varying degrees of distribution in the analysis, design, and implementation phases of software development impacted requirements for several globally-distributed projects.

Some work has also been done that takes both the humans and the requirements artifacts into account in the evaluation of the requirements management processes. For example, Damian and Moitra [69] looked at several approaches to managing a large number of requirements when the people involved in the software development process is distributed around the globe. In addition, Ebert and De Man [85, 86] have done a large study on the costs of requirements change, based on 200 different development projects. The study investigated possible factors that contribute to the volatility of requirements during projects and looked at how that volatility can be minimized. They examined several elements of the requirements processes used, for example, enforcing requirements development standards, supporting requirements traceability, and giving system stakeholders responsibility in product releases. They observed that no one of the elements, when examined in isolation, had a significant effect on reducing the requirements-change rates, but that when all of the elements were examined in aggregate, the effect was unmistakable. This suggests that effective research into RE processes must include not only support for the technical aspects of requirements development and maintenance but also components that are informed by an understanding of human psychology, sociology, and linguistics.

2.8 Summary

RE is concerned with three areas of system development: discovering how the system should behave (requirements elicitation), representing those discoveries (informal/semi-formal requirements modeling, formal property specification, and requirements organization), and managing changing expectations of system behavior (requirements analysis, validation, and management processes). Our approach, which we describe in the next chapter, focuses on supporting aspects of the first two areas, and specifically on supporting property elicitation and formal property specification activities. PROPEL provides three different views that specifiers can work with to elicit and formally specify their properties: FSA templates, DNL templates, and QTs. Although the use of FSAs, NL, and decision trees is not new in RE, our approach is novel in three ways:

- it combines all three property views;
- it automatically keeps the three property views synchronized with each other: as specifiers make a change in one of the views, the other views change correspondingly; and
• it supports elicitation guidance in all three property views by providing templates that explicitly indicate possible variations in common property specification patterns.

Although our approach does not contribute to the third area of RE (managing changing expectations), the formal nature of the FSA property specifications that are produced using our approach means that they can be used in many of the approaches to requirements analysis, validation, and change management.
CHAPTER 3
A PROPERTY ELUCIDATION APPROACH

Property specifications are often used in RE to concisely describe important aspects of a system’s behavior, where that system might involve the coordination of hardware, software, and people. These specifications can then be used as the basis for system development and validation. Ideally, property specifications should be precise enough to support automated analyses and understandable enough to be readily comprehended by all system developers. Automated analysis tools typically accept property specifications represented in mathematical formalisms, such as temporal logic. Such formalisms have not been widely adopted in requirements specifications, in part because their use requires significant expertise [172]. In practice, requirements engineers tend to write requirements specifications in natural language [200]. Although natural language may offer understandability, properties written with such informality are often ambiguous and thus are of limited value when doing rigorous analysis of the system. Additionally, because of all the subtle details that have to be considered, it can be surprisingly difficult to correctly specify a property, even one that focuses on a very limited subset of the system’s behavior. Overlooking these details often leads to inaccuracies that are not revealed until verification or testing, or perhaps even deployment. System developers may invest considerable effort trying to make sure that the system conforms to a property, only to later determine that the property has been specified incorrectly.

What is needed is a property elucidation approach that supports the creation of property specifications that are not only understandable to all system developers, but are also mathematically precise, so that they can be used as the basis for rigorous analysis of the system. Our approach is built on the property patterns developed by Dwyer, Avrunin, and Corbett [80–82] to assist users of finite-state verification (FSV) tools, such as SPIN [141], SMV [197], INCA [62], and FLAVERS [83]. They argued that the difficulty of specifying properties correctly in the various input formalisms used by such tools was a substantial obstacle to the adoption of FSV technology and indeed, to the adoption of formal property specifications in practice. To address this obstacle, they proposed specification patterns, which are modeled on Design Patterns [104], as a way to capture the experience of property specification experts and to enable the transfer of that experience to other practitioners. The property pattern approach recognized that there are certain types of properties that system developers often want to check, and Dwyer et al. observed that nearly all the properties found in
the FSV literature could be classified into a small number of what they described as “high-level, formalism-independent, specification abstractions.”

A property created using these specification abstractions, or property patterns, is composed of two parts: a behavior (or “intent,” as it is called in the property patterns work) and a scope. A behavior describes the restrictions on occurrences of states or events, and a scope describes the parts of the state- or event-sequences within which those restrictions apply. For example, the Response behavior is a cause-and-effect relationship between a pair of states or events in which the occurrence of the “cause” requires a subsequent occurrence of the “effect,” and the Before scope requires the behavior to hold from the start of the state- or event-sequence until the first occurrence of a given state or event. Dwyer et al. introduced this separation of scope and behavior because they observed that most of the informal requirements that they had encountered were specified with the assumption, implicitly or explicitly, that the requirements only had to hold within certain subset(s) of a system’s operation, and they wanted the property pattern approach to reflect this assumption.

The property pattern approach identified eight behaviors and five scopes that can be combined to create forty different properties, and the property pattern website [80] gives mappings from all the scope-behavior combinations to several specification formalisms (e.g., regular expressions, various temporal logics). Although the property patterns are extremely valuable, it can be difficult even for those who have expertise in those property specification formalisms to accurately specify a property based on the the property patterns, if the intended property varies from the forms given. We are thus especially interested in pointing out the possible variations in how the property patterns might be interpreted and in helping specifiers make informed choices about which variation is suitable for their intended property, whether or not they have expertise in specification formalisms.

To support these goals, our approach extends the property patterns in two important ways. First, instead of just parameterizing the property patterns in terms of the events or states, we extend the property patterns by providing a set of property pattern templates that explicitly indicate the alternative options associated with the property patterns, thereby helping to ensure that important subtle details are not overlooked. These property pattern templates are each composed of a scope template, which contains options related to the selected scope, and a behavior template, which contains options related to the selected behavior. Providing these options helps specifiers to consider the relevant alternatives and subtleties associated with their intended properties. Using these scope and behavior templates, specifiers can create 13 possible scopes and 147 possible behaviors, which can be composed into over 1,900 different possible properties.¹ As with the

¹These totals differ from those published in our earlier work because we have chosen to remove two scope-template options in the interests of providing a clearer separation between the concepts of scope and behavior. Given the current set of options, only behaviors can cause a property to be violated.
property pattern approach, however, the space of properties covered is not comprehensive, but is intended to support the properties that seem to occur most often. Since the majority of the properties that Dwyer et al. found in their survey are covered by only four of the eight behaviors that they identified, the current implementation of our approach supports variations on just those four. In addition, although the Dwyer et al. work includes both state- and event-based forms of the property patterns, our work currently concentrates only on the event-based forms. The second way that our approach extends the property patterns is that we present these property pattern templates to specifiers by showing multiple views, which were chosen to support precision, understandability, and guidance. We have developed a tool, called PROPEL, for “PROPerty ELucidator,” that provides support for specifying properties based on these property pattern templates, using these complementary property views. PROPEL specifiers can work with one or more of the property views at the same time and these views are automatically kept synchronized with each other: as specifiers make a change in one of the views, the other views change correspondingly.

In PROPEL, specifiers instantiate a property by making decisions about, or resolving, each of the available options in a selected property pattern template. To resolve an option, specifiers select one of the option’s possible settings. For a property to be fully instantiated, not only must all the options be resolved, but specifiers must also define the set of events, or the alphabet, that are considered relevant to the meaning of the property. Like the property patterns, the property pattern templates have between one and four pre-defined placeholders, or parameters, that can be associated with specifier-defined event names: at most two for the property’s scope and at most two for the property’s behavior. In PROPEL, a parameter’s default name is displayed in the property pattern templates until specifiers associate a specifier-defined event with the parameter. We refer to events that are associated with the scope parameters as scope delimiters, and we refer to events that are associated with the behavior parameters as primary events. Although the available behavior templates have at most two parameters, specifiers can define additional events, referred to as secondary events. Secondary events can be used when a property must constrain more than just the occurrences of primary events. For example, it may be important that certain other events do not occur in between an occurrence of a “cause” and a subsequent occurrence of its “effect,” where the “cause” and the “effect” are the property’s primary events. In such a situation, specifiers can include those other events in the alphabet of the property and define the property so that it explicitly prohibits all occurrences of those secondary events between the “cause” and its “effect.”

---

2We do not yet support the two Chain behaviors, which involve ordered sequences of states/events; the Bounded Existence behavior, which limits the number of occurrences of a state/event; or the Universality behavior, which requires that a proposition hold in all states.
PROPEL provides three different property views that specifiers can work with to resolve the options in a selected property pattern template:

- a graphical, extended Finite-State Automata (FSA) template, which offers precision;
- a Disciplined Natural Language (DNL) paragraph, which offers understandability; and
- a hierarchical Question Tree (QT), which offers guidance for selecting a property pattern template.

Together these property views are meant to bridge the gap between precision and understandability in property specification. In addition to these property views, which support both scope and behavior templates, there is also a Scope Timeline (ST), which provides an alternative graphical view of a property’s scope. Specifiers can work with any one of these views, or they may choose to use any combination of them. After specifiers select a property pattern template, the changes they make using any one of these views are reflected in the other views whenever possible. A full description of the mapping between the views is given in Appendices A.1.1, A.1.2, A.2, A.3, and A.4. As can be seen in the brief descriptions above, these three property views and the ST scope view have complementary strengths and limitations. We have proposed these views because they seem to be particularly useful, but other views could be developed and supported as well. In the remainder of this chapter, we use an illustrative example to describe the three property views and the ST scope view in detail, and we finish by comparing each of these views in terms of their design goals and briefly discussing areas where future work is needed.

3.1 Specifying a Behavior

In our approach, a property’s behavior is used to describe the desired operation of the system in terms of restrictions on the possible sequences of events that are allowed to occur. For example, the Response behavior mentioned previously is a cause-and-effect relationship between a pair of events in which the occurrence of the “cause” requires a subsequent occurrence of the “effect.” Most event-based formalisms, including the one supported by PROPEL, use an interleaved model of concurrent computation, where two events cannot coincide. Within this framework, PROPEL currently supports four event-based behavior templates, which are based on four of the eight behaviors proposed by Dwyer et al. in the property pattern approach: Response, Precedence, Existence, and Absence.

---

3Since the DNL template property view sometimes requires specifiers to resolve a group of options together while the FSA template and QT property views allow specifiers to resolve each of those options individually, the DNL template property view does not reflect changes made in the FSA template or QT property views until all the options in the group have been resolved.
Table 3.1. The Four Behavior Templates

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Response</td>
<td>If A occurs, B must occur subsequently</td>
</tr>
<tr>
<td>Precedence</td>
<td>A must occur before B is allowed to occur</td>
</tr>
<tr>
<td>Absence</td>
<td>A is prohibited from ever occurring in a given sequence of events</td>
</tr>
<tr>
<td>Existence</td>
<td>A is required to occur at least once in a given sequence of events</td>
</tr>
</tbody>
</table>

Table 3.2. Summary of the Behavior Templates’ Options

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
<th>Existence</th>
<th>Response</th>
<th>Precedence</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bounded</td>
<td>This option determines whether A is allowed to occur exactly once or more than once.</td>
<td>✓</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Nullity</td>
<td>This option determines whether A is required to occur at least once.</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Precedency</td>
<td>This option determines whether B is allowed to occur before the first occurrence of A.</td>
<td>✓</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Pre-arity</td>
<td>After A occurs, this option determines whether A is allowed to occur again before the first subsequent B occurs.</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>Immediacy</td>
<td>After A occurs, this option determines whether secondary events are allowed to occur before the first subsequent B occurs.</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>Post-arity</td>
<td>After A and B occur, this option determines whether B is allowed to occur again.</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>Finalization</td>
<td>After A and B occur, this option determines whether A is allowed to occur again.</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>Repeatability</td>
<td>After A and B occur, this option determines whether, if A occurs subsequently, the situation is the same as when the first A occurred, meaning that the restrictions described on any events that take place after that occurrence would again apply.</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
</tbody>
</table>

3.1.1 Behavior Templates

Table 3.1 gives a brief summary of the four behavior templates that are supported by PROPEL, showing their names and a description of their core concepts. These behavior templates extend the original four property pattern behaviors by making a number of possible variations in those behaviors explicit and by presenting these variations to specifiers as alternative options for them to resolve. These behavior templates can be used to create the four original behaviors and many more. Each behavior template uses either one or two parameters, and those parameters are named A and B. For simplicity, our approach puts one restriction on the behavior templates that refer to both A and B: the specifier-defined events that are associated with those two parameters must be distinct.
There are options associated with each of the behavior templates except for Absence, and these associations are shown in Table 3.2. For each option, this table gives the option’s name, shows a brief description of how the option can vary, and uses check-marks to denote which behavior templates contain that option. Each option is a binary choice. For example, the Nullity option determines whether or not A is required to occur at least once in a given sequence of events, and both the Response and Precedence behavior templates contain this option. Although each option is a binary choice when it is examined in isolation, some options cannot be resolved in isolation. The possible combinations of two of the options, Finalization and Repeatability, are constrained, and Appendix A.2.1.1 discusses the constraints in detail. In addition to the constraints on those two options, the semantics of another option, Immediacy, relies on further decisions regarding what the property’s secondary events are.

Returning to Table 3.2, the Existence behavior template has one option, Bounded, which leads to two distinct variations on the concept of Existence. The Response behavior template has seven options, which can be combined to produce a total of 96 distinct variations on the concept of Response. The Precedence behavior template shares six of those options with the Response behavior template, and those six options can be combined to produce a total of 48 distinct variations on the concept of Precedence. The Precedency option, which is in the Response behavior template, is not included in the Precedence behavior template because one of that option’s settings directly conflicts with the core concept of the Precedence behavior. The other setting of that option allows PROPEL specifiers to create a two-event behavior that encompasses both the core concept of the Response behavior and the core concept of the Precedence behavior. Chapter 7 discusses an alternative approach to unifying the Response and Precedence behavior templates, and that approach relies on a generalized two-event behavior template that supports the current Response and Precedence behavior templates and also expands the set of supported behaviors to include additional two-event behaviors, such as the concept of alternation. In the following sections, however, we focus on describing the three property views for the current set of four behavior templates, where Response and Precedence are treated separately.

### 3.1.2 Behavior Question Tree

#### 3.1.2.1 QT Notation

The QT is a property view that was designed to provide guidance to PROPEL specifiers for how to select a scope or behavior template, so that specifiers would know how to get started with the elucidation of their intended properties. This is not the only purpose for the QT property view, however. Like the other property views, the QT can be used to resolve the options that are associated with the selected property pattern template. Thus, the same options that must be resolved in the other property views are options in the QT property view. In this property view, PROPEL specifiers are presented with questions and a choice of possible answers.
How many events of primary interest are there in this behavior?

One event

Which of the following choices best describes the restriction on A?

- A is never allowed to occur (Absence)
- A is required to occur at least once (Existence, no max bound)
- A is required to occur exactly once (Existence, max bound of 1)

Two events

Which of the following choices best describes how A and B interact?

- If A occurs, B is required to occur subsequently (Response)
- B is not allowed to occur until after A occurs (Precedence)

Both statements describe how A and B interact: if A occurs, B is required to occur subsequently, and B is not allowed to occur until after A occurs (Response, with Precedence option enforced)

Figure 3.1. Behavior Template Organization in the BQT

to select from. The QT question- and answer-text is similar to the corresponding DNL template text, which is described in Section 3.1.4.2. Specifiers can select only one answer for each question and, based on which answer is selected, new questions and their associated answers may then be presented for further consideration. In keeping with the hierarchical nature of the QT property view, we refer to each new question that can be revealed after answering a given question as a child question of that previous question, and we refer to the previous question as the parent question of the child questions. When an answer to a question is selected and this selection reveals a new set of child questions, then for as long as that answer remains selected, all those child questions and their associated answers remain visible and can be revisited if necessary. By selecting a different answer to a parent question, specifiers will explore a different set of child questions that are relevant to the new answer and specifiers can thus arrive at a different property altogether. The QT property view hides all questions that are not relevant to the currently selected answers, allowing specifiers to focus on one set of concerns at a time. The root question of a QT property view and its associated answers are always visible. The QT property view is structured such that the questions that lead to the choice of a behavior template must be answered first, and in a predefined order, before questions that resolve the options associated with that template. If the questions about the options in a behavior template are conceptually orthogonal to each other, they are represented as sibling questions that can be answered in any order. After answering all of the questions in the BQT, and after associating specifier-defined events with the parameters in the underlying behavior template, PROPEL specifiers are left with a fully-instantiated behavior.

Figure 3.1 shows how the Behavior QT (BQT) organizes the set of available behavior templates. In this figure, each tree node is either a question or an answer. Answers to a question are indented under the question and child questions are indented under the answers to their parent question. The BQT’s guidance for choosing...
among the four behavior templates is structured by three questions. The root question asks how many events of primary interest there are in the intended behavior and provides two answers, either one event or two events. Specifiers can select just one of those two answers. If there is only one event of primary interest, a child question asks how the occurrence of that event is restricted and provides three answers. Alternatively, if there are two events of primary interest, a child question asks how their interactions are restricted and provides three answers. Selecting an answer to the root question and an answer to one of these two child questions selects a behavior template. Once a behavior template is selected, specifiers can continue to make decisions about the behavior template’s options, if any exist, using any or all of the three property views that support behavior templates: the BQT, the FSA template property view, or the DNL template property view.

It should be noted that Figure 3.1 differs from the actual BQT in two ways. One difference is that this figure shows descriptions next to each of the behavior templates’ choices, but the actual BQT does not. The variations that are shown for the Existence and Response behavior templates in this figure are each discussed in Appendix A.1.1. The other way that this figure differs from the actual BQT is that in the actual BQT it would not be possible for specifiers to expand both the “One event” and “Two events” answers to the root question simultaneously. A complete, fully-expanded version of the BQT is available in Appendix A.1.1. Chapter 7 has a discussion about other ways of organizing the behavior templates, such as replacing the separate Response and Precedence answers with a more unified series of questions and answers about a more general 2-event behavior.

3.1.2.2 An Example of Instantiating a Property’s Behavior in the BQT

In this section, we give an example of instantiating a property’s behavior using only the BQT. Later sections present this same example for each of property views. Consider the following property specification, which is based on a property from the Blood Transfusion case study (see Chapter 5):

If the nurse discovers that the patient’s type and screen (T&S) are not available in the lab, the nurse must obtain a blood specimen from the patient.

Figure 3.2 shows the BQT view of a partially-resolved behavior for this example property. The highlighted nodes in the BQT are answers that specifiers have either selected in response to the questions, or that have been automatically selected when specifiers resolved the associated options using one of the other property views. Unselected answers are not highlighted. Child questions are only visible when specifiers select the appropriate answer to the parent questions.

As can be seen in Figure 3.2, the first question the BQT asks specifiers about the intended property is, “How many events of primary interest are there in this behavior?” Let us assume that specifiers identify two
Figure 3.2. Choosing a Behavior Template in the Behavior Question Tree

events that are in the intended property’s alphabet, no-T&S and obtain-specimen, and that specifiers make no-T&S the first primary event and obtain-specimen the second primary event.\textsuperscript{4} For this example, specifiers would thus select the answer “Two events.” Since the four child questions under the “Two events” answer are orthogonal to each other, they can be answered in any order. In this discussion, we answer them in the order that they are shown in the BQT.

The first of the four child questions is about the nature of the interaction between the two events. The BQT offers three possible answers. To make a decision about which of these three answers to choose, specifiers must consider how the nurse interacts with the lab. If the lab does not have a T&S for the patient, the nurse must obtain a blood specimen from the patient and send it to the lab, so that the lab can get a T&S. Given this, specifiers can eliminate the second answer from consideration, because that answer is the only one that does not require obtain-specimen to follow an occurrence of no-T&S. To decide between the first and third answers, which are the two possible settings of the Precedency option in the Response behavior template, specifiers need to know whether obtain-specimen is allowed to occur before the first occurrence of no-T&S.

\textsuperscript{4}This association of event names with the parameters is done via the Alphabet View (see Section 4.3), but that dialog is not shown here. After this association, PROPEL substitutes the event names in place of the parameters’ default names, as shown by the boldface words in Figure 3.2.
Since there are many reasons to obtain a blood specimen besides just discovering that no T&S is available, obtaining a blood specimen is allowed to occur before discovering that no T&S is available. Knowing this eliminates the third answer, and as is shown in Figure 3.2, specifiers thus select the first answer about how the two events interact.

Now that specifiers have selected the first answer, the BQT reveals a child question under that answer. This child question is concerned with the Nullity option, which determines whether no-T&S is required to occur at least once, independent of the circumstances. There are two possible answers to this question in the BQT. To make a decision about which of these answers to choose, specifiers must consider whether there are reasonable circumstances in which no-T&S is allowed to not occur. If there are such circumstances, the second answer should be chosen. For example, since the lab may have a T&S for the patient already available when the nurse inquires, this is a reasonable circumstance in which no-T&S does not occur. Knowing this eliminates the first answer and, as is shown in Figure 3.3, specifiers thus select the second answer to that question: no-T&S is not required to occur.

The second child question under the “Two events” answer in the BQT is concerned with the Pre-arity option, which determines whether, after the first no-T&S has occurred, no-T&S is allowed to occur again.
before the first subsequent obtain-specimen occurs. There are two possible answers to this question in the BQT. To make a decision about which of these answers to choose, specifiers must consider how much a nurse can interact with the lab after discovering that the lab does not have a T&S for the patient. Since the nurse can repeatedly check with the lab to find out if the patient’s T&S is available before obtaining a blood specimen, there exist reasonable circumstances in which no-T&S is allowed to occur zero or more times before the first subsequent obtain-specimen occurs. Knowing this eliminates the second answer, and as is shown in Figure 3.4, specifiers thus select the first answer to this question.

The third child question under the “Two events” answer in the BQT is concerned with the Finalization option, which determines whether, after no-T&S and the first subsequent obtain-specimen occur, no-T&S is allowed to occur again. There are two possible answer to this question in the BQT. To make a decision about which of these answers to choose, specifiers must consider how a nurse interacts with the lab after sending a blood specimen to them. In this situation, a nurse can check repeatedly with the lab to find out if the patient’s T&S is available yet. Knowing this, specifiers select the “Yes, no-T&S is allowed to occur again” answer to resolve that option. As is shown in Figure 3.5, upon selecting that answer, a new child question “Is obtain-specimen allowed to occur again?” appears, along with three possible answers.
Which of the following choices best describes how no-T&S and obtain-specimen interact?

- Both statements describe how no-T&S and obtain-specimen interact: if no-T&S occurs, obtain-specimen is required to occur subsequently, and obtain-specimen is not allowed to occur until after no-T&S occurs.

After no-T&S occurs, is no-T&S allowed to occur again before the first subsequent obtain-specimen occurs?

- Yes, no-T&S is allowed to occur again, zero or more times, before the first subsequent obtain-specimen occurs.
- No, no-T&S is not allowed to occur again before the first subsequent obtain-specimen occurs.

After no-T&S and the first subsequent obtain-specimen occur, is no-T&S allowed to occur again?

- Yes, no-T&S is allowed to occur again, but not until after another no-T&S occurs. If another no-T&S does occur, the situation is the same as when the first no-T&S occurred, meaning that the restrictions described on any events that take place after that occurrence would again apply.
- Yes, obtain-specimen is allowed to occur again, zero or more times, whether or not another no-T&S occurs.
- If another no-T&S does occur, is the situation the same as when the first no-T&S occurred, meaning that the restrictions described on any events that take place after that occurrence would again apply?
  - Yes, if no-T&S does occur again, the restrictions described on those events would again apply.
  - No, even if no-T&S does occur again, there are no restrictions on the occurrences of any future events.
- No, obtain-specimen is not allowed to occur again.
- No, no-T&S is not allowed to occur again.

Are there any events of secondary interest in this behavior?

- Yes, there is an event of secondary interest in this behavior: it is $C_1$.
- No, there are no events of secondary interest in this behavior.

**Figure 3.5. Resolving Three Options Together in the Behavior Question Tree**

This new child question is concerned with two options, *Post-arity* and *Repeatability*. Along with the *Finalization* option covered in the last paragraph, these two options determine what is allowed to happen after no-T&S and the first subsequent obtain-specimen occur. *Post-arity* determines whether obtain-specimen is allowed to occur again in this situation, and *Repeatability* determines whether, if no-T&S occurs again in this situation, all the restrictions described thus far in this behavior must again apply. To make a decision about the three possible answers that the BQT has shown in Figure 3.5, the specifiers must again consider...
how a nurse interacts with the lab after sending a blood specimen to them. A nurse may be asked to obtain blood specimens for purposes other than a T&S, so after obtaining the initial blood specimen for the T&S, the nurse is allowed to obtain other blood specimens. As is shown in Figure 3.5, specifiers thus select the second of the three possible answers, “Yes, obtain-specimen is allowed to occur again, zero or more times, whether or not another no-T&S occurs.” Selecting this particular answer resolves the Pre-arity option, but it does not resolve the Repeatability option. To resolve this option, specifiers must answer the new child question that the BQT reveals. As is shown in Figure 3.5, the BQT provides two possible answers to the new child question. To decide which of these answers to choose, specifiers must consider whether there are reasonable circumstances in which a nurse would again discover that the lab does not have a T&S for the patient, and even knowing this, the nurse does not obtain another blood specimen. There are such circumstances: after obtaining a blood specimen and sending it to the lab, a nurse would normally check with the lab until the T&S is available, and seeing that it is not available in the interim would not necessarily mean that the nurse must obtain more blood specimens from the patient. It would only mean that the nurse must wait until the lab finishes performing the T&S. Knowing this, specifiers thus select the second answer.

There is one option left to resolve in the Response behavior template, Immediacy. Recall that specifiers can put secondary events into the property’s alphabet. The fourth child question under the “Two events” answer in Figure 3.6 asks whether there are any secondary events in this behavior. There is a secondary event in this situation: it is important that a nurse never labels a specimen vial before putting a blood specimen in it, because labels are difficult to affix smoothly to the specimen vials, and if there are any wrinkles or folds in the applied label, the sticky part of the label could catch on the inside of the syringe and become a problem when the nurse tries to extract it. Given this information, specifiers add label-vial to the property’s alphabet, using the Alphabet Views described in Section 4.3, and select the “Yes” answer to the question of whether there are secondary events. Upon selecting this answer, the BQT reveals a final child question, which is concerned with the Immediacy option, and two possible answers. The Immediacy option determines whether label-vial is prohibited from occurring between an occurrence of no-T&S and the first subsequent occurrence of obtain-specimen. As discussed previously, label-vial should not occur before obtain-specimen, and thus it is not allowed to occur between no-T&S and obtain-specimen.\(^5\)

At this point, all the options in the behavior template have been resolved, as indicated by Figure 3.6, and thus a Response behavior has been selected. Throughout the elucidation of this property, specifiers can revisit

---

\(^5\)A separate property specification would be needed to prevent label-vial from occurring before obtain-specimen occurs in other situations, since our approach limits each property to specifying a very focused subset of system behavior.
How many events of primary interest are there in this behavior?
- One event
- Two events

Which of the following choices best describes how **no-T&S** and **obtain-specimen** interact?
- If **no-T&S** occurs, **obtain-specimen** is required to occur subsequently
  - Is **no-T&S** required to occur?
    - Yes, **no-T&S** is required to occur
    - No, **no-T&S** is not required to occur
  - **obtain-specimen** is not allowed to occur until after **no-T&S** occurs

Both statements describe how **no-T&S** and **obtain-specimen** interact: if **no-T&S** occurs,
**obtain-specimen** is required to occur subsequently, and **obtain-specimen** is not allowed to
occur until after **no-T&S** occurs

After **no-T&S** occurs, is **no-T&S** allowed to occur again before the first subsequent **obtain-specimen** occurs?
- Yes, **no-T&S** is allowed to occur again, zero or more times, before the first subsequent **obtain-specimen** occurs
- No, **no-T&S** is not allowed to occur again before the first subsequent **obtain-specimen** occurs

After **no-T&S** and the first subsequent **obtain-specimen** occur, is **no-T&S** allowed to occur again?
- Yes, **no-T&S** is allowed to occur again
  - Is **obtain-specimen** allowed to occur again?
    - Yes, **obtain-specimen** is allowed to occur again, but not until after another **no-T&S** occurs. If another **no-T&S** does occur, the situation is the same as when the first **no-T&S** occurred, meaning that the restrictions described on any events that take place after that occurrence would again apply
    - Yes, **obtain-specimen** is allowed to occur again, zero or more times, whether or not another **no-T&S** occurs
      - If another **no-T&S** does occur, is the situation the same as when the first **no-T&S** occurred, meaning that the restrictions described on any events that take place after that occurrence would again apply?
        - Yes, if **no-T&S** does occur again, the restrictions described on those events would again apply
        - No, even if **no-T&S** does occur again, there are no restrictions on the occurrences of any future events
    - No, **obtain-specimen** is not allowed to occur again
- No, **no-T&S** is not allowed to occur again

Are there any events of secondary interest in this behavior?
- Yes, there is an event of secondary interest in this behavior: it is **label-vial**
  - After **no-T&S** occurs, is the event of secondary interest in this behavior allowed to occur before the first subsequent **obtain-specimen** occurs?
    - Yes, after **no-T&S** occurs, the event of secondary interest in this behavior is allowed to occur zero or more times before the first subsequent **obtain-specimen** occurs
    - No, after **no-T&S** occurs, the event of secondary interest in this behavior is not allowed to occur before the first subsequent **obtain-specimen** occurs
  - No, there are no events of secondary interest in this behavior

---

Figure 3.6. Example Fully-Resolved Behavior in the BQT
any decisions made previously and can change their answers. Specifiers can also resolve these options in a
different order than the order given in this example.

3.1.3 Behavior Finite-State Automata Templates

The FSA template property view was chosen to provide both mathematically well-defined semantics and
a graphical representation for the properties supported in PROPEL. Like the other property views, the FSA
templates can help specifiers to resolve the options that are associated with the selected property pattern
template. Thus, the same options that must be resolved in the QT, ST, and DNL template property views are
options in the FSA template property view as well.

3.1.3.1 FSA Template Notation

A traditional FSA view of a property is defined by the tuple \(\langle S, s, A, \Sigma, \delta \rangle\), where \(S\) is the finite set of
states, \(s \in S\) is the unique start state, \(A \subseteq S\) is the set of accepting states, \(\Sigma\) is the alphabet of events, and
\(\delta : S \times \Sigma \rightarrow S\) is a transition function. A sequence \(e_1, e_2, \ldots, e_n \in \Sigma^*\) is accepted by the FSA if a sequence of
states \(s_0, s_1, \ldots, s_n\) exists in \(S\) such that:

1. \(s_0 = s\),
2. \(s_n \in A\), and
3. \(\delta(s_i, e_{i+1}) = s_{i+1}\) for \(i \in \{0, \ldots, n-1\}\).

Traditionally, when depicting an FSA graphically, states are shown as circles, the start state is denoted
by an incoming arrowhead on the circle, accepting states are denoted by inner concentric circles, and the
transitions are denoted by arrows between states, indicating the direction of flow in the FSA. Each transition
is labeled by one or more events from the alphabet \(\Sigma\). The FSA template notation extends the traditional FSA
notation with the following additions:

- optionally-accepting states,
- optional transitions,
- multi-labels,
- \(\ldots\), the wildcard character, representing all of \(\Sigma\), and
- \(\neg\), the set-complement operator.

As can be seen in Figure 3.7, an optionally-accepting state is denoted by a dashed inner concentric circle.
An optionally-accepting state resolves either to an accepting state or to a non-accepting state. An optional
transition is denoted by an arrow with a dashed line. Every transition in an FSA template has an associated multi-label, which is a list of alternative multi-label items, where each multi-label item is separated from the others in the list by the word “or.” A multi-label item can be an individual label or a compound label. An individual label is either a parameter’s default name or the specifier-defined event from $\Sigma$ that is associated with that parameter, if there is such an event. A compound label consists of either the set of all events in $\Sigma$, a set of individual labels, or the complement of a set of individual labels. There are two shorthand notations that are used in the multi-label items: the “.” wildcard character indicates the set of all events in $\Sigma$ and the “¬” operator indicates the complement of the given set of events with respect to $\Sigma$. A multi-label on an optional transition resolves to at most one of its multi-label items and if there is exactly one multi-label item left, the optional transition resolves to a regular transition. The optional transition disappears if all of the items in its multi-label have been eliminated from consideration. A multi-label can also be associated with a transition that is required by the property pattern template, and a multi-label on such a transition resolves to exactly one of the multi-label items. After resolving all of the optional components in an FSA template and associating specifier-defined events with the parameters in the underlying property pattern template, PROPEL specifiers are left with a fully-instantiated FSA property, which is required to be total and deterministic.\textsuperscript{6}

3.1.3.2 An Example of Instantiating a Property’s Behavior FSA Template

Figure 3.7 shows the FSA template property view of a partially-resolved behavior for the example Blood Transfusion property given Section 3.1.2.2. Before PROPEL can show this FSA template view to specifiers, however, the Response behavior template must be selected in the BQT, as is shown in Figure 3.2. Given

\textsuperscript{6}For brevity, we do not show the transitions that go to a non-accepting trap state. When no transition is provided that explicitly allows an event to occur, it should be assumed that an occurrence of that event puts the FSA into a non-accepting trap state.
the structure of the BQT, when specifiers select the Response behavior template, the Precedency option is resolved along with that selection. Outside of the BQT, however, that option is not required to be tied to the selection of the behavior template. In the FSA template view shown in Figure 3.7, the Precedency option can be resolved separately, by selecting one of the two multi-label items on the start state’s self-loop. For comparison, the same choice that was made in the BQT to resolve the Precedency option is highlighted in the menu shown in Figure 3.7. Thus, similar to the BQT in Figure 3.2, the resulting FSA template does not prohibit obtain-specimen from occurring before the first occurrence of no-T&S.

Continuing to resolve the options in the same order that is given in the BQT example in Section 3.1.2.2, the next option to resolve is Nullity. In the FSA template property view, that option is determined by the accepting status of the start state. If the start state is accepting, then no-T&S is not required to occur, but if the start state is not accepting, then no-T&S is required to occur. In the FSA template view shown in Figure 3.8, the Nullity option can be resolved by selecting the start state’s accepting status in the menu shown in that figure. For comparison, the same choice that was made in the BQT in Figure 3.3 is highlighted in the menu in Figure 3.8, and the resulting FSA template, which is shown in Figure 3.9, does not require no-T&S to occur at least once.

Although the BQT and FSA template property views allow specifiers to resolve the Precedency and Nullity options independently, those two property views group the remaining five options in the Response behavior template differently. For example, the Pre-arity and Immediacy options in the BQT must be resolved independently of each other, whereas in the FSA template those two options are tied together. The self-loop on the second state in the Response FSA template in Figure 3.9 determines the settings of both of these options simultaneously. This self-loop can be left in one of four possible settings, one for each of the four combinations of these two options: either the self-loop exists and is labeled with one of the three multi-label
items available, or the self-loop is removed entirely. Recall that the \( \neg \) is a set-complement operator and that in the BQT example in Section 3.1.2.2 specifiers added a secondary event to the property’s alphabet, \textit{label-vial}. Thus, to make the same decisions about \textit{Pre-arity} and \textit{Immediacy} in the FSA template view that were made in the BQT examples shown in Figures 3.4 and 3.6, respectively, specifiers would select the \textit{no-T&S} multi-label item, as is shown highlighted in the menu in Figure 3.9. This selection results in the self-loop on the second state existing and being labeled by \textit{no-T&S}, as is shown in Figure 3.10. This means that, similar to the BQT in Figure 3.6, \textit{no-T&S} would be allowed to occur zero or more times before \textit{obtain-specimen} occurs, and \textit{label-vial} would not be allowed to occur after an occurrence of \textit{no-T&S} and before the first subsequent occurrence of \textit{obtain-specimen}.

\begin{figure}[h]
\centering
\includegraphics[width=\textwidth]{figure3.9.png}
\caption{Resolving Two Options Together in the FSA Template}
\end{figure}

\begin{figure}[h]
\centering
\includegraphics[width=\textwidth]{figure3.10.png}
\caption{Resolving Three Options Together in the FSA Template}
\end{figure}
The remaining three options that must be resolved are *Finalization*, *Post-arity*, and *Repeatability*, and in the FSA template property view, these three options are closely tied together because they are collectively represented by the two outgoing transitions on the third state. Since the FSAs that can be created using this FSA template view are required to be total and deterministic, *no-T&S* can only occur on one of the third state’s outgoing transitions. If specifiers make a decision about the existence of the transition from the third state to the second state, that decision limits which multi-label items can be selected for the self-loop on the third state, and vice versa. The decisions made about these two transitions affect all three options simultaneously. To make the same decisions about these three options that were made in the BQT in Figure 3.5, specifiers would select the “.” multi-label item on the third state’s self-loop, as is shown highlighted in the menu in Figure 3.10. As can be seen in Figure 3.11, this selection results in the self-loop on the third state being labeled by “.” and the transition from the third state to the second state being removed from the FSA. This means that all the events in the alphabet of the property (i.e., *no-T&S*, *obtain-specimen*, and *label-vial*) are allowed to occur without restriction in the third state. This completed behavior FSA is the formal view that corresponds to the answers chosen in the completed BQT property view in Figure 3.6.

### 3.1.4 Behavior Disciplined Natural Language Templates

The DNL template property view is based on a restricted subset of natural language that is meant to be domain-independent. This property view was chosen for three reasons. One reason is that this it is more readily understandable than property specification formalisms, which require significant expertise to use. In practice, developers tend to write requirements in natural language and thus we expect that providing a property view in PROPEL that is based on natural language will aid in making a PROPEL property specification more understandable to a wide audience. Another reason that this property view was chosen is that, unlike the others, it is customizable. A single concept can sometimes be expressed in different but synonymous ways in natural language and, where clarity is preserved, the DNL template property view can support the use of such synonyms. The synonyms are available so that PROPEL specifiers can adjust the DNL to use whatever phrasing seems most natural for their intended property. The other reason that the DNL property view was
chosen is that it is written in a grammatically-correct and readable paragraph form and can thus be easily used in the creation of requirements documents.

3.1.4.1 DNL Template Notation

A DNL template property view for a behavior template consists of the following numbered phrases, in the given order:

1. A statement about the what the primary event(s) in the behavior template are. In a one-event behavior template, the statement is of the form:
   
   “The event of primary interest in this behavior is A.”

   and in a two-event behavior template, the statement is of the form:

   “The events of primary interest in this behavior are A and B.”

2. A statement about the secondary event(s) in the behavior template. If there are one or more secondary events, the statement is of the form:

   “The {event / events} of secondary interest in this behavior {is C₁ / are C₁, C₂, …, Cₙ₋₁, and Cₙ}.”

   where the “{ / }” notation used here denotes how the phrasing of the natural language text changes based on how many secondary events are in the behavior. If there is just one secondary event, then the left side of the “{ / }” is displayed. If there is more than one secondary event, then the right side of the “{ / }” is displayed. If there are n secondary events, where n can be any positive integer, then the specifier-defined names of the secondary events are displayed in alphabetical order in a list of length n. If there are no secondary events, the statement is of the form:

   “There are no events of secondary interest in this behavior.”

3. The Core phrase, which describes the core concept of the behavior template.

4. Zero or more subsidiary phrases, which describe the various options and their possible settings in the behavior template.

We now look in more detail at how these phrases are represented in the DNL template property view for an example behavior template.
3.1.4.2 An Example of Instantiating a Property’s Behavior DNL Template

The DNL template shown in Figure 3.12 is a representation of the example Blood Transfusion property that was used in Sections 3.1.2.2 and 3.1.3.2. Phrase 1 in this Response DNL template describes the two primary events that were identified in Section 3.1.2.2 and phrase 2 describes the secondary event also identified in that section. Phrase 3 expresses the core concept of the Response behavior. Phrases 4-7 are the subsidiary phrases for the Response DNL template, and they express the settings for each of the Response behavior template’s options.

Going in the order that the phrases are shown in Figure 3.12, phrase 4 expresses the setting for the Precedency option. Given the structure of the BQT, when specifiers select the Response behavior template, the Precedency option is resolved along with that selection. Outside of the BQT, however, that option is not required to be tied to the selection of the behavior template. Just as is shown in the FSA template property view in Figure 3.7, Figure 3.12 shows how the Precedency option can be resolved separately in the DNL template, by selecting one of the choices in the combo box for phrase 4. For comparison, the same choice that was selected in the BQT in Figure 3.2 and in the FSA template in Figure 3.7 is highlighted in the combo box for phrase 4 in Figure 3.12.

Continuing to resolve the options in the same order that is given in Figure 3.12, the next option to resolve is Nullity. In the DNL template property view, that option is determined by the setting of phrase 5. As is shown in the DNL template property view in Figure 3.13, the Nullity option can be resolved by selecting one
of the choices in the combo box for that phrase. For comparison, the same choice that was selected in the BQT in Figure 3.3 and in the FSA template in Figure 3.8 is highlighted in that combo box.
BEHAVIOR:

1. The events of primary interest in this behavior are no-T&S and obtain-specimen.
2. The event of secondary interest in this behavior is label-vial.
3. If no-T&S occurs, obtain-specimen is required to occur subsequently.
4. Before the first no-T&S occurs:
   - obtain-specimen is allowed to occur zero or more times;
   - label-vial is allowed to occur zero or more times.
5. no-T&S is not required to occur.
6. After no-T&S occurs, but before the first subsequent obtain-specimen occurs:
   - no-T&S is allowed to occur again, zero or more times;
   - label-vial is not allowed to occur.
7. After no-T&S and the first subsequent obtain-specimen occur:
   - obtain-specimen is allowed to occur again, zero or more times, before another no-T&S occurs;
   - label-vial is allowed to occur zero or more times;
   - no-T&S is allowed to occur again and, if it does, then the situation is the same as when the first no-T&S occurred, meaning that the restrictions described in parts 3, 6, and 7 would again apply.
   - obtain-specimen is not allowed to occur again until after another no-T&S occurs;
   - label-vial is allowed to occur zero or more times;
   - no-T&S is allowed to occur again and, if it does, then the situation is the same as when the first no-T&S occurred, meaning that the restrictions described in parts 3, 6, and 7 would again apply.
   - Both no-T&S and obtain-specimen are allowed to occur again, zero or more times, and there are no restrictions on the occurrences of any future events;
   - label-vial is allowed to occur zero or more times.
   - no-T&S is allowed to occur again, zero or more times;
   - obtain-specimen is never allowed to occur again;
   - Further occurrences of no-T&S do not impose additional restrictions on the occurrences of any future events;
   - label-vial is allowed to occur zero or more times.
   - obtain-specimen is allowed to occur again, zero or more times;
   - no-T&S is never allowed to occur again;
   - Further occurrences of obtain-specimen do not impose additional restrictions on the occurrences of any future events;
   - label-vial is allowed to occur zero or more times.
   - neither no-T&S nor obtain-specimen are allowed to occur again;
   - label-vial is allowed to occur zero or more times.

Figure 3.15. Resolving Three Options Together in the DNL Template

The DNL template property view groups the remaining five options in the Response behavior template in the same way that the FSA template property view does. For example, Pre-arity and Immediacy are tied together in one phrase in the DNL template property view, just as they are tied together in one optional transition in the FSA template property view. As is shown in Figure 3.14, phrase 6 in the Response DNL template requires specifiers to resolve these two options simultaneously. Since there are two options with two settings apiece involved in this phrase, there are four choices available in the combo box for this phrase. There are two bullet points in each choice: the first point is a setting for the Pre-arity option and the second
**BEHAVIOR:**

1. The events of primary interest in this behavior are no-T&S and obtain-specimen.
2. The event of secondary interest in this behavior is label-vial.
3. If no-T&S occurs, obtain-specimen is required to occur subsequently.
4. Before the first no-T&S occurs:
   - obtain-specimen is allowed to occur zero or more times;
   - label-vial is allowed to occur zero or more times.
5. no-T&S is not required to occur.
6. After no-T&S occurs, but before the first subsequent obtain-specimen occurs:
   - no-T&S is allowed to occur again, zero or more times;
   - label-vial is not allowed to occur.
7. After no-T&S and the first subsequent obtain-specimen occur:
   - Both no-T&S and obtain-specimen are allowed to occur again, zero or more times, and there are no restrictions on the occurrences of any future events;
   - label-vial is allowed to occur zero or more times.

**Figure 3.16. Example Behavior DNL Specification**

point is a setting for the *Immediacy* option. By selecting the highlighted third choice in the combo box in phrase 6, specifiers can make the same decision about *Pre-arity* that was made in both the BQT in Figure 3.4 and in the FSA template in Figure 3.9. In addition, since *label-vial* is considered a secondary event (as is shown in phrase 2), that same combo box choice allows specifiers to put the same *Immediacy* restriction on *label-vial* that is made in both the BQT shown in Figures 3.6 and the FSA template shown in Figure 3.9.

The remaining three options that must be resolved are *Finalization*, *Post-arity*, and *Repeatability*, and in the DNL template property view, similar to the FSA template property view, these three options are closely tied together. As is shown in Figure 3.15, phrase 7 in the Response DNL template requires specifiers to resolve these three options simultaneously. To make the same decision about these three options that were made in the BQT in Figure 3.5 and in the FSA template in Figure 3.10, specifiers would select the answer that is highlighted in the combo box for phrase 7 in Figure 3.15. As can be seen in Figure 3.16, the resulting completed behavior DNL description is a readable, natural-language property view that corresponds to the answers chosen in the completed BQT in Figure 3.6 and the completed FSA in Figure 3.11.

### 3.1.5 Using All Three Views to Instantiate a Property’s Behavior

Although the previous sections illustrated the three property views by elucidating an example behavior using only one property view at a time, in PROPEL specifiers can work with one or more of the property views and can change back and forth between them as the property is being elucidated. As was discussed previously, specifiers must start elucidating the behavior by selecting a behavior template in the BQT property...
view, because neither the FSA nor DNL template property views provide support for selecting among the behavior templates. Once a behavior template has been selected, however, specifiers can resolve the options using any or all of those three property views. As specifiers resolve the options, PROPEL automatically keeps these property views synchronized.

For example, consider one possible series of figures that illustrates what specifiers might encounter if they choose to work with all three property views. To elucidate the behavior for the example property that has been presented in the previous sections, specifiers would first select a behavior template, as is shown in the BQT in Figure 3.2. When specifiers select the Response behavior template in the BQT, they also resolve the Precedency option. Specifiers could continue to resolve the options in that behavior template using the BQT, but let us assume that they switch to using one of the other property views, such as the FSA template shown in Figure 3.8. In that figure, the Precedency option is already resolved to match the specifiers’ previous selection in the BQT, and now the specifiers are resolving the Nullity option. Let us assume that the specifiers choose to continue using this property view to resolve the Pre-arity and Immediacy options, as is shown in Figure 3.9. Specifiers could continue to resolve the remaining three options in the Response behavior template using the FSA template property view, but let us assume that they switch to using the DNL template property view. At this point in the behavior elucidation, PROPEL would present what is shown in Figure 3.15 to the specifiers, and they would resolve the final three options. Now that the specifiers have resolved all the options in the Response behavior template, they are left with the BQT shown in Figure 3.6, the FSA shown in Figure 3.11, and the DNL text shown in Figure 3.16. The selection of the behavior template and the settings of any option can be revisited and changed whenever specifiers need to do so. Specifiers can resolve the options in a different order than the one described above, and can make different decisions about when to use these three property views while resolving those options.

3.1.6 Behavior Summary

Our approach supports three different views that enable specifiers to elucidate their desired property’s behavior: the BQT view, the FSA template view, and the DNL template view. The BQT view was designed to provide specifiers with guidance for how to select the appropriate behavior template, something which neither of the other views do. The FSA template view was designed to assist specifiers in creating a graphical, formal view of their desired behavior, in this case an FSA. Since the fully-resolved form of this view is mathematically precise, this view provides the rigor necessary to allow the resulting property specification to be used as an input for many types of automated analyses. The DNL template view was designed to assist specifiers in creating an understandable representation of their behavior. This view does not require specifiers to have expertise in a particular property specification formalism, since the DNL is intended to appeal to those
Specifiers who prefer a natural language specification of their behavior. Specifiers can work with any of the views to resolve each of the options associated with the selected behavior template. Thus, the same options that must be resolved in one view are options in the other views as well. After resolving the options using any view, specifiers can revisit their decisions at any time. A fully-resolved behavior is represented with several DNL paragraphs, one FSA, and one fully-resolved BQT. The relationship between the choices in a behavior template’s DNL template view and those in the associated FSA template and BQT views is not necessarily one-to-one, however, since some choices in the DNL template view affect the associated FSA template and BQT views in more than one location. This is because the DNL template view sometimes displays a group of options settings together while the FSA template and BQT views display the settings of each of those options individually. Thus, the DNL template view does not reflect changes made in the other two views until all the options in the group have been resolved. The behavior example given in this section showed only how the Response behavior template is represented in the three views. A detailed description of the other three behavior templates and how they are represented in all three views is included in Appendix A.

3.2 Specifying a Scope

In this section, we discuss the concept of a property’s scope and how it is applied to a behavior. In our approach, a property’s scope defines the restricted intervals within which the property’s behavior is required to hold. A restricted interval is defined in terms of a starting delimiter, an ending delimiter, or both. For example, a restricted interval might begin with the first occurrence of the starting delimiter and end with the first subsequent occurrence of the ending delimiter. PROPEL currently supports four scope templates, which are based on the set of scopes proposed by Dwyer et al. [81] in the property pattern approach. As was true with the behaviors, Dwyer et al. expected users to have considerable expertise with the specification formalisms to be able to tailor those scopes as needed. Our scope templates extend the property pattern scopes in a way similar to how the behavior templates extend the property pattern behaviors. Specifically, we explicitly provide a number of possible variations in the scopes and present these alternative options to specifiers via the scope templates, using the same three property views that were described previously for the behavior templates. These four scope templates and the ways they are represented in the three property views and in an additional Scope Timeline (ST) scope view are discussed in the remainder of this section.

3.2.1 Scope Templates

The four scope templates supported by PROPEL extend the original Dwyer et al. property pattern scopes by making a number of possible variations in those scopes explicit and by presenting alternative options to specifiers for them to resolve. These four scope templates, with their options, can be used to create the five
original property pattern scopes and many more. The scope templates use two parameters, named **START** and **END**, for the starting and ending delimiters, respectively. Our approach puts a restriction on the options that can be in the scope templates: scopes cannot cause a property to be violated. Thus, options that are concerned with various ways in which occurrences of the scope delimiters, or the lack thereof, can cause a property violation are prohibited. We chose not to include these options in the scope templates at this time, to more clearly separate the concepts of scope and behavior. With this restriction, the only part of a property that can cause a violation is the behavior, and thus the property’s scope is only involved in defining when the behavior is required to hold. A separate property specification would be needed to impose restrictions on the events that are the scope delimiters, since our approach limits each property specification to restricting a very focused subset of system behavior. In scope templates that do not have the **START** parameter, the beginning of the event sequence is the default starting delimiter. Likewise, in scope templates that do not have the **END** parameter, the end of the event sequence is the default ending delimiter. Our approach puts two restrictions on the starting and ending delimiters. The first is that the starting and ending delimiters cannot be in the alphabet of the behavior. We make this restriction because some undesirable interactions could otherwise occur. For example, specifiers would be able to create a property with a Response behavior whose stimulus event is the same as the scope ending delimiter, and thus no event sequence could ever satisfy this property. The second restriction is that the starting and ending delimiters must be distinct events. This restriction also avoids some complications, but it does not allow some forms of two-event alternation. Relaxing these restrictions requires careful consideration and is discussed in more detail in Chapter 7.

In terms of their scope delimiters, the four scope templates available in PROPEL are Global, After **START**, Before **END**, and Between **START** and **END**. For brevity, however, we hereafter refer to them without including the scope delimiters in their names. The Global scope template is the only one that has no alternative options for specifiers to resolve and it does not use either delimiter. In this scope template, the behavior is required to hold from the start of the event sequence until the end of the event sequence. The Global scope template is the default for a new property specification. The After scope template requires that in any event sequence, the behavior must hold after an occurrence of **START** until the end of that event sequence. The Before scope template requires that the behavior must hold from the start of any event sequence until the first occurrence of **END** in that sequence. The Between scope template requires that the behavior must hold after an occurrence of **START** until the first subsequent occurrence of **END**. Only the Between scope template can define multiple restricted intervals in a single event sequence. Figure 3.17 shows example scopes that can be instantiated based on the four scope templates. In this figure, the four horizontal arrows are the timelines on which the events occur in sequence. The vertical dashed lines that are labeled with either “**START**” or
“END” denote points in the event sequences at which those events occur. The shaded regions on a given timeline show where the restricted intervals exist in that example event sequence.

There are a number of options associated with each scope template. Since the Between scope template contains all the options that are associated with the After and Before scope templates, plus one more option that neither of those templates contains, we only describe the Between scope template in detail here. The Between scope template has three options. These options are orthogonal to each other and can be resolved in any order. One option, called first/last START, which the After scope template also has, determines what happens if START occurs more than once before the end of the given restricted interval. There are two alternative settings associated with this option. One setting is that the first occurrence of START begins a restricted interval and later occurrences of START have no effect. This is the setting used in the example After scope in Figure 3.17(b). The other possible setting is that only the last occurrence of START before the end of a restricted interval begins this restricted interval. That is, each occurrence of START resets the beginning of this restricted interval. Another option in the Between scope template, called single/multiple restricted intervals, determines whether more than one restricted interval can exist in a single event sequence. There are two alternative settings associated with this option, which are based on two possibilities of what can happen after the end of a restricted interval. One possibility is that a subsequent occurrence of START potentially begins a new restricted interval and thus multiple restricted intervals can exist consecutively. The Between scope template does not currently support nesting of restricted intervals. This topic is discussed in more detail in Chapter 7.

---

Figure 3.17. Examples of Scopes Based on the Four Scope Templates
the event sequence. This is the setting used in the example Between scope in Figure 3.17(d). The other possibility is that a subsequent occurrence of \texttt{START} does not begin a new restricted interval and thus there can be only one restricted interval in the event sequence.

Figure 3.18 illustrates the remaining option in the Between scope template and uses the same notation used in Figure 3.17. This option, called \textit{missing END}, which the Before scope template also has, determines what should happen if, after a restricted interval is begun, \texttt{END} never occurs to terminate it. There are two alternative settings associated with this option. Figure 3.18(a) shows the setting where this restricted interval exists until the end of the event sequence anyway, even though the second occurrence of \texttt{START} has no subsequent occurrence of \texttt{END} to end the second restricted interval. Figure 3.18(b) shows the other possible setting, where this potential restricted interval does not exist without a subsequent occurrence of \texttt{END} to end it.

The scope templates described here differ from earlier work done regarding scopes. The two possible settings shown in Figure 3.18 for the \textit{missing END} option distinguish two of the scopes in the Dwyer et al. property pattern approach. The “After \texttt{START} Until \texttt{END}” scope in the property pattern approach corresponds to the first setting (Figure 3.18(a)), and the “Between \texttt{START} and \texttt{END}” scope in the property pattern approach corresponds to the second setting (Figure 3.18(b)). In addition to a different organization of the scopes, there are other options and option settings that can be in the scope templates. These issues are discussed in more detail in Chapter 7.

3.2.2 Scope Question Tree

In PROPEL, we break the QT property view into two separate parts, one for the scope template (i.e., the Scope QT (SQT)) and another for the behavior template (i.e., the Behavior QT (BQT)). Similar to the BQT and its guidance for selecting a behavior template, the SQT was designed to provide guidance to PROPEL specifiers for how to select a scope template. This was not the only purpose for the SQT, however, since like the other property views, the SQT can be used to resolve each of the options that are associated with the selected scope template. Thus, the same scope options that must be resolved in the FSA template and DNL...
template property views, and in the ST scope view, are options in the SQT property view. The SQT uses the same kind of hierarchical structure and interaction that the BQT does, with the only difference being that the focus of the SQT is on the scope templates rather than on the behavior templates.

Figure 3.19 shows how the SQT organizes the set of available scopes. The SQT’s guidance for choosing among the four scope templates is structured by two questions. The root question asks whether the behavior is only required to hold within restricted intervals in the event sequence and provides two answers: either the behavior is only required to hold within certain restricted intervals, or it is required to hold throughout the entire event sequence. Specifiers can select just one of those two answers. If the behavior is only required to hold within certain restricted intervals, a child question asks specifiers how to describe those restricted intervals and provides three answers, which correspond to the After, Before, and Between scope templates, respectively. If, however, the behavior is required to hold throughout the entire event sequence, then the specifiers select that answer, which corresponds to the Global scope template. Selecting an answer to the root question and, if necessary, an answer to the child question, selects a scope template. Once specifiers select a scope template, they can continue to make decisions about the scope template’s options, if any exist, using any or all of the three property views and the ST scope view. It should be noted that Figure 3.19 differs from the actual SQT in that this figure shows the names of the scope templates next to each of the scope templates’ choices. A complete, fully-expanded version of the SQT is available in Appendix A.1.2.

3.2.2.1 An Example of Instantiating a Property’s Scope in the SQT

We can use the SQT to instantiate a scope for the example property given in Section 3.1.2.2. Suppose that the full property specification is the following:

Figure 3.19. Scope Template Organization in the SQT
Is the behavior only required to hold within restricted interval(s) in the event sequence?

Yes, the behavior is only required to hold within restricted interval(s) in the event sequence

Which of the following choices best describes the restricted interval(s)?

- There can be at most one restricted interval in the event sequence and it has a starting delimiter, receive-order: the behavior is required to hold from an occurrence of receive-order through to the end of the event sequence
- There can be at most one restricted interval in the event sequence and it has an ending delimiter, prepare-docs: the behavior is required to hold from the start of the event sequence through to the first occurrence of prepare-docs

A restricted interval in the event sequence can have both a starting delimiter, receive-order, and an ending delimiter, prepare-docs. The behavior is required to hold from an occurrence of receive-order through to the end of that restricted interval

What happens if there are multiple occurrences of receive-order without an occurrence of prepare-docs in between them?

- Only the first of those occurrences of receive-order potentially starts a restricted interval; later occurrences of receive-order within that restricted interval do not have an effect
- Only the last of those occurrences of receive-order potentially starts a restricted interval; each of those occurrences of receive-order resets the beginning of that restricted interval

If an occurrence of receive-order is not followed by an occurrence of prepare-docs, is the behavior still required to hold, until the end of the event sequence?

- Yes, if prepare-docs does not occur subsequently, then the behavior is required to hold until the end of the event sequence
- No, if prepare-docs does not occur subsequently, then the behavior is not required to hold for the remainder of the event sequence

What happens if receive-order occurs after the end of a restricted interval?

- The situation would be the same as after the first occurrence of receive-order, meaning that the restrictions described on any events that take place after that occurrence would again apply; the new occurrence of receive-order would potentially start a new restricted interval
- That new occurrence of receive-order does not have an effect; no new occurrence of receive-order will ever start a new restricted interval

No, the behavior is required to hold throughout the entire event sequence

Figure 3.20. Choosing a Scope Template in the Scope Question Tree

After the nurse receives a physician’s order to transfuse blood into a patient, if the nurse discovers that the patient’s type and screen (T&S) are not available in the lab, the nurse must obtain a blood specimen from the patient before the nurse can prepare documentation for picking up the necessary unit(s) of blood product from the blood bank.

Figure 3.20 shows the SQT view of a partially-resolved scope for this example property. This figure uses the same notation as that given in Figure 3.2. As can be seen in Figure 3.20, the first question that the SQT asks specifiers about the scope for their intended property is, “Is the behavior only required to hold within restricted interval(s) in the event sequence?” The new phrases in the example property specification above imply a beginning and an end to when it is required that the nurse check the lab for the T&S and
Is the behavior only required to hold within restricted interval(s) in the event sequence?

Yes, the behavior is only required to hold within restricted interval(s) in the event sequence.

Which of the following choices best describes the restricted interval(s)?

- There can be at most one restricted interval in the event sequence and it has a starting delimiter, receive-order: the behavior is required to hold from an occurrence of receive-order through to the end of the event sequence.
- There can be at most one restricted interval in the event sequence and it has an ending delimiter, prepare-docs: the behavior is required to hold from the start of the event sequence through to the first occurrence of prepare-docs.
- A restricted interval in the event sequence can have both a starting delimiter, receive-order, and an ending delimiter, prepare-docs. The behavior is required to hold from an occurrence of receive-order through to the end of that restricted interval.

What happens if there are multiple occurrences of receive-order without an occurrence of prepare-docs in between them?

- Only the first of those occurrences of receive-order potentially starts a restricted interval; later occurrences of receive-order within that restricted interval do not have an effect.
- Only the last of those occurrences of receive-order potentially starts a restricted interval; each of those occurrences of receive-order resets the beginning of that restricted interval.

If an occurrence of receive-order is not followed by an occurrence of prepare-docs, is the behavior still required to hold, until the end of the event sequence?

- Yes, if prepare-docs does not occur subsequently, then the behavior is required to hold until the end of the event sequence.
- No, if prepare-docs does not occur subsequently, then the behavior is not required to hold for the remainder of the event sequence.

What happens if receive-order occurs after the end of a restricted interval?

- The situation would be the same as after the first occurrence of receive-order, meaning that the restrictions described on any events that take place after that occurrence would again apply; the new occurrence of receive-order would potentially start a new restricted interval.
- That new occurrence of receive-order does not have an effect; no new occurrence of receive-order will ever start a new restricted interval.

No, the behavior is required to hold throughout the entire event sequence.

Figure 3.21. Resolving an Option in the Scope Question Tree

subsequently obtain a blood specimen if necessary. Let us assume that given these new phrases, specifiers identify two new events in the property’s alphabet, receive-order and prepare-docs, and associate those two events with the START and END parameters, respectively. Given these events, the specifiers decide that the intended property’s behavior is only required to hold between the two scope delimiters, and select the answer that represents the Between scope template. At this point, the SQT reveals three child questions about the alternative options in the Between scope template. For the remainder of this section, we continue to elucidate this scope by using the SQT, but since a scope template has been selected, the specifiers can now work with

---

8 Again, we do not show the Alphabet View dialog (see Section 4.3) where event names are assigned to the parameters, although the results of this assignment are shown by the boldface words in Figure 3.20.
Is the behavior only required to hold within restricted interval(s) in the event sequence?
- Yes, the behavior is only required to hold within restricted interval(s) in the event sequence
- Which of the following choices best describes the restricted interval(s)?
  - There can be at most one restricted interval in the event sequence and it has a starting delimiter, receive-order: the behavior is required to hold from an occurrence of receive-order through to the end of the event sequence
  - There can be at most one restricted interval in the event sequence and it has an ending delimiter, prepare-docs: the behavior is required to hold from the start of the event sequence through to the first occurrence of prepare-docs
  - A restricted interval in the event sequence can have both a starting delimiter, receive-order, and an ending delimiter, prepare-docs. The behavior is required to hold from an occurrence of receive-order through to the end of that restricted interval
  - What happens if there are multiple occurrences of receive-order without an occurrence of prepare-docs in between them?
    - Only the first of those occurrences of receive-order potentially starts a restricted interval; later occurrences of receive-order within that restricted interval do not have an effect
    - Only the last of those occurrences of receive-order potentially starts a restricted interval; each of those occurrences of receive-order resets the beginning of that restricted interval
  - If an occurrence of receive-order is not followed by an occurrence of prepare-docs, is the behavior still required to hold, until the end of the event sequence?
    - Yes, if prepare-docs does not occur subsequently, then the behavior is required to hold until the end of the event sequence
    - No, if prepare-docs does not occur subsequently, then the behavior is not required to hold for the remainder of the event sequence
  - What happens if receive-order occurs after the end of a restricted interval?
    - The situation would be the same as after the first occurrence of receive-order, meaning that the restrictions described on any events that take place after that occurrence would again apply; the new occurrence of receive-order would potentially start a new restricted interval
    - That new occurrence of receive-order does not have an effect; no new occurrence of receive-order will ever start a new restricted interval
    - No, the behavior is required to hold throughout the entire event sequence

Figure 3.22. Resolving Another Option in the Scope Question Tree

any or all of the three property views or the ST scope view to continue elucidating the details of the intended property’s scope.

Since the Between scope template’s three child questions are orthogonal to each other, they can be answered in any order. In this discussion, we answer them in the order that they are shown in the SQT. The first child question represents the first/last START option and is concerned with what should happen if more than one receive-order occurs without an occurrence of prepare-docs in between them: should only the first of those occurrences of receive-order be expected to start a restricted interval, or should only the last of those occurrences be expected to start a restricted interval? In other words, after which occurrence of START in Figure 3.17(b) should the restricted interval start? For simplicity, let us assume that the intended property
Is the behavior only required to hold within restricted interval(s) in the event sequence?

Yes, the behavior is only required to hold within restricted interval(s) in the event sequence.

Which of the following choices best describes the restricted interval(s)?

There can be at most one restricted interval in the event sequence and it has a starting delimiter, receive-order: the behavior is required to hold from an occurrence of receive-order through to the end of the event sequence.

There can be at most one restricted interval in the event sequence and it has an ending delimiter, prepare-docs: the behavior is required to hold from the start of the event sequence through to the first occurrence of prepare-docs.

A restricted interval in the event sequence can have both a starting delimiter, receive-order, and an ending delimiter, prepare-docs. The behavior is required to hold from an occurrence of receive-order through to the end of that restricted interval.

What happens if there are multiple occurrences of receive-order without an occurrence of prepare-docs in between them?

- Only the first of those occurrences of receive-order potentially starts a restricted interval; later occurrences of receive-order within that restricted interval do not have an effect.

- Only the last of those occurrences of receive-order potentially starts a restricted interval; each of those occurrences of receive-order resets the beginning of that restricted interval.

If an occurrence of receive-order is not followed by an occurrence of prepare-docs, is the behavior still required to hold, until the end of the event sequence?

- Yes, if prepare-docs does not occur subsequently, then the behavior is required to hold until the end of the event sequence.

- No, if prepare-docs does not occur subsequently, then the behavior is not required to hold for the remainder of the event sequence.

What happens if receive-order occurs after the end of a restricted interval?

- That new occurrence of receive-order would potentially start a new restricted interval. The situation would thus be the same as after the first occurrence of receive-order, meaning that the restrictions described on any events that take place after that occurrence would again apply.

- That new occurrence of receive-order does not have an effect; no new occurrence of receive-order will ever start a new restricted interval.

No, the behavior is required to hold throughout the entire event sequence.

Figure 3.23. An Example Between Scope in the Scope Question Tree

is meant to be in the context of a single patient. In this situation, the most common reason that the nurse would receive multiple physician orders for the patient’s blood transfusion is that one or more physicians, not realizing that a transfusion has already been ordered, would place additional, unnecessary, orders. In this case, the nurse would contact the physicians who have sent the redundant orders and would ask them to retract those orders. Given this knowledge, the specifiers decide to start enforcing the behavior after the first of the occurrences of receive-order, and thus select that answer, as is shown in Figure 3.21.

The next child question represents the missing END option and pertains to any situation in which a nurse receives a blood-transfusion order from a physician, but never eventually prepares the documentation for blood pick-up. In such a situation, this question is concerned with whether the behavior is still required to
hold. Since events might occur that would negate the need to check for a T&S or to obtain a blood specimen (e.g., the physician could cancel the order before the nurse starts to prepare documentation), the specifiers select the “No” answer to the question, as is shown in Figure 3.22. This means that if the nurse never eventually prepares the documentation for blood pick-up, the behavior did not have to hold. In other words, in that situation, it doesn’t matter whether the nurse obtained a blood specimen after it was discovered that the lab did not have a T&S for the patient. Since the information that would be gained by obtaining a blood specimen will not be used for the documentation, there is no need to make sure that the blood specimen was obtained.

The final child question in the SQT represents the single/multiple restricted intervals option and is concerned with what happens after an occurrence of prepare-docs ends a restricted interval. Does a subsequent occurrence of receive-order begin a new restricted interval in which, if the nurse again discovers that there is no T&S available, a new blood specimen must be obtained? It is certainly possible that after the nurse has prepared documentation for blood pick-up (and probably after the nurse also completed the administration of the blood transfusion), the nurse may receive another physician order for a blood transfusion for the patient. Given this knowledge, the specifiers select the first answer to the question, as shown in Figure 3.23. With this final answer, the specifiers have instantiated a scope. Just for comparison, given the example sequence of starting and ending delimiters in Figure 3.18, this scope would match Figure 3.18(b).

3.2.3 Scope Finite-State Automata Templates

A scope template can be applied to a behavior FSA template by including additional states that we call “scope states” and by adding transitions between the scope states and the states of the FSA that represents the behavior. These transitions are labeled by the scope delimiters, START and END. Any specifier-defined events that are associated with these scope delimiters are added to the property’s alphabet, Σ. The scope templates’ options are represented using the same FSA template notation that is defined for the behavior FSA templates in Section 3.1.3.1. For a given scope template, the scope states and transitions that it corresponds to can be applied to any behavior FSA template, independent of which behavior FSA template that it is. The details of how to apply each scope template are given in Appendix A.2.2. For the sake of clarity in the figures given in this section, instead of separately showing the self-loops that are added for occurrences of the scope delimiters, we have changed the multi-labels on the existing self-loops to reflect these additions where possible.

It should be noted that although scope templates can be represented in the FSA template property view, they can only be shown if a behavior template has already been selected. In this situation, it is not easy to distinguish the scope-template components in the combined scope-and-behavior FSA template property.
3.2.3.1 An Example of Instantiating a Property’s Scope FSA Template

Figure 3.24 shows the FSA template property view of a partially-resolved scope for the example Blood Transfusion property given in Section 3.2.2.1. Before PROPEL can show this FSA template view to specifiers, however, the Between scope template must be selected, either in the SQT, as is shown in Figure 3.20, or in the ST (see Figure 3.34). The options in the behavior template that is shown in the FSA template property view in Figure 3.24 have already been resolved according to the settings given in Section 3.1, so all the optional components shown in Figure 3.24 are those associated with the selected scope template.
To create the same scope that we previously described using the SQT, specifiers must resolve the three options in the Between scope template. It is possible to use the FSA template property view to resolve the options in a scope template in any order, but here we resolve the options in the same order that they were resolved in the SQT discussion in Section 3.2.2.1. The first option to resolve is first/last START, and in the FSA template view shown in Figure 3.24, that option is determined by multiple optional components. Since this option is concerned with what happens if receive-order occurs more than once before the end of a given restricted interval, this option is represented in the FSA template property view by the outgoing transitions on receive-order from each of the states that are also in the original behavior FSA template: states 2, 3, 4, and 6. Specifiers can select an option setting by determining the existence of these transitions. Since all of these transitions represent the same option, changing the existence of one of them affects all of them. As mentioned previously, the FSAs produced by PROPEL are deterministic, so for a given state, the existence of an optional outgoing transition and the existence of a self-loop on the same event are mutually exclusive. In this case, each of those four states has a self-loop that can be labeled by receive-order. To make the same choice that was made in the SQT to resolve the first/last START option, specifiers can edit the self-loop on state 3, for example, as is shown by the highlighted choice in the menu shown in Figure 3.24, and cause receive-order
Figure 3.26. Resolving a Scope Option in the FSA Template

to occur on those four states’ self-loops. Thus, similar to the SQT in Figure 3.21 the resulting FSA template potentially starts to enforce the behavior after the first occurrence of \textit{receive-order}.\footnote{When the transition on \textit{receive-order} from state 6 to state 2 in Figure 3.24 is removed by the chosen option setting, state 6 effectively becomes a non-accepting trap state, and is thus not shown in the subsequent FSA template figures in this section.}

Continuing to resolve the options in the same order that is given in the SQT example in Section 3.2.2.1, the next option to resolve is \textit{missing END}. In the FSA template property view in Figure 3.25, that option is determined by the accepting status of state 3. If that state is accepting, then it is not possible for the behavior to violate the property unless \textit{prepare-docs} occurs at that state, and thus the behavior is not enforced unless \textit{prepare-docs} occurs to end the given restricted interval. By contrast, if that state is not accepting, then it is possible for the behavior to violate the property whether or not \textit{prepare-docs} occurs to end the given restricted interval, and thus the behavior is enforced either way. In the FSA template view shown in Figure 3.25, the \textit{missing END} option can be resolved by selecting state 3’s accepting status using the menu shown in that figure. For comparison, the same choice that was made in the SQT in Figure 3.22 is highlighted in the menu in Figure 3.25, and the resulting FSA template does not enforce the behavior if \textit{prepare-docs} does not occur to end the given restricted interval.

The remaining option that must be resolved in the Between scope template is \textit{single/multiple restricted intervals}. In the FSA template property view in Figure 3.26, that option is determined by which outgoing transition on \textit{receive-order} from state 5 exists. If \textit{receive-order} is on the transition from state 5 to state 2, then...

---

\begin{itemize}
\item \textit{receive-order}:
\item \textit{prepare-docs}:
\item \textit{obtain-specimen}:
\item \textit{label-vial}:
\item \textit{no-T&S}:
\end{itemize}
there can be multiple restricted intervals in the sequence of events. By contrast, if receive-order is on state 5’s self-loop, then there can be at most one restricted interval in the sequence of events. For comparison, the same choice that was made to resolve single/multiple restricted intervals in the SQT in Figure 3.23 is highlighted in the menu in Figure 3.26, and the resulting FSA, which is shown in Figure 3.27, allows multiple restricted intervals to occur in the sequence of events. This FSA corresponds to the completed SQT in Figure 3.23.

3.2.4 Scope Disciplined Natural Language Templates

A scope can be applied to a behavior DNL template by simply adding more phrases to those that are described in Section 3.1.4.1. To aid the readability of the combined scope-and-behavior DNL template property view, we divide the combined DNL template into three sections, in the following manner:

1. The preamble, which provides some context for how to interpret the property:

   “This property is composed of two parts: a behavior and a scope. The behavior specifies the restrictions on the occurrences of events, and the scope specifies the interval(s) in the event sequences within which the behavior restrictions are required to hold. This property does not impose any restrictions on the occurrences of events that are not explicitly mentioned in this property specification.”

2. The description of the scope:
A scope DNL template consists of the text “SCOPE:” as a header, and one or more numbered phrases, in the given order:

(a) A statement about what the scope delimiters in the scope template are, if there are any.
(b) A statement about when in the event sequence the behavior is required to hold. If the scope template contains a START, this statement also describes the first/last START option and its possible settings.
(c) Zero or more subsidiary phrases, which describe the other two scope-template options and their possible settings.
(d) A statement about when there are no restrictions placed on the behavior, if there are any such intervals.

3. The description of the behavior:

A behavior DNL template consists of the text “BEHAVIOR:” as a header, and several subsidiary phrases. See Section 3.1.4.1 for a description of those phrases in the behavior DNL templates.

For examples of combined scope-and-behavior DNL property specifications, see Appendix C. We now look in more detail at how the phrases that describe the scope are represented in the DNL template property view of an example scope template.

3.2.4.1 An Example of Instantiating a Property’s Scope DNL Template

Recall the example Blood Transfusion property given in Section 3.2.2.1. Just as the FSA template in Figure 3.24 is a representation of the Between scope template (when it is applied to the Response behavior) that was selected in the SQT in Figure 3.20, the DNL template shown in Figure 3.28 is also a representation of that scope template. Before PROPEL can show this DNL template property view to specifiers, however, the Between scope template must be selected either in the SQT or in the ST (see Figure 3.34). Phrase 1 in this Between DNL template describes the two scope delimiters that were identified in Section 3.2.2.1, receive-order and prepare-docs. Phrase 2 describes the core concept of the Between scope. Phrases 3-5 are the subsidiary phrases for the Between DNL template, expressing the settings for the three options in the Between scope template. Phrase 6 describes when there are no restrictions placed on the behavior.

To create the same scope that was previously described using the SQT and the FSA template property views, specifiers must resolve the three options in the Between scope template. It is possible to use the DNL template property view to resolve the options in a scope template in any order, but here we resolve them in
SCOPE:
1. A restricted interval in the event sequence can have both a starting delimiter, receive-order, and an ending delimiter, prepare-docs.

2. The behavior is required to hold from an occurrence of receive-order, if it ever occurs, through to the first subsequent occurrence of prepare-docs, if it ever occurs.

3. If there are multiple occurrences of receive-order without an occurrence of prepare-docs in between them, only the first occurrence of receive-order potentially starts a restricted interval; later occurrences of receive-order within this restricted interval do not have an effect.

4. receive-order is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if receive-order does occur, prepare-docs is not required to occur subsequently.

5. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

Figure 3.28. Example Partially-Resolved Scope DNL Template

SCOPE:
1. A restricted interval in the event sequence can have both a starting delimiter, receive-order, and an ending delimiter, prepare-docs.

2. The behavior is required to hold from an occurrence of receive-order, if it ever occurs, through to the first subsequent occurrence of prepare-docs, if it ever occurs.

3. If there are multiple occurrences of receive-order without an occurrence of prepare-docs in between them, only the first occurrence of receive-order potentially starts a restricted interval; later occurrences of receive-order within this restricted interval do not have an effect.

4. receive-order is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if receive-order does occur, prepare-docs is not required to occur subsequently.

5. and if it does not occur subsequently, then the behavior is not required to hold for the remainder of the event sequence.

6. Even if END does not occur subsequently, the behavior is still required to hold, until the end of the event sequence.

Figure 3.29. Resolving an Option in the Scope DNL Template
There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

**SCOPE:**

1. A restricted interval in the event sequence can have both a starting delimiter, `receive-order`, and an ending delimiter, `prepare-docs`.

2. The behavior is required to hold from an occurrence of `receive-order`, if it ever occurs, through to the first subsequent occurrence of `prepare-docs`, if it ever occurs.

3. If there are multiple occurrences of `receive-order` without an occurrence of `prepare-docs` in between them, only the first of those occurrences of `receive-order` potentially starts a restricted interval; later occurrences of `receive-order` within this restricted interval do not have an effect.

4. `receive-order` is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if `receive-order` does occur, `prepare-docs` is not required to occur subsequently and if it does not occur subsequently, then the behavior is not required to hold for the remainder of the event sequence.

5. There can be at most one restricted interval in the event sequence.

6. There might be many restricted intervals in the event sequence. If `receive-order` occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of `receive-order`, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a potential new restricted interval.

**Figure 3.30.** Resolving an Option in the Scope DNL Template

the order that they were resolved in the SQT example in Section 3.2.2.1. Phrase 3 expresses the setting for the first/last START option. Figure 3.28 shows how this option can be resolved in the DNL template by selecting one of the choices in the combo box for phrase 3. For comparison, the same choice that was selected in the SQT in Figure 3.21 and in the FSA template in Figure 3.24 is highlighted in the combo box for phrase 3 in Figure 3.28.

Continuing to resolve the options in the same order that is given in the SQT example in Section 3.2.2.1, the next option to resolve is missing END. In the DNL template property view in Figure 3.29, that option is expressed by phrase 4. The missing END option can be resolved in the DNL template by selecting one of the choices in the combo box for that phrase. For comparison, the same choice that was selected in the SQT in Figure 3.22 and in the FSA template in Figure 3.25 is highlighted in the combo box for phrase 4 in Figure 3.29.

The remaining option that must be resolved in the Between scope template is single/multiple restricted intervals. In the DNL template property view shown in Figure 3.30, that option is expressed by phrase 5. The single/multiple restricted intervals option can be resolved in the DNL template by selecting one of the choices in the combo box for that phrase. For comparison, the same choice that was selected in the SQT in Figure 3.23 and in the FSA template in Figure 3.26 is highlighted in the combo box for phrase 5 in Figure 3.30.
There might be many restricted intervals in the event sequence. If receive-order occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of receive-order, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a potential new restricted interval.

<table>
<thead>
<tr>
<th>SCOPE:</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. A restricted interval in the event sequence can have both a starting delimiter, receive-order, and an ending delimiter, prepare-docs.</td>
</tr>
<tr>
<td>2. The behavior is required to hold from an occurrence of receive-order, if it ever occurs, through to the first subsequent occurrence of prepare-docs, if it ever occurs.</td>
</tr>
<tr>
<td>3. If there are multiple occurrences of receive-order without an occurrence of prepare-docs in between them, only the first of those occurrences of receive-order potentially starts a restricted interval; later occurrences of receive-order within this restricted interval do not have an effect.</td>
</tr>
<tr>
<td>4. receive-order is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if receive-order does occur, prepare-docs is not required to occur subsequently and if it does not occur subsequently, then the behavior is not required to hold for the remainder of the event sequence.</td>
</tr>
<tr>
<td>5. There might be many restricted intervals in the event sequence. If receive-order occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of receive-order, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a potential new restricted interval.</td>
</tr>
<tr>
<td>6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).</td>
</tr>
</tbody>
</table>

**Figure 3.31. Example Scope DNL Specification**

The resulting DNL description, shown in Figure 3.31, corresponds to the completed SQT in Figure 3.23 and the completed FSA in Figure 3.27.

### 3.2.5 Scope Timeline

One of our aims in choosing property views was to provide a graphical representation of a property. Although scopes can be shown in the graphical FSA template property view, it can be difficult to understand the scope templates in that property view. (See Section 3.2.3 for an explanation of why.) Since the scope and behavior concepts are orthogonal, it is possible to provide a separate graphical representation of each of them. In keeping with this aim, we have been experimenting with the ST scope view, which is another view (in addition to the SQT) that provides guidance for selecting a scope template. Some form of a graphical timeline representation has been used to represent events and time intervals in several other approaches, such as GIL [77], the Timeline Editor [238], and the original descriptions of the scopes in the property pattern approach [81, 82].

The ST scope view is in the spirit of the timelines used to illustrate the scope templates’ options in Figures 3.17 and 3.18. Figure 3.32 gives the notation used in the ST scope view and explains the notation using the equivalent DNL descriptions. Time flows from left to right, in the direction of the timeline arrow, which is the top arrow. The restricted interval, which defines the chosen scope, is denoted by the bottom arrow. Unlike the timeline arrow, the restricted-interval arrow can change its graphical starting and ending
locations, based on the presence or absence of the starting and ending delimiters, and based on the option settings for the selected scope template. As can be seen in Figure 3.33, specifiers can interact with this scope view via contextual menus whose contents are dependent on which of the scope delimiters are present and what the option settings currently are. Specifiers can click on different areas in the ST to select a scope template (i.e., by adding or removing the scope delimiters) and to resolve the options in the selected template.

3.2.5.1 An Example of Instantiating a Property’s Scope in the Timeline

Figure 3.33 shows the ST scope view of the default Global scope for the example Blood Transfusion property that was used in Section 3.2.2.1. A Global scope is not the appropriate scope for this property, however. To create the same scope with the ST scope view that we previously described using the SQT, FSA template, and DNL template views, specifiers must first add the scope delimiters, receive-order and prepare-docs, to the property’s alphabet and associate them with the START and END parameters, respectively. Then the specifiers must add those scope delimiters to the ST scope view. As is shown in Figures 3.33 and 3.34, specifiers can work with the appropriate contextual menus to add those two delimiters. After adding both delimiters, specifiers have selected the Between scope template, as is shown in Figure 3.35.
It is possible to use the ST scope view to resolve the options in a scope template in any order, but in this example, we resolve the options in the same order that they were resolved in the SQT discussion in Section 3.2.2.1. One option that can be resolved in the ST is *first/last START*. To make the same decision about this option that was made in the SQT in Figure 3.21, specifiers must change the graphical start of the restricted-interval arrow to be associated with the first occurrence of *receive-order* on the ST. To do this, specifiers can select the choice that is highlighted in the contextual menu in Figure 3.35. The other option that can be resolved in the ST scope view is *missing END*. To make the same decision about this option that was made in the SQT in Figure 3.22, specifiers must change the graphical end of the restricted-interval arrow to be before the occurrence of *prepare-docs* on the ST. To do this, specifiers can select the choice that is highlighted in the contextual menu in Figure 3.36.
Figure 3.36. Resolving an Option in the Scope Timeline

Figure 3.37. An Example Between Scope in the Scope Timeline

Due to the limitations of the current version of the ST scope view, only two of the three options in the Between scope template can be resolved using this scope view. Since the current version of the ST does not support the third option in the Between scope template, at this point the ST scope view of the Between scope that is shown in Figure 3.37 is completed. To resolve that third option in the Between scope template, single/multiple restricted intervals, specifiers must either work with the SQT, as is shown in Figure 3.23, or with the FSA or DNL template property views, which are presented in the following sections. Chapter 7 discusses possible extensions and improvements to this scope view, and a version that allows specifiers to resolve all the scope template options is given in Appendix A.4.5.

3.2.6 Using All Four Views to Instantiate a Property’s Scope

Although the previous sections illustrated the three property views and the ST scope view by elucidating an example scope using only the ST or only one view at a time, in PROPEL specifiers can work with one or more of the property views or with the ST scope view and can change back and forth between them as the property is being elucidated. As was discussed previously, specifiers must start elucidating the scope by selecting a scope template using either the SQT property view or the ST scope view, because neither the FSA nor DNL template property views provide support for selecting among the scope templates. Once a scope
template has been selected, however, specifiers can resolve the scope options using any or all of the three property views or the ST scope view. As specifiers resolve the scope options, PROPEL automatically keeps the three property views and the ST scope view synchronized.

For example, consider one possible series of figures that illustrates what specifiers might encounter if they choose to work with all three property views and the ST scope view. To elucidate the scope for the example property that has been presented in the previous sections, specifiers would first select a scope template using the ST, as is shown in Figures 3.33 and 3.34. Specifiers could continue to resolve the options in that scope template using the ST scope view, but let us assume that they switch to using one of the property views, such as the SQT shown in Figure 3.21. In that figure, the Between scope template is already selected, to match the specifiers’ previous selection in the ST scope view, and now the specifiers are resolving the first/last START option. Specifiers could continue to resolve the remaining two options in the Between scope template using the SQT, but let us assume that they switch to using the FSA template property view. Figure 3.25 shows the first/last START option already resolved to match the specifiers’ previous selection in the SQT, and now the specifiers are resolving the missing END option. Specifiers could continue to resolve the last option in the Between scope template using the FSA template property view, but let us assume that they switch to using the DNL template property view. Figure 3.30 shows the missing END option already resolved to match the specifiers’ previous selection in the FSA template property view, and now the specifiers are resolving the single/multiple restricted intervals option. After that point, specifiers have resolved all the options in the Between scope template, and they are left with the SQT shown in Figure 3.23, the ST shown in Figure 3.37, the FSA shown in Figure 3.27, and the DNL text shown in Figure 3.31. Any option can be revisited and changed whenever specifiers need to do so. Specifiers can also resolve the options in a different order than the one described above, and can make different decisions about when to use the property and scope views while resolving those options.

3.2.7 Scope Summary

Our approach supports four different views that enable specifiers to elucidate their desired property’s scope: the SQT view, the FSA template view, the DNL template view, and the ST view. The SQT view was designed to provide specifiers with hierarchical, question-based guidance for how to select the appropriate scope template. The FSA template view was designed to assist specifiers in creating a graphical, formal view of their desired scope, in this case an FSA. The DNL template view was designed to assist specifiers in creating an understandable representation of their scope. The ST view was designed to provide specifiers with limited guidance for selecting the appropriate scope template and also a graphical, timeline-based view of their scope. Specifiers can work with any of the views to resolve each of the options associated with the
Table 3.3. Comparison of Property and Scope Views in Terms of Design Goals

<table>
<thead>
<tr>
<th>Design Goal</th>
<th>Reason for Importance</th>
<th>QT</th>
<th>DNL</th>
<th>FSAT</th>
<th>ST</th>
</tr>
</thead>
<tbody>
<tr>
<td>Based on natural language</td>
<td>Provides an <strong>understandable</strong> representation of the property; does not require specifiers to learn a formalism</td>
<td>✓</td>
<td>✓</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Based on a formalism</td>
<td>Provides a <strong>precise</strong> representation of the property; eliminates ambiguity and can be used in automated analyses</td>
<td></td>
<td></td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>Supports selection of a scope and/or behavior template</td>
<td>Provides <strong>guidance</strong> through the property-elucidation process</td>
<td>✓</td>
<td></td>
<td></td>
<td>✓</td>
</tr>
</tbody>
</table>

A selected scope template. Thus, the same options that must be resolved in one view are options in the other views as well. After resolving the options using any view, specifiers can revisit their decisions at any time. A fully-resolved scope is represented with several DNL paragraphs, one FSA, one fully-resolved SQT, and one fully-resolved ST. The scope example given in this section showed only how the Between scope template is represented in the four views, but a detailed description of the other three scope templates and how they are represented in all four views is included in Appendix A.

### 3.3 Summary

The primary motivation behind our work is to enable specifiers who are unlikely to have expertise in specification formalisms to create precise, rigorous, and understandable property specifications. To support this goal, our approach provides property pattern templates that leverage the experience of property specification experts, and these templates explicitly draw specifiers’ attention to the possible variations in how the property patterns might be interpreted. The aim of this approach is to help specifiers make informed choices about which variation is suitable for their intended property. In addition to providing property pattern templates, we also provide three different property views, which are designed to support precision, understandability, and guidance, and a scope view, which is designed to provide additional guidance for reasoning about scopes. We developed a tool, called PROPEL, which enables specifiers to work with one or more of the property or scope views at the same time, depending on the specifiers’ level of comfort with each. We found this aspect of our approach to be valuable during our case study evaluation, since different domain experts were comfortable using different property views. This observation is discussed in more detail in Section 5.4.4.2.2. In addition to supporting multiple property views, PROPEL automatically keeps these views synchronized with each other whenever possible: as specifiers make a change in one, the others change correspondingly. The design of PROPEL is discussed in more detail in Chapter 4.
There were several considerations that guided our selection of the property and scope views that PROPEL supports. Table 3.3 shows this set of design goals and the reasons for each of them and, for each of the views, which of the design goals that view meets. As this table shows, these views are complementary in their strengths and limitations. None of the views meets all of the design goals, since some of the goals (e.g., understandability and precision) appear to be mutually exclusive in practice. Thus, our intention is to provide multiple views to ensure good coverage of these design goals.

One property view that PROPEL supports is a graphical, extended FSA template that resolves to an FSA. We selected FSAs as our specification formalism, because in our experience the graphical FSA notation seems to be more understandable than most other formal notations. In addition, since the fully-resolved form of this property view is an FSA, which is mathematically well-defined, this view thus offers the precision necessary to support unambiguous communication and to be used as the input to many types of automated analysis tools, such as those that support FSV or automated test-generation techniques. This property view has some limitations, however. Although in our experience this graphical FSA notation seems to be more understandable than most other formal notations, its adoption might still be limited because it requires specifiers to obtain additional training to learn its formal language. This observation is discussed in more detail in Section 5.4.4.2.2. Another limitation that is unique to the FSA template property view is that although a property’s behavior can be viewed and modified independent of the property’s scope, the reverse is not true. A property’s scope is conceptually orthogonal to its behavior and it should thus be possible for specifiers to reason about the two parts independently. For complex properties, such as those with a two-event behavior and a Between scope, it can be difficult to distinguish which parts of the FSA are related to the scope and which are related to the behavior, and this can impede specifiers’ attempts to understand and modify their property’s scope.

By contrast, PROPEL also supports an informal DNL template property view, which provides a list of alternative phrases for each of the options in the property pattern templates, as well as a few synonyms, to support specifier customization. The DNL template property view is designed to appeal to those specifiers who prefer a natural-language-based description of their intended property. We selected DNL templates because specifiers do not need to have expertise in a particular property specification formalism to use this property view. The fully-resolved form of this property view is a readable, paragraph-style description of the property. This property view is also the only one that allows specifiers to customize their property specification somewhat, since it supports the use of synonymous phrases. One drawback that has been observed with the DNL template property view, however, is that the DNL paragraphs can appear to be long and legalistic. This limitation seems necessary to express all of the subtle details unambiguously.
Neither of those two property views addresses the issue of how to guide specifiers in selecting the appropriate property pattern template to start with, however, and thus we designed another property view, the QT, to provide specifiers with this guidance. In addition, the QT property view can also help specifiers to resolve the options associated with the selected template. The QT property view is a hierarchical, natural-language-based property view that poses questions and, for each question, provides a set of alternative answers for specifiers to select from. The QT property view breaks up the problem of deciding which property pattern template is most appropriate by asking specifiers to consider only one differentiating attribute at a time. This property view is basically a decision tree, a representation that has often been used in RE, and its tree-style formatting supports an isolation of concerns, only presenting specifiers with questions that are relevant in the context of the specifiers’ previous answers. The QT property view, unlike all the others, shows the decisions that were made to select a property pattern template and the alternatives to those decisions: specifiers can see the sibling answers that were rejected. The strength of this property view may also be a limitation, however: the hierarchical presentation of questions and of both the chosen and rejected answers might not provide a single, coherent picture of the whole property, since there is purposely a lot of repetition in the natural-language text. Finally, the QT property view is an informal one, and thus does not support precision to the degree that the FSA template property view does.

Finally, we have experimented with the ST scope view, which is designed to provide an understandable graphical representation of a property’s scope, and it guides specifiers through selecting a scope template. Unlike the FSA template property view, which also provides a graphical representation of a property’s scope, the ST scope view provides a representation of the scope in isolation, thus allowing specifiers to reason about the scope independently of the behavior. This scope view uses a timeline metaphor to support understandability, because this metaphor is commonly used when graphically describing a sequence of events or states. There are some limitations in this scope view, however. It does not leverage natural language except in specifier-defined event names, so specifiers are required to learn the view’s notation to be able to understand it. It is not based on an established formalism, so it is not amenable to use in automated analyses. It is awkward to express an unbounded number of multiple restricted intervals using such a timeline metaphor, which is part of why this scope view does not yet support the single/multiple restricted intervals scope option. In addition, even for those options that are expressed in the ST scope view, not all of the optional components that represent them are clearly indicated as such, and the ST scope view does not indicate when the scope options are unresolved. This scope view can thus mislead or confuse specifiers when it appears inconsistent with the scope information displayed in the property views. A discussion of future work on this topic is given in Chapter 7.
In addition to the limitations of the individual property and scope views, there are also a number of larger challenges that our approach needs to address. For example, there is a need for more guidance on how to decompose a property into the appropriate scope and behavior. There are also several areas where PROPEL’s current expressibility might need to be expanded, such as new behaviors, a loosening of the restrictions on the scopes, and support for more than just an event-based paradigm. Beyond these issues, there are certain aspects of our approach that challenge its ease of use. One notable issue is that the fact that our approach exposes numerous subtle details and asks specifiers to resolve all of them, which can potentially cause specifiers to become distracted by all the details and lose their focus on the core concept that motivates the property. Another issue is that in some properties, the subtle details require computer science expertise to resolve. For example, a core assumption of our approach is that each property specification should focus on a single aspect of system behavior, and this sometimes requires that specifiers make a non-intuitive mapping from an initial informal specification to multiple formal specifications in PROPEL. Although there are a number of limitations to our approach, our aim in showing these four views is to provide guidance so that specifiers can carefully resolve the details of their intended property and thus create a property specification that is both precise and understandable. Chapter 5 describes an evaluation of this approach and Chapter 7 includes a discussion about the extent to which this aim has been met.
CHAPTER 4
THE PROPEL TOOL

We have developed a system, called PROPEL, for “PROPerty ELucidation,” for the purpose of evaluating our approach. PROPEL supports the elucidation of formal property specifications, by providing an implementation of the property pattern templates and the property and scope views described in Chapter 3. In this chapter, we first describe a few of PROPEL’s design goals and the architecture that we chose to support those goals, including its support for the property pattern templates and the property and scope views. We then discuss how PROPEL also provides support for managing a property’s alphabet and we conclude with a description of the tool’s two forms of support for organizing and viewing multiple property specifications.

4.1 Design Goals

There are five main design goals for PROPEL. The foremost goal is usefulness: since the primary aim of our approach is to enable specifiers who might not have expertise in a property-specification formalism to create precise and understandable property specifications, the tool must provide property views that support those qualities. Another goal is that the tool should enable specifiers to quickly learn our approach to property specification and remember how to use it with little effort when elucidating multiple properties. Chapter 5 includes a discussion of how well PROPEL supports these usefulness and learnability design goals. The focus of this chapter is on the remaining three design goals. One goal is that PROPEL should be able to quickly keep all of the property and scope views that specifiers might be working with synchronized, so that specifiers can continually have a coherent and accurate representation of the results of their decisions about their property. Another goal is that PROPEL should enable users to not just create one property specification, but to also organize and navigate through multiple property specifications. The remaining design goal for PROPEL is focused on its support of our future research agenda. This research is by no means complete, and as we continue to develop our approach and to evaluate it, we want PROPEL to have an extensible architecture, so that we can add support for new property pattern templates, new options, new parameters, new property or scope views, new ways to connect to other tools, and new insights into how to better support all of these design goals.
4.2 Architecture

PROPEL’s architecture is based on an object-oriented implementation of the Model View-Controller design pattern [104], with the Observer pattern [104] as the primary communication structure. PROPEL’s Model is the property pattern template, which has the set of scope templates and the set of behavior templates. In a fully-instantiated property, only one behavior template and only one scope template are selected, although the property pattern template keeps references to all the behavior- and scope-template possibilities so that specifiers can revisit their decisions and make changes if necessary. Each scope and behavior template has a set of zero or more options and defines zero or more relationships (i.e., constraints) between those options, in terms of the options’ settings. PROPEL’s View-Controller is the collection of property and scope views and the embedded logic for how those views are related to the underlying Model. The property and scope views contain the instructions for how to display themselves on the screen. These views are each based on a separate view-model. Each view-model has a number of view components. For example, in the case of the FSA template view-model, there are separate view components for each of the states, the transitions, and the multi-labels. Each view component has a set of view component settings, each of which contains a mapping between one possible visual display of that view component and the associated collection of option settings in the corresponding scope or behavior template. For example, a non-accepting status for the Response behavior FSA template’s start state view component is associated with the “A is required to occur” setting for the Nullity option in the Response behavior template. A full description of the view component settings for each of the property and scope views is given in Appendices A.1, A.2, A.3 and A.4.

As specifiers make decisions about the options in the selected scope and behavior templates by changing the view component settings, the scope and behavior templates in the Model are notified of those changes. The scope and behavior templates each manage their own options to maintain the internal relationships, if any exist. When a property is fully instantiated, all of the options in the selected scope and behavior templates are resolved, although the scope and behavior templates keep references to all the possible option settings so that specifiers can revisit their decisions and make changes if necessary. Each option has three possible settings: unresolved, allowed, and not allowed. An option can be in only one of its three settings at any point in time.

1The actual implementation of PROPEL has only one scope template class, which supports the four separate scope templates via appropriate method calls. This design makes it easier and more efficient to manage the scope templates’ shared options than it would be in a design where the four scope templates are in separate classes. A similar design could be applied to the behavior templates, but partly for historical reasons, and partly because the relationships between the behavior templates are likely to change in the future, we chose to keep the behavior templates separate.

2The abstract concept of the Controller as it is described in [104] is actually separated into two parts in PROPEL. One part is a collection of third-party libraries, which translate low-level keyboard and mouse actions into events that are meaningful in terms of the view-models’ visual component settings. This part of the Controller is not unique to PROPEL, however, and is not discussed further here. The other part of the Controller is what we refer to as PROPEL’s “View-Controller,” because it is logic that is distributed throughout the view-models to define a mapping between each view component’s settings and the option settings in the corresponding scope or behavior template in the Model.
Figure 4.1. PROPEL Architecture
Figure 4.1 shows all the major parts of PROPEL’s architecture and how they communicate with each other. The figure shows how the design is divided into Model and View-Controller. The dashed arrows show the has-a relationships, and the solid numbered arrows show which parts use the Observer pattern to keep the Model and the View-Controllers synchronized. The numbers on those solid arrows denote the order that communications occur in. It should be noted that, for the sake of clarity in this figure, not all of the arrows are shown. In actuality, there should be solid numbered arrows and dashed arrows going from every view component setting to the property pattern template. In addition, there should be solid numbered arrows going between every pair of behavior templates that share the same options.

As mentioned previously, PROPEL’s communication structure is primarily based on the Observer design pattern. In this design pattern, an object referred to as a listener registers to observe the activities of another object. When the second object does something of interest, it sends a message to all of its listeners so that they can respond appropriately. To illustrate how this design pattern is used to keep the Model and the View-Controller in PROPEL synchronized, we give an example where specifiers make a decision about their property using the QT property view, and that decision propagates to all the other property and scope views as well. Let us assume that the specifiers have already selected the Response behavior template and are now resolving the Nullity option by answering the question shown in Figure 3.3 in Chapter 3. The specifier-editable view components in the BQT property view are answer nodes in its tree display. Specifiers highlight the appropriate answer node and the underlying view-model in the View-Controller sends a message to the property pattern template in the Model. This message corresponds to the solid arrow labeled “1” in Figure 4.1, and it contains Controller logic that maps from the selected answer to a particular option setting. The property pattern template forwards the message on to the Response behavior template. This message corresponds to the solid arrow labeled “2” in Figure 4.1. The Response behavior template receives the message and makes sure that if there are any relationships between the setting the specifiers chose for the Nullity option and the settings for the other options in this behavior template, those relationships are enforced (i.e., other options’ settings are updated). Once this has been completed, the Response behavior template sends a message to the Precedence behavior template with the changed option setting(s), since Precedence shares all of its options with Response. This message corresponds to the solid arrow labeled “3” in Figure 4.1. The Precedence behavior template receives the message and makes sure that its own internal relationships are enforced. The method call that the property pattern template used to send a message to the Response behavior template then returns. The property pattern template sends a message to each of the property and scope views. This message

---

3 A message is also commonly referred to as an “event”, but to avoid confusion over that term, since it has a different meaning elsewhere in our work, we use “message” instead.
corresponds to the solid arrows labeled “4”, “5”, “6”, and “7” in Figure 4.1, and it contains information about which options were changed. Each property and scope view receives the message in turn and, if that view represents those options, it goes through all of its view component settings and has them update themselves by getting the current settings for all of their options from the property pattern template. At this point, all of the property and scope views and the Model have been updated with the change originally made to the Nullity option in the BQT property view.

This architecture supports the design goals for PROPEL to the extent necessary to use the tool in an evaluation of this approach, but there is also room for improvement. With respect to the technical aspects of keeping the property and scope views synchronized, the widespread use of the Observer pattern does enable the tool to quickly provide consistent feedback as specifiers make decisions about their intended property. With respect to the extensibility needed to support our future research agenda, this architecture makes it easy to add any number of new property or scope views, options, parameters, and (although it is not discussed in detail here) ways of connecting to other tools. This architecture is not as extensible as we need it to be, however, since its organization of the behavior templates is very strongly tied to the current concept of separate behaviors in our approach. Chapter 7 describes an alternative organization of the behavior templates that would require a different architecture that supports a more fluid understanding of the current and future behaviors and their relationships to each other.

4.3 Alphabet Views

In addition to support for the property and scope views, PROPEL also provides support for managing a property’s alphabet, which is the set of events of interest in the property. An alphabet has zero or more specifier-defined events and one or more parameters, where the parameters that it includes are the ones that are associated with the selected scope and behavior templates. Recall from Chapter 3 that parameters are placeholders that can be associated with specifier-defined events. A parameter’s default name (e.g., A, B, START, and END) is displayed in the property pattern template until a specifier-defined event is associated with the parameter. Specifiers can add events to the alphabet and must associate those events with the parameters to fully instantiate the property. Based on the example property used throughout Chapter 3, Figure 4.2 shows the parameters in the selected Response behavior template and Between scope template, and the specifier-defined events that would be created for that example property.

Specifiers interact with the property’s alphabet through the two alphabet views: the Parameter Associations View and the Alphabet Manager, shown in Figure 4.2. To simplify the previous discussion about PROPEL’s architecture, the alphabet and the alphabet views were not mentioned in Section 4.2, but they are
also woven throughout the Model View-Controller architecture. The alphabet and the alphabet views also use the Observer pattern to stay synchronized with the rest of PROPEL; they send messages to and receive messages from the PROPEL Model and the property and scope views. As is shown in Figure 4.2, one alphabet view is the Parameter Associations View, which is shown in the upper portion of the figure. This view displays the parameters in the selected scope and behavior templates. Specifiers can create events and select events that they previously created, using the appropriate combo box in this view. After specifiers have selected the events that they want to associate with the parameters, they must press the Apply button to replace the parameters’ default names with the event names in the property and scope views. The other alphabet view, which is shown in the bottom portion of Figure 4.2, is the Alphabet Manager. This view displays the list of all specifier-defined events in the alphabet and allows specifiers to add, edit, and remove events. In so doing,
this view allows specifiers to add secondary events; see Chapter 3 for a discussion of the uses of secondary events.

The alphabet views have a number of strengths and weaknesses. Like the property and scope views, PROPEL keeps the alphabet views continually synchronized with the other views and with the Model, so that specifiers can have an accurate and coherent representation of their decisions about the property and its alphabet at all times. The alphabet views also support a degree of flexibility, since the parameters can be associated with any specifier-defined events and can be changed as necessary, at any time. For example, by using only the parameters’ default names at first, specifiers can explore the scope and behavior templates without needing to associate specifier-defined events until they are ready to do so. Conversely, the alphabet views enable specifiers to keep the parameter associations constant while trying different scope and behavior templates to see which ones seem to fit their intended property. This ability also exposes a weakness in the alphabet views, however. The alphabet views provide no clear guidance for which specifier-defined events should be associated with which parameters. This presents a problem if specifiers associate the parameters with the wrong events, because it can cause scope and behavior templates that would be appropriate for the intended property to seem inappropriate. Another weakness, which we observed during the evaluation described in Chapter 5, is that the alphabet views do not yet offer explicit support of event disjunctions and conjunctions. Possibilities for addressing these shortcomings are discussed in Chapter 7.

### 4.4 Support for Multiple Properties

Since the primary purpose of PROPEL is to support an evaluation of our approach, and in an evaluation based on a real-world domain there are likely to be many properties involved, it is important that PROPEL enables specifiers to work with multiple properties. Thus, in addition to the support for elucidating individual property specifications, there are two capabilities that PROPEL supports with respect to multiple properties. It enables specifiers to interact with a set of properties as a cohesive group, called a project, and to organize properties and projects as they see fit, using the Project Tree View. Finally, PROPEL enables specifiers to quickly summarize a set of properties and view that summary from multiple perspectives, using the Summary Views. These two views are discussed in more detail in the following sections.

#### 4.4.1 Project Tree View

The Project Tree View, shown in Figure 4.3, allows specifiers to organize multiple properties and multiple alphabets under a single root project. As its name indicates, the Project Tree View displays a project and its constituent properties and alphabets in a tree structure. A project is essentially just a folder that contains zero or more subfolders, called subprojects. In the Project Tree View, the root project and the subprojects are each
represented by a folder icon, followed by the specifier-defined name of that (sub)project. The root project and each subproject contains zero or more alphabets and zero or more properties. In the Project Tree View, an alphabet is represented by a document icon with an $\alpha$ symbol on it, followed by the specifier-defined name of the alphabet. Similarly, a property is represented by a document icon with a $P$ symbol on it, followed by the specifier-defined name of the property. Each property is associated with exactly one alphabet, and each alphabet is associated with zero or more properties. Specifiers can double-click on the document icon or the specifier-defined name of an alphabet or property to open it and begin editing it. Working with context-sensitive menus, specifiers can rename the root project and can add, rename, move, and delete the subprojects, the alphabets, and the properties.

The Project Tree View has a number of strengths and weaknesses. Its support for creating and moving the subprojects and the alphabets allows specifiers to keep related properties and their alphabets grouped together,
providing a way to keep properties organized and flexibly re-organize them as necessary. For example, during the evaluation described in Chapter 5, we were able to organize the properties in a manner that is similar to the organization of the informal requirements documents. Doing this greatly improved the navigability of the set of properties, especially as the size of the set grew and our understanding of how to organize the set changed. The Project Tree View also provides a way for specifiers to focus on only a subset of the properties in a project, collapsing and expanding various parts of the project tree to show only the properties that are currently of interest. One weakness of the current Project Tree View layout, however, is that within each project, the layout is project- and alphabet-oriented, so to find a property in the project tree, specifiers have to know which project it is in and which alphabet it is associated with. In addition, a property can only be in one project at a time. Possible ways to address these issues are discussed in Chapter 7.

4.4.2 Summary Views

The Summary Views allow specifiers to summarize a set of properties and view that summary from multiple perspectives. Unlike the other views within PROPEL, the Summary Views are a read-only visualization of a project’s information and are designed to provide a quick reference for specifiers. As is shown by the example in Figure 4.4, a Summary View provides a tabular representation of a project’s structure and contents, from a particular perspective. A Summary View is comprised of two separate parts. The top part contains the

---

4For space reasons, not all available columns are shown in this figure.
tabular representation and the bottom part lets specifiers view the comments that they have associated with the root project and with the subprojects, properties, alphabets, and events.

There are five pre-defined Summary Views available in PROPEL: Full, Project, Alphabet, Property, and Event. Each pre-defined Summary View displays similar information, but organizes it around that particular perspective. The Full Summary View displays an entire project’s structure and contents from the perspective of the root project, including all the comments for every part of the structure. The Full Summary View includes the following columns:

- **Project and Comments:** the name of the root project and its associated comments.
- **Subproject and Comments:** the names of the subprojects and their associated comments. The number of subproject-and-comments column pairs correlates to the depth of the project’s hierarchical structure. A parent subproject spans the height of its child alphabets as well as its child subprojects, with the alphabets appearing above the subprojects.
- **Alphabet and Comments:** the names of the alphabets and their associated comments, grouped by project/subproject.
- **Event and Comments:** the names of the events and their associated comments, grouped by alphabet.
- **Property and Comments:** the names of the properties and their associated comments, grouped by alphabet.
- **Behavior:** the name of each of the behavior templates used in the properties that are in the project; each behavior template name shown is associated with the property that is in the same row(s), in the Property column to the behavior template’s left. If there is no behavior template chosen yet for a given property, the word “BLANK” is displayed in place of the name of the behavior template.
- **Behavior Parameters:** the parameters’ names (i.e., A or B), followed by a colon (“:”) and the name of the specifier-defined event associated with each parameter, if there is such an event. If there is no associated event, the word “NOT SET” is displayed in place of the event name.
- **Scope:** the name of each of the scope templates used in the properties in the project; each scope template shown is associated with the property that is in the same row(s), in the Property column to the scope template’s left.
- **Scope Parameters:** the parameters’ names (i.e., START or END), followed by a colon (“:”) and the name of the specifier-defined event associated with each parameter, if there is such an event. If there is no associated event, the word “NOT SET” is displayed in place of the event name.
The remaining Summary Views each display a subset of the information that is given in the Full Summary View, and rearrange that subset in terms of the focus of that particular Summary View. More details about each of the other Summary Views are given in Appendix B.2, including example figures for each one.

The Summary Views have a number of strengths and weaknesses. They allow specifiers to view a summary of a set of properties and abstract away some of the more detailed information, such as the options and their settings, and how those option settings are displayed in the different property and scope views. These Summary Views can be used to explore a project’s structure and contents more easily than via the Project Tree View, since specifiers can focus their exploration based on the type of information that they want to see. For example, if specifiers want to know how a particular event is used in a project, they can look through the Event Summary View to find that event and see, for all the properties associated with its alphabet(s), which properties associate it with their parameters. This exploration is just a visual scan, however, since PROPEL’s Summary Views do not currently support any automatic searches. The tables generated for the Summary Views can sometimes be large and, without search functionality, they can be difficult to navigate. In addition, as mentioned earlier, these Summary Views are read-only, so specifiers must go elsewhere in PROPEL to edit the information. This can sometimes be frustrating, especially when specifiers have to look through the table to know where to go to edit the information that they are viewing. The Summary Views are automatically kept synchronized with the other views in PROPEL, however, which mitigates this read-only limitation somewhat. Chapter 7 includes a discussion of possible improvements to the Summary Views.

4.5 Summary

We have implemented a prototype tool, PROPEL, to support the evaluation of our approach. Although the architecture provides much of the initial flexibility needed to support our research, further work to provide more expressibility will require changes to this architecture. We found that the current support for elucidating a property and its alphabet, and also the support for managing multiple properties, was sufficient for most aspects of the case study evaluations described in Chapter 5, but that a number of usability issues remain. These expressibility and usability issues, and possible ways to address them, are discussed in more detail in Chapter 7. It is also important to note that the implement of PROPEL is covered only briefly here, and a more complete description of it is available in the online user guide [59].
CHAPTER 5
CASE STUDY EVALUATION

5.1 Overview

We performed an evaluation of our approach to determine how effective it is at enabling specifiers to create property specifications that are both precise and understandable. We are interested in learning what the strengths and weaknesses of this approach are when it is applied to real-world problems that involve specifiers who do not have expertise in creating property specifications. To conduct this evaluation, we used PROPEL to specify properties for five case studies in the medical domain. The research question that we intended to address with these case studies is:

**Research Question.** To what extent does our approach address the problem of how to enable specifiers, including those who do not have expertise in property specification formalisms, to develop precise and understandable property specifications?

To guide what types of observations we should make with respect to this research question, we identified two propositions:

**Proposition 1.** It will be possible to use PROPEL to formally specify at least 80% of the properties encountered in these case studies.

**Proposition 2.** PROPEL will enable the domain experts who will be involved in the case studies, and who will not be experts in property specification formalisms, to participate in creating precise and understandable property specifications that reflect the desired behaviors of the medical processes in their domains.

Proposition 1 is intended as a means of evaluating whether our approach, as it is currently implemented, will be able to provide support for formally specifying enough of the properties that we encounter that it would be reasonable to further develop and investigate the approach. Proposition 2 is intended to approximate, within the real-world constraints of working with the people who were the domain experts in our case studies, the core hypothesis that motivated the development of our approach: that this approach to property specification provides valuable guidance even to those who are not experts in property specification formalisms, thus
enabling them to create precise and understandable property specifications that they can be confident are correct.

The empirical method that we used to evaluate our approach is the case study. As a research method, the case study has worked well in situations where strict control over the variables in an environment is not possible. The structure that a case study provides in this type of situation allows for the rigorous support of an argument, even in the absence of formal controls. [254, 267] We have observed that it can be difficult to identify and control the variables involved in property specification, since doing so requires a deep understanding of how people conceive of their properties and how they map those conceptions onto a specification formalism. In addition, we were not able to enforce strict controls on our selection of the domain experts that elucidated the properties with us, on their availability, or on the environments within which we interacted. Given these difficulties, we consider the case study to be a suitable method for evaluating our approach. The case study method also allows us to keep the complex characteristics of real-world domains, so that our evaluation is not limited to a set of property specification exemplars [170] or to the use of “toy” problems that could be tailored to our approach, since both of those types of subject matter tend to a priori simplify aspects of the domains from which they are drawn.

In this chapter, we first discuss our case study methodology and briefly describe the domains and sources of each of the case studies. We then discuss the quantitative results of formally specifying the case studies’ properties using PROPEL. Following that, we give some qualitative observations about our experiences working with the domain experts to identify, clearly state, organize, and formally specify those properties. Finally, we end with a discussion of the threats to the validity of our results and a brief summary of what we have learned from this evaluation.

### 5.2 Methodology

Medical errors are recognized as one of the major causes of death and other undesirable medical outcomes in the United States, and it has been estimated that medical errors are the root cause of approximately 98,000 deaths per year [162]. The Institute of Medicine reported that many of the medical errors that occur in practice are caused or exacerbated by faulty processes and situations that lead medical professionals to make mistakes or to fail to prevent mistakes from occurring [198]. In addition, Wilson et al. [260] did a retrospective survey of 14,000 deaths in Australian hospitals and found that process or communication failures were twice as likely to be a contributing factor than a lack of appropriate medical skills. One reason why such failures occur is that medical processes tend to be complex, involving a wide variety of resources and multiple medical professionals. These processes also tend to be prone to many exceptional behaviors, which existing safety
policies either do not cover, or cover only partially. In addition to work that focuses on medical processes, there is also a growing body of work that indicates that insufficient RE efforts for medical computer systems can contribute to increased patient mortality [20, 87, 121, 152]. Given these findings, it is our hypothesis that medical processes and systems can be better understood and thus improved if the requirements for their use are specified in an unambiguous and understandable way, so that medical errors due to misinterpretation are less likely to occur. We thus consider the medical domain to be a rich source of complex, real-world requirements that are suitable for an evaluation of our approach, and it is our expectation that using PROPEL to formally specify properties that describe such requirements would contribute towards improving patient safety.

The five case studies that we have used for this evaluation were all drawn from the medical domain. Four of the five case studies, Chemotherapy (Chemo), Blood Transfusion (BT), Emergency Department (ED), and Verification of Patient ID (VPID), are part of a larger medical safety project that our research group is participating in. The focus of this larger project is to improve patient safety through the application of a number of Software Engineering techniques. In this project, several different types of information are being elicited in addition to the properties, such as descriptions of the control and data flow in various processes and the resources needed to perform tasks in those processes. The fifth case study, the Defense Blood Standard System (DBSS), was part of a larger software development project to improve a U.S. Department of Defense system that manages a national blood repository by tracking each unit of blood product from its donation to its eventual transfusion or disposal. There are more detailed descriptions of each of these domains in Appendix C.

For each of the case studies, we elicited properties by interviewing domain experts and by examining source documents that domain experts gave us. Our elicitation plan for the four case studies in the larger medical safety project involved starting with process elicitation and then shifting to property elicitation after a suitable period of time. For those four case studies, where we elicited all the informal property specifications primarily by interviewing domain experts, the interviews followed a certain format. One or more computer scientists met with one or more domain experts for 1-2 hours on a weekly basis. In between interview sessions, the computer scientists prepared questions about areas that required more elucidation, based on what was discussed in previous interviews. During the interviews, the domain experts answered questions verbally, by drawing pictures, and occasionally by providing source documents. At the end of each interview session, the whole team (i.e., both the computer scientists and the domain experts) agreed on an outline of a plan for the next meeting.
Our overall elicitation plan for these four case studies was to start with the process elicitation and after a suitable period of time, shift to property elicitation. For the process elicitation activities, we followed this general framework:

- Get an informal description of the process from the domain experts.
  - Try to get access to domain experts who have different areas of expertise.
  - Try to elicit both nominal and exceptional flow in the process.
- Capture the domain experts’ terminology in a glossary.
- As the domain experts describe their processes, ask “why” questions and listen for statements of abstract goals that motivate the tasks being done in the process.
- Describe the process using a formal process language.
- Conduct a review of the formal process description with the domain experts.
- Refine the formal process description based on the domain experts’ feedback.
- Iterate as necessary.

As is discussed in Section 5.4.1, when the computer scientists and the domain experts thought that they had learned enough about each other’s areas of expertise to communicate effectively, the plan was that we would then shift primarily to property elicitation activities. For these activities, we followed this general framework:

- Help the domain experts to identify the abstract goals that motivate their process. Make sure to focus on understanding the goals behind the both the nominal and exceptional flow in the process.
- Help the domain experts to state their abstract goals as informal property specifications.
- Refine the informal property specifications so that they are at a suitable level of abstraction.
- Help the domain experts to refine their terminology and use the glossaries effectively.
- Organize the resulting sets of informal property specifications, as necessary.
- Formally specify properties using PROPEL and try to get the domain experts to use PROPEL directly.
- Iterate as necessary.
Throughout all these process- and property-elicitation activities, we gathered quantitative data and recorded qualitative observations. This quantitative and qualitative information is discussed in more detail in Sections 5.3 and 5.4, respectively. We did not create a specific plan about which types of qualitative observations to record, but we did plan to gather quantitative information about the following things:

- How many informal and formal property specifications were elicited.
- How many informal property specifications can be expressed in PROPEL.
- What is impossible or awkward to express in PROPEL.
- How often each of the scopes, behaviors, and option settings was used.
- How many domain experts were involved, and to what extent, in each aspect of the property elicitation activities.

In addition to the elicitation plan for the case studies where we could interview domain experts, we also developed an elicitation plan for the DBSS study, where we only did property elicitation. We had very limited access to the domain experts in this study. Our elicitation plan was thus to obtain the properties primarily from examining source documents that the domain experts gave us. We examined these documents with an eye towards identifying possible abstract goals behind the processes that the source documents described. For the limited period in which we had access to the domain experts, we followed the general framework described previously for property elicitation, and we were briefly able to get their input on refining and formalizing the property specifications, iterating only a couple times on a subset of the specifications. Throughout the course of this case study, we also gathered the same quantitative and qualitative information that was gathered in the other four case studies.

Before we discuss the results of this evaluation in detail, however, we should clarify what is meant when we refer to “informal property specifications” in this chapter. This term is used to refer to chunks of natural-language text that were given a unique ID in the informal property specification documents for each of the case studies (see Appendix C). It is important to note, however, that there are chunks of natural-language text in the informal property specification documents that are given unique IDs but are not treated as informal property specifications by our count. Unique IDs were assigned to those chunks of text for convenience, because use of the IDs made the documents easier to navigate and understand. The four types of text that were given unique IDs but are not considered informal property specifications are:

- Text that describes a set of properties at a higher level of abstraction than the level that is being used for the property specifications. Such text provides a high-level summary of a set of related properties.
5.3 Quantitative Results

5.3.1 Overview

We encountered a total of 145 informal property specifications in the five case studies and we were able to use PROPEL to formally specify 95% of them. We elicited most of these informal property specifications during interviews with the domain experts, rather than from inspecting source documents. For most of the properties, a computer scientist created the first drafts of the formal specifications and then conducted further interviews with the domain experts to confirm or improve the details of those drafts; for the most part, the domain experts did not work directly with PROPEL. During the course of these case studies, we encountered some limitations on PROPEL’s ability to express certain concepts that were in the informal property specifications. 5% of the properties cannot be formally specified in the current version of PROPEL and, of those properties that can be expressed, 56% of them would probably be expressed more effectively if PROPEL supported additional types of information. It should also be noted that, due to the fact that our approach limits each property specification to restricting a very focused subset of system behavior, a single informal property specification often mapped to multiple formal property specifications in PROPEL. When we look at the resulting body of formal property specifications for these case studies, we can make some observations about their distribution. We found some interesting differences between these case study results and the results of the pattern survey that was conducted by Dwyer, Avrunin, and Corbett [82]. In addition, we observed certain trends in the option settings’ frequencies. All these issues are discussed in detail in the remainder of this section.

Table 5.1 shows a high-level summary of the results of developing formal property specifications for the five case studies. For each case study, Table 5.1 gives the name of the medical domain, the total number of informal property specifications, how many of those informal property specifications can be completely
Table 5.1. Property Expressibility Summary

<table>
<thead>
<tr>
<th>Case Study</th>
<th>Total # of Informal Properties</th>
<th>Completely Expressible in PROPEL</th>
<th>Awkward in PROPEL</th>
<th>Percent Covered</th>
</tr>
</thead>
<tbody>
<tr>
<td>BT</td>
<td>44</td>
<td>28</td>
<td>14</td>
<td>95%</td>
</tr>
<tr>
<td>Chemo</td>
<td>59</td>
<td>25</td>
<td>34</td>
<td>100%</td>
</tr>
<tr>
<td>VPID</td>
<td>5</td>
<td>0</td>
<td>5</td>
<td>100%</td>
</tr>
<tr>
<td>DBSS</td>
<td>11</td>
<td>0</td>
<td>11</td>
<td>100%</td>
</tr>
<tr>
<td>ED</td>
<td>26</td>
<td>8</td>
<td>13</td>
<td>81%</td>
</tr>
<tr>
<td>TOTAL</td>
<td>145</td>
<td>61</td>
<td>77</td>
<td>95%</td>
</tr>
</tbody>
</table>

expressed formally in the current implementation of PROPEL, and how many are expressed in only a limited way in PROPEL, since they refer to types of information that probably could be handled more effectively if additional expressibility were available. We describe what those types of information are in more detail in Section 5.3.2, but for now we can summarize the data in Table 5.1 by noting that we were able to use PROPEL to create formal specifications of 95% of the informal property specifications that we encountered in these case studies. This result differs from our preliminary results [60] for two main reasons. One reason is that the focus of each of the case studies shifted over time. As the domain experts grew in their understanding of how to elucidate the properties with us, the informal property specifications—and thus the formal property specifications—changed in some important ways. Many new properties and a new case study were introduced, and a few of the properties that were included in the preliminary data were dropped and are thus not included in Table 5.1. The other reason for the difference is that our understanding of how to formally specify the properties changed as our experience grew. Most of the formal property specifications that remain from that preliminary work were revised as we learned more about the case studies’ domains and we learned to avoid some elicitation pitfalls, which are discussed in more detail in Section 5.4.

Table 5.2 provides a summary that compares the five case studies in terms of how we elicited the informal property specifications, how many domain experts were involved, who created the formal property specifications, to what extent at least one domain expert was available to vet those informal and formal property specifications, and how many informal and formal property specifications were created for each case study. For most of the properties, a computer scientist created the first drafts of the formal specifications and then had the domain experts vet those drafts; the domain experts usually did not work directly with PROPEL. There is a discussion in Section 5.4 of why this was the case. The domain experts only vetted the final versions of the property specifications in four of the five case studies: the ED study was the only one without that external review. While domain experts were involved in early versions of the informal property specifications for all five case studies, the domain experts in the DBSS and ED studies were not available to vet most of the final property specifications that are included in Appendix C. In addition, it should be noted that the domain
### Table 5.2. Case Studies’ Property Elucidation Summary

<table>
<thead>
<tr>
<th>Case Study</th>
<th>Elicitation Source</th>
<th>Domain Experts (DEs)</th>
<th>Who Formalized</th>
<th>DE Vetted</th>
<th>Total Informal</th>
<th>Total Formal</th>
</tr>
</thead>
<tbody>
<tr>
<td>BT</td>
<td>interviews with DE, document inspection</td>
<td>1 Nursing Ph.D.</td>
<td>mostly done by computer scientists</td>
<td>all informal and formal specifications</td>
<td>44</td>
<td>75</td>
</tr>
<tr>
<td>Chemo</td>
<td>interviews with DEs</td>
<td>1 M.D., 1 Pharm.D., 1 R.N., 1 M.A.</td>
<td>mostly done by computer scientists</td>
<td>all informal and formal specifications</td>
<td>59</td>
<td>119</td>
</tr>
<tr>
<td>VPID</td>
<td>interviews with DE</td>
<td>1 Nursing Ph.D.</td>
<td>mostly done by computer scientists</td>
<td>all informal and formal specifications</td>
<td>5</td>
<td>7</td>
</tr>
<tr>
<td>DBSS</td>
<td>interviews with DEs, document inspection</td>
<td>1 software development manager, 2 laboratory technicians</td>
<td>manager created 3 specifications in PROPEL; rest done by computer scientists</td>
<td>no informal specifications, 3 formal specifications</td>
<td>11</td>
<td>22</td>
</tr>
<tr>
<td>ED</td>
<td>interviews with DE</td>
<td>1 M.D.</td>
<td>all done by computer scientists</td>
<td>no informal or formal specifications</td>
<td>26</td>
<td>27</td>
</tr>
<tr>
<td><strong>TOTAL</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>145</td>
<td>250</td>
</tr>
</tbody>
</table>
experts’ vetting of the three formal property specifications in the DBSS study was only partial. Although the scopes, behaviors, and option settings remain the same as what the software development manager in that study chose for those three formal property specifications, the wording of the event names in those specifications has since been changed to clarify their meaning and to be more consistent with similar event names in other properties. Section 5.4.4 goes into more detail on the issues involved in how the domain experts vetted the formal property specifications.

5.3.2 Limitations on PROPEL’s expressibility

5.3.2.1 Concepts that were not expressible

Returning to an examination of the data in Table 5.1, we can see that it indicates that 95% of the properties that we encountered in these case studies can be expressed in the limited set of scopes and behaviors that is available in the current implementation of PROPEL, but that 7 of the properties are not considered expressible. There are two types of information that characterize those inexpressible properties. Most of the inexpressible properties (6 of the 7) refer to multivariate state information, where the conjunction of more than one variable’s state is the condition that defines when a behavior has to be enforced. Such information requires the use of nested scopes to be expressed in the event-based framework that PROPEL uses. PROPEL does not currently support nested scopes, however, and thus these properties cannot be formally specified in the tool. The remaining inexpressible property refers to an ordered sequence of events that act as a trigger to require another event to occur subsequently. This property is considered a 3-1 Chain Response in the pattern system [81], but it cannot be expressed in the current implementation of PROPEL.

We also encountered two other types of inexpressible information in earlier versions of some of the Chemo properties. One such type of information was a conjunction of two or more events that, if all those events occurred, could act as a trigger to require another event to occur subsequently. These properties were not versions of a Chain Response, however, since the trigger events were not required to occur in any particular order. These properties were more likely an instance of one of the Composite Propositions described by Mondragon and Gates [204]. In addition to the event conjunctions, we also observed an event bound in one of the properties, where an event was not allowed to happen more than a certain number of times. For example, the Chemo.D.3 property (see Appendix C.2) states that a patient must have appropriate I.V. access before chemotherapy drugs can be administered. An earlier, more detailed version of this property focused on how I.V. access is achieved. In particular, the property stated that there is a maximum bound on the number of times that the nurses are allowed to attempt to gain access to a vein before they are required to stop trying and to discharge the patient without administering any drugs. Since those earlier versions of the properties
Table 5.3. Frequencies of the Problematic Issues

<table>
<thead>
<tr>
<th>Case Study</th>
<th>Real-time</th>
<th>Event Disjunction</th>
<th>Simple State</th>
<th>Allow &amp; Prohibit SEs</th>
<th>Not all SEs known</th>
</tr>
</thead>
<tbody>
<tr>
<td>BT</td>
<td>3</td>
<td>4</td>
<td>0</td>
<td>7</td>
<td>9</td>
</tr>
<tr>
<td>Chemo</td>
<td>0</td>
<td>7</td>
<td>31</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>VPID</td>
<td>0</td>
<td>5</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>DBSS</td>
<td>5</td>
<td>9</td>
<td>2</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>ED</td>
<td>1</td>
<td>11</td>
<td>2</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>TOTAL</td>
<td>9</td>
<td>36</td>
<td>35</td>
<td>7</td>
<td>10</td>
</tr>
<tr>
<td>%</td>
<td>12%</td>
<td>47%</td>
<td>45%</td>
<td>9%</td>
<td>13%</td>
</tr>
</tbody>
</table>

were set aside by the domain experts, however (see Section 5.4 for a discussion of why this occurred), they are not included in the data in Table 5.1.

Though we observed those two particular types of information showing up only in the more detailed versions of the property specifications, we did not find in general that properties informally specified at a more detailed level of abstraction were more difficult to express in PROPEL than properties informally specified at a higher level of abstraction. We did observe that some types of information seemed to show up more frequently at the more detailed levels of abstraction, however. One involved the concept of resources, which are any physical or virtual objects, including humans and automated systems, that are needed to complete a task in the given domain. We saw that there was a greater variety of resources at the more detailed levels, both in the number of types and in the number of instances. This does not make the properties at those more detailed levels of abstraction necessarily more difficult to specify, but it does imply the need for more instance disambiguation and more events that are parameterized by the set of resource types that could be involved in those events. For example, instead of just a single “verify chemotherapy orders” event, there would be several events, one for each medical professional who could do that task: “Pharmacist verifies chemotherapy orders,” “Clinic RN verifies chemotherapy orders,” etc. We also saw that when the properties were specified at a more detailed level of abstraction, there tended to be more exceptions to when the properties had to hold. For example, there would be more properties of the form “B cannot occur until after A occurs, unless C occurs.” Such exceptions do not necessarily imply that the property is not expressible in PROPEL, however. These exceptions can usually be expressed as prohibited secondary events or they can be expressed with either a Before scope (for exceptional events that did not repeat within the domain boundaries) or a Between scope (for exceptional events that can occur repeatedly within the domain boundaries).

5.3.2.2 Issues that can probably be handled better

In addition to the properties that could not be expressed in the current implementation of PROPEL, 56% of the properties that were formally specified in PROPEL could only be expressed in a limited way. That
is, the formal property specifications were detailed enough to do the kinds of analyses that were part of the larger medical safety project that we were participating in, but the current implementation of PROPEL was not designed with these types of properties in mind, so there were some awkwardnesses in their formal specifications. Recall that Table 5.1 gives the total number of these properties in each case study. Here, Table 5.3 shows the breakdown of these properties amongst the five different types of problematic information. For each case study, this table shows the name of the medical domain and the number of properties that refer to each type of problematic information.¹

5.3.2.2.1 Event disjunctions. In the case studies, the most common type of information that can be awkward to specify in PROPEL is an event disjunction. In an event disjunction, there is some abstract event that maps to a disjunction of more detailed events, and if any one of the more detailed events in the disjunction occur, the abstract event is understood as having occurred. While event disjunctions pose no theoretical difficulties for the scopes and behaviors in PROPEL, the tool has no explicit support for them yet. Thus, they can only be specified by putting a description of the event disjunction in the comments that are associated with the events and the properties, or by naming an event in a compound fashion, with the keyword “OR” inserted between the individual event names. For example, consider the ED.D.2 property’s informal specification:

*Before a Patient can be discharged from the Emergency Department, that Patient’s vital signs must be checked, unless that Patient leaves the Emergency Department against medical advice or without notifying a medical professional.*

This property is specified using an event disjunction to capture the information in the “unless” clause: a single event in the formal specification was named “Patient leaves Emergency Department against medical advice OR Patient leaves Emergency Department without notifying a medical professional.” See Appendix C.3 for the formal specification of this property. Better support for event disjunctions in PROPEL would allow them to be specified as a set of individual events and would preserve the real semantics of event disjunction.

5.3.2.2.2 Single-Variable state information. Another type of information that can be awkward to specify in PROPEL is state information. The only way to capture state information in the event-based paradigm that PROPEL uses is to create events that represent changes in a variable’s state and then use a Between scope to define when the variable is in a desirable state and enforce a behavior only as necessary, using that information. For example, consider the Chemo.C.10 property:

---

¹The percentages in Table 5.3 do not sum to 100% because some properties can be characterized by multiple types of problematic information.
No treatment plan that is based on a Patient’s stale or disparate data can be created for that Patient.

This property is specified using two events, “Patient data in treatment plan becomes problematic” and “resolve problematic Patient data in the treatment plan” to represent the state of the data. See Appendix C.2 for the full formal specification of this property. Given that this event-based representation of state requires a Between scope for each variable, more complex state information that refers to two or more variables’ states in conjunction cannot be specified in the current implementation of PROPEL, because it does not yet offer support for nesting scopes.

5.3.2.2.3 Two limitations on secondary events. We also encountered two limitations when trying to specify secondary events in PROPEL: how to specify both prohibited and allowed secondary events, and how to handle unknown secondary events. These issues would often show up when a domain expert said that all events except for the ones in a given set were prohibited from occurring between two primary events. In this situation it is necessary to specify both the (relatively small) set of allowed events and also the set of all the rest of the events that could occur in that domain, which are prohibited. In such cases, only the allowed events are explicitly specified in PROPEL. The prohibited events were just described in the property’s comments and were not formally specified as prohibited. Although it would be possible to use a separate property with a Between Absence scope to prohibit all the other events, we encountered a second limitation on specifying secondary events in PROPEL that prevented the use of Between Absence from being a viable approach. In several of the properties we had to explicitly enumerate the set of prohibited events, but we did not know what they all were in concrete terms. For example, consider the BT.D.5 property:

After receiving a blood bank or physician order to obtain a specimen from a Patient and immediately before obtaining any specimen from that Patient, it must be made sure that the first and last name and medical record number on that Patient’s ID band match the first and last name and medical record number on the specimen container label. Every event in the transfusion process (except for applying a label to a specimen container) is prohibited from occurring between making sure that the ID band and specimen container label match and obtaining the specimen.

It is not possible to use the current implementation of PROPEL to formally specify the part of this property that states that every other event in the transfusion process is prohibited from occurring at a particular point, because there is no explicit enumeration of that set of prohibited events available. See Appendix C.1 for the formal specification of this property. For other properties where there are only prohibited secondary events, we try to specify the concept of “every secondary event in the domain is prohibited” by creating a single
event named “*” (asterisk) that is intended to represent the set of all possible events in that case study’s domain (excluding the events associated with the property’s parameters). This workaround is not sufficient to handle the problem with having both allowed and prohibited secondary events, however, because using the * event would mean erroneously including the set of allowed events in the set of prohibited events and vice versa. We consider this limitation on the secondary events to lead to formal property specifications that are just awkwardly specified rather than not expressible in PROPEL, because once all the events in a case study’s domain are known, then this limitation can be bypassed by an explicit enumeration of both the prohibited and the allowed events, specified in separate properties.

5.3.2.2.4 Real-time information. Another type of information that is problematic to specify in PROPEL is a reference to real-time information. Since PROPEL uses an event-based framework without real-time support, we represent this type of information with events that are checks for whether deadlines have passed. For example, consider the DBSS.B.2 property:

_If a Patient’s blood specimen has a positive antibody screen, if that Patient has a history of clinically-significant red cell antibodies or of ABO / Rh incompatibility, or if that Patient’s history is unknown, then if that Patient’s blood specimen is more than 72 hours old, an up-to-date blood specimen must be obtained from that Patient before an ABO / Rh identification and antibody screen can be performed for that Patient._

The real-time information in this property is specified with an event named “find that blood specimen is more than 72 hours old.” See Appendix C.5 for the formal specification of this property. Such a property specification is sufficient for the purposes of the analyses used in the larger medical safety project that we participated in, but other specification approaches that address these real-time concepts more directly, such as the one described in [165], might be preferable in the long term.

5.3.2.3 Parameterization

There was one other issue that is related to property expressibility in PROPEL, but it is not clear that the formal property specifications that include it are necessarily written in an awkward way. These are properties that need to be parameterized to disambiguate between multiple instances of the same type of resource. For example, consider the BT.B.9 property:

_The infusion of a unit of blood product must begin within 30 minutes of the unit of blood product being picked up from the blood bank._

This property exists in the context of one patient who might have multiple units of blood product infused during the course of fulfilling one physician order for a blood transfusion. In this situation, there is an infusion
deadline for each unit of blood product, and while the deadline for one unit might expire, that does not imply that all of the units of blood product that have been picked up from the blood bank are no longer suitable for infusion. Thus, the deadline for each instance of a unit of blood product has to be disambiguated from all the others. For this BT.B.9 property, the events are named “pick up unit of blood product \(i\) from blood bank” and “30-minute deadline passes for unit of blood product \(i\),” where \(i\) is an instance-disambiguation parameter that can take any value that is present in the space of possible unique identifiers for a unit of blood product. This kind of parameter\(^2\) is meant to disambiguate resources that are instances of the same type, and within a given property’s alphabet, all events that use the same instance-disambiguation parameter in their names (such as “\(i\)”) are meant to refer to the same instance of that type.

It should be noted, however, that not all properties that refer to resources necessarily require instance disambiguation. If a property’s alphabet contains only one event that refers to a particular resource, then that event does not need to be parameterized, because there is no possibility for confusion between instances within the context of the property. For example, consider the BT.B.4.1 property:

*Before infusing each unit of blood product into a Patient, it must be confirmed that that Patient has exactly one ID band.*

In this property, the restriction can be enforced for each unit of blood product by specifying the property so that one confirmation can only enable one infusion to occur.

Both of these examples were drawn from the BT study, and with 51% of the parameterized properties, the BT study contains more than any other study, by far. This is due to two factors: the BT study is the second-largest case study in terms of the number of informal property specifications that it contains, and even when all the properties only concern a single patient, the domain still includes multiple instances of the same types of resources. For example, in a given blood transfusion, multiple units of blood product can be given, multiple specimens can be obtained, and each instance of both of those types of resources must be disambiguated from the others. By contrast, the VPID study does not have any parameterized properties: it concerns the verification of only one ID band and thus there is no need for multiple instances to be disambiguated.

### 5.3.3 Informal-to-formal mappings

There were two different types of property specifications that were elucidated in these case studies, informal and formal. As was described in Section 5.2, we regard the informal property specifications to be those expressed in arbitrary natural language and the formal property specifications to be those expressed

\(^2\)The instance-disambiguation parameter described here should not be confused with the kind of parameter that is described in Chapter 3.
Table 5.4. Types of Informal-to-Formal Property Specification Mappings

<table>
<thead>
<tr>
<th>Case Study</th>
<th>Resource-based</th>
<th>State-based</th>
<th>Ordered AND</th>
<th>Scope Constraints</th>
</tr>
</thead>
<tbody>
<tr>
<td>BT</td>
<td>15</td>
<td>0</td>
<td>10</td>
<td>3</td>
</tr>
<tr>
<td>Chemo</td>
<td>13</td>
<td>18</td>
<td>4</td>
<td>9</td>
</tr>
<tr>
<td>VPID</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>DBSS</td>
<td>6</td>
<td>4</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>ED</td>
<td>0</td>
<td>0</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>TOTAL</td>
<td>34</td>
<td>22</td>
<td>18</td>
<td>13</td>
</tr>
<tr>
<td>%</td>
<td>49%</td>
<td>31%</td>
<td>26%</td>
<td>19%</td>
</tr>
</tbody>
</table>

using PROPEL. For every informal specification that can be expressed in PROPEL, there is at least one formal specification. In addition, 51% of those informal specifications map to a conjunction of two or more formal specifications. Recall that besides a summary of the domain experts’ involvement, Table 5.2 also gives the total number of informal and formal property specifications in each case study. As that table shows, the BT, Chemo, and DBSS case studies have about twice as many formal property specifications than informal ones, and indeed, the majority of the 1-to-many mappings occur in only those three case studies. These differences between the case studies are due to the differing prevalence of four types of 1-to-many property mappings: multiple resources, single-variable state information, ordered AND decompositions, and scope constraints. Table 5.4 shows how often each of those types of mappings occur in the case studies.\(^3\)

5.3.3.1 Multiple resources

The most common reason in the case studies for mapping a single informal property specification to multiple formal specifications is that the informal specification refers to multiple resources, and each of those resources requires a separate formal property specification. For example, the Chemotherapy and BT case studies include properties where there are identical restrictions on how checks are done for different resources. For example, the Chemo.D.2 property refers to two different kinds of treatment that can be administered to a patient:

\[\text{Before pre-chemotherapy supportive care medications or chemotherapy can be administered to a Patient, that Patient must be well enough to receive treatment.}\]

This aspect of the property is formally specified as two\(^4\) separate properties (see Chemo.D.2a1 and Chemo.D.2b1 in Appendix C.2), which only differ by the kind of treatment the events refer to. Though

\(^3\)Because some informal-to-formal property mappings can be characterized by multiple types of information, the totals for each type of mapping in Table 5.4 sum to more than the total number of informal-to-formal property mappings, and similarly, the percentages sum to more than 100%.

\(^4\)Since this property has both a reference to multiple resources and a reference to single-variable state information (see Section 5.3.3.2), the informal specification of this property actually maps to a total of four formal specifications in PROPEL.
this example of a reference to multiple resources shows a mapping to only two formal property specifications, several of the resource-based mappings are to three or more formal property specifications. We only observed these resource-based mappings in the Chemotherapy, BT, and DBSS case studies, but we suspect that there would be examples of this type of mapping in the ED domain as well, if the properties in that case study were to be specified at a more detailed level of abstraction.

5.3.3.2 Single-variable state information

Another common reason in the case studies for mapping a single informal property specification to multiple formal specifications is the need to express single-variable state information within the limitations of PROPEL. There are two different types of references to such state information in the case studies, but only one requires a mapping from one informal specification to multiple formal specifications. The type of reference that can be expressed with a 1-to-1 mapping is where the behavior only has to hold if a variable is in a certain state. The other type of reference to single-variable state information is where a variable has to be in a certain state before some event is allowed to occur, and two formal property specifications are needed to express this in PROPEL. For example, consider the Chemo.D.2 property from the previous section again, and this time note that neither type of treatment can be administered unless it is safe to do so. For each type of treatment, two formal specifications must be created. One is designed to ensure that the state of the world becomes safe at least once before the treatment is performed (e.g., Chemo.D.2a1 in Appendix C.2), and the other is designed to ensure that each time the state of the world becomes unsafe, it has to become safe again before the treatment is performed (e.g., Chemo.D.2a2 in Appendix C.2).

It is interesting to note that this type of reference to state information only appears in the DBSS and Chemotherapy studies, the latter of which has it in 31% of its informal property specifications. This type of reference is common in that study because when we agreed on a level of abstraction with the domain experts, we made the decision to avoid writing the informal property specifications in terms of events and to focus instead on the relevant state of the world. We made this choice so that the focus of the properties is on the real safety issues and not just on whether the medical professionals involved notice problems, which would have been the focus of the equivalent events. This issue is discussed in more detail in Section 5.4.3.

5.3.3.3 Ordered AND decompositions

Properties that are organized by an ordered AND decomposition (see Section 5.4.5.1.1 for a more detailed discussion of decompositions) also need to map from a single informal property specification to multiple formal specifications. This is because, given the current limitations of PROPEL, if an informal specification refers to \( N \) events that all must happen in a particular order, it has to be formally specified using \( N - 1 \)
properties with Precedence behaviors to enforce each pairwise ordering of those events (when \( N \geq 2 \)). The behavior that is being decomposed must also be considered, however. There are two different types of ordered AND decompositions that appear in the case studies. There are 13 Precedence-based decompositions and 5 Response-based decompositions. The Precedence-based ordered AND decompositions require that a series of events precede one guarded event, and the Response-based ordered AND decompositions require that after a single event occurs, a series of events are required to follow that trigger event. These ordered AND decompositions are examples of \( N-1 \) Chain Precedence and \( 1-N \) Chain Response behaviors, respectively, which are described in the pattern system [81]. Using that system, an ordered AND decomposition could possibly be formally specified using a single property with a Chaining behavior. Since PROPEL does not yet support Chaining behaviors, however, multiple formal property specifications are required to enforce the pairwise ordering of the events, in addition to the specification of the Response or Precedence behavior involved.

The informal Precedence-based ordered AND decompositions that appear in the case studies each actually map to \( 2N-1 \) formal specifications. As mentioned previously, \( N-1 \) Precedence properties are required to enforce the pairwise orderings, but we also need an additional \( N \) Precedence properties to enforce the relationship between each of the events in the decomposition and the guarded event. Although some of these formal property specifications appear to be redundant, there is still reason to keep them. Consider, for example, a Precedence-based ordered AND decomposition that is of the form:

Before event C can occur:

1. A must occur;
2. B must occur;
3. the events described above must occur in that order.

According to the method we use to specify the Precedence-based ordered AND decompositions, we would create the following three formal property specifications (these are summarized descriptions):

1. A must precede C.
2. B must precede C.
3. A must precede B.

Logically, property 1 is unnecessary because, by transitivity, the conjunction of property 2 and property 3 implies property 1. Despite the fact that one of the formal specifications is redundant by this logic, there are two reasons for still including it. One reason is that it may be the case that there are subtleties in the
relationship between A and C that need to be expressed and the transitivity does not necessarily cover all the restrictions on the interactions between the two events. Another reason to keep the redundant formal specifications is to avoid losing information if the universe of events referred to in those properties were ever to change. For example, if B were to be removed from the universe of events and property 1 were not present in the set of formal specifications, the fact that A must precede C might be overlooked. It is likely that this whole issue of redundancy would disappear if the Precedence-based ordered AND decomposition could be more concisely expressed with a Chain Precedence behavior. This issue is discussed briefly in Chapter 7.

There is one more interesting observation to make about the prevalence of the ordered AND decomposition in the case studies. In terms of the total number of informal property specifications that refer to it, the ordered AND decomposition is not the most common reason in the case studies for mapping one informal property specification to multiple formal specifications. It is interesting to note, however, that it is the reason that shows up in the most case studies (4 of the 5). The only study that does not have any ordered AND decompositions is the DBSS study.

5.3.3.4 Restrictions on scope delimiters

The least common reason in the case studies for mapping a single informal property specification to multiple formal specifications is a reference to restrictions on the scope delimiters. The scope delimiters are restricted in situations where the domain experts understand certain assumptions about practical constraints in their domains, and such assumptions have an impact on the correct interpretation of the scopes that are chosen for the formal property specifications. In these situations, this 1-to-many mapping is required because PROPEL does not support restrictions on the scope delimiters, per se. Those restrictions must instead be specified in a separate property from the one that is focused on the main point of the informal property specification. These separate properties are necessary whenever the main property has a Between scope where the end delimiter is required to follow an occurrence of the start delimiter. Such restrictions on the scope delimiters usually don’t warrant a separate informal specification, however, because it is information about the domain that can be safely assumed as common knowledge, such as the fact that once a real-time countdown begins, that countdown will eventually stop.

An example of a restriction on the scope delimiters is the formal specification of the Chemo.F.6 property. The informal specification of that property is:

After chemotherapy has been administered to a Patient, all of that Patient’s laboratory results and all of that Patient’s chemotherapy administration data must be entered into that Patient’s record on the same business day.
This property is formally specified as two separate properties. The main property, Chemo.F.6a, is given in Appendix C.2, and this property enforces the repeatability of recording the patient information via a Between scope, with the “administer chemotherapy” event as the start delimiter and the “business day ends” event as the end delimiter. There are two additional aspects of this property that the domain experts knew to be true: chemotherapy is only allowed to be administered to a patient after the start of a business day and before the end of it (so an administration of chemotherapy would imply that the business day had already started), and once a business day begins, it must eventually end. Thus, after “administer chemotherapy” occurs, “business day ends” is eventually required to occur. This constraint on the scope delimiters is formally specified in a second property, Chemo.F.6b, which is given in Appendix C.2.

5.3.4 Observations about the distribution of properties

The distribution of the behavior and scope frequencies in our case studies is quite different from the distribution in the property pattern survey [82]. The differences in the distributions are likely due to a number of factors. Some differences may be influenced by the nature of the sources: the property pattern survey covered property specifications for software systems that were described in the finite-state verification literature, while our case study properties are drawn from real-world medical processes. There are also differences in the breadths of the sources: the pattern survey was drawn from many independent sources in a wide array of different domains, while the case studies are drawn from five sources, four of which involved domain experts who were part of a single institution. Another reason for differences might be that only four of the eight behaviors in the pattern system are supported by PROPEL. In fact, 21% of the properties in the pattern survey were covered by a behavior that is not supported by PROPEL. It is interesting to note that while the distribution of the scope and behavior frequencies is quite different between the pattern survey and the case studies, PROPEL is still able to express a high percentage (95%) of the properties that we encountered in the case studies. Similarly, the pattern system [81], with all eight of its behaviors and the same basic types of scopes, was able to cover 92% of the properties that Dwyer et al. encountered.

5.3.4.1 Differences in scope frequencies

Figure 5.1 shows the relative frequencies of the four types of scopes\(^5\) for the pattern survey and the case studies, in descending order of their frequencies in the case studies. We found that only two of the scopes in the case studies, Between and Global, accounted for 94% of all the scopes in the formal property specifications. Although the Between and Global scopes were also the two most common scopes in the

\(^5\)Note that two scopes in the pattern survey, there called “Between Q and R” and “After Q Until R,” are variations on the Between scope in PROPEL, and are thus combined and counted as “Between” in Figure 5.1.
Figure 5.1. Comparison of Case Study vs. Pattern Survey Scope Frequencies

pattern survey, the difference in the Between scope between the pattern survey results and the case study results is more striking. It was expected that we would encounter many Global scopes in our case studies, so the fact that Global remained in the top two most common scopes was not surprising. The Between scope was only marginally more common than the Before and After scopes in the pattern survey however, so the significant increase in the prevalence of Between in the case studies is of more interest in this discussion.

There are two main factors that contributed to the frequency of the Between scope in the case studies. One factor is that most of the properties in the case studies require some type of repeatability. There are two ways to enforce repeatability in the set of scopes and behaviors that PROPEL currently supports: repeatability can be enforced if a property’s behavior is either Response or Precedence, and repeatability can be enforced if a property’s scope is Between. In cases where the repeatability is needed but cannot be enforced in the behavior, a Between scope is necessary. Part of the need for repeatability in the case studies is due to the focus on medical processes, since most of the safety checks that guard interactions with a patient are required to be repeated each time an intervention for a patient is performed. For example, the domain boundaries for the Chemotherapy study are defined by one course of chemotherapy. Since one course can contain one or more cycles of chemotherapy, and since each cycle can contain one or more administration episodes, many of the properties use a Between scope so that repeatability can be enforced for each administration episode. In addition to the focus on medical processes, it is also necessary to use a Between scope to enforce repeatability in the properties for an ordered AND decomposition. In all instances of an ordered AND decomposition in the case studies, there are no restrictions on how many times the individual ordered events can occur. In this situation, an ordering property’s behavior cannot by itself repeatedly enforce the ordering each time a new intervention for a patient is required, and thus a Between scope is necessary to enforce the repetition. An example of this type of property is given in Section 5.4.4, and this issue is discussed in more detail there.
The other factor that contributed to the frequency of the Between scope is the number of properties that refer to single-variable state information. To represent state changes for a single variable in PROPEL’s event-based paradigm, we must use a Between scope to express when that variable is in a desirable (or undesirable) state, at which point the appropriate behavior can be enforced. For example, consider the informal specification of the ED.D.3 property:

*Before a Patient can be discharged from the Emergency Department, an MD must evaluate that Patient, unless that Patient leaves the Emergency Department against medical advice or without notifying a medical professional.*

This property is specified using a Between scope to represent the state of the patient: as long as the patient is in the ED and has not left against medical advice or without notifying a medical professional, then the behavior (i.e., that the patient cannot be discharged from the ED until after an MD had evaluated them), should be enforced. The formal specification of ED.D.3 is given in Appendix C.3. While properties that refer to state information show up a few times in the DBSS and ED studies, the Chemotherapy study contributed 89% of the total number of these properties. This is because the change in the level of abstraction at which the properties were specified in this case study prompted us focus many of the properties in the Chemotherapy study on state information. This issue is discussed in Sections 5.3.3 and 5.4.3 in more detail.

### 5.3.4.2 Differences in behavior frequencies

In addition to significant differences in the relative scope frequencies between the case study results and the pattern survey results, there were also noticeable differences in the relative behavior frequencies, as is shown in Figure 5.2.6 This figure shows the relative frequencies of the four types of behaviors that are

---

6The pattern survey percentages in this graph do not sum to 100% because that work used several additional behaviors that are not shown here.
supported by PROPEL, in descending order of the frequencies of the behaviors in the case studies. In the pattern survey, the most common behavior was Response, and Absence was the next most common. By contrast, the most common behavior in the case studies is Precedence, followed by Existence. Similar to the reasons given for the scope differences, the differences in the relative frequencies of these behaviors might be due to the different sources. Recall that the pattern survey covered property specifications for software systems that were described in the finite-state verification literature, while our case study properties are drawn from real-world medical processes. For example, we observed that many of the medical properties are of the form “a safety check must be completed successfully before a medical intervention can be performed for a patient.” This type of restriction on the occurrences of events in the medical domain can often be captured using a Precedence behavior.

Another reason that the Precedence behavior is more common in the case studies is that to enforce the ordering of $N$ events in an ordered AND decomposition, $N - 1$ properties with a Precedence behavior are required. The reason for this is discussed in more detail in Section 5.3.3. In the pattern survey, the use of the N-1 Chain Precedence behavior to express this type of ordered AND decomposition means that if such decompositions were encountered in the pattern survey, significantly fewer Precedence-based properties were needed. To take this into account in our comparison, if we suppose that PROPEL could support some form of N-1 Chain Precedence and some form of 1-N Chain Response, and that those forms could express the ordered AND decompositions that we encountered in the case studies, then the behavior frequency comparison would be noticeably different. Figure 5.3\(^7\) shows the relative frequencies of the six types of behaviors that are

\(^7\)The case studies’ percentages in this graph do not sum to 100% because of rounding imprecision.
supported by this hypothetical PROPEL, taking those two Chaining behaviors into account, in descending order of the frequencies of the behaviors in the case studies. There were 14 Response-based properties and 89 Precedence-based properties involved in the ordered AND decompositions in the case studies. 44 of those Precedence-based properties were used solely for enforcing the order of the chained events. Barring appropriate scopes, those 44 Precedence-based properties are all potentially too restrictive, since the chained events might not be required to occur in the specified order outside of the context of their Chaining behaviors, especially if there are exceptional conditions that have not been taken into account. If the Chaining behaviors were supported, those 14 Response-based properties and 89 Precedence-based properties would be replaced by 5 1-N Chain Response-based properties and 13 N-1 Chain Precedence-based properties. As can be seen in Figure 5.3, including the Chaining behaviors significantly decreases the percentage of Precedence-based properties in the case studies. It also actually slightly increases the percentage of Response-based properties because the total number of properties has dropped from 250 to 165. For similar reasons, although the number of Existence-based properties remains the same, including the Chaining behaviors increases the percentage of Existence-based properties such that they become the most common behavior in the case studies.

This is consistent with our original findings, because if we return to Figure 5.2 and the four behaviors that PROPEL actually supports, we see that the Existence behavior is the second most common. Its more frequent use in the case studies is directly tied to the frequency of the Between scope, especially when the Between scope has been used to represent single-variable state information. In fact, 45% of the formal property specifications in the Chemotherapy study are Between Existence, and all of those are used to express state information. For example, consider the informal specification of the Chemo.B.3 property:

*Before chemotherapy can be administered to a Patient, that Patient’s signed consent form for chemotherapy must be present in that Patient’s record.*

As is mentioned in Section 5.3.3, since many of these properties are focused on making sure that the state of the world is “safe” before any treatments are performed for a patient, if the world can change from a safe state to an unsafe state, it is necessary to specify two separate formal properties in PROPEL, one to ensure that the state of the world becomes safe at least once before the treatment is performed, and the other to ensure that each time the state of the world becomes unsafe, it has to become safe again before the treatment is performed. It is this latter type of restriction that is specified in the Chemo.B.3b property in Appendix C.2, where a Between scope and an Existence behavior are used together to ensure that a signed consent form is present in a patient’s record before chemotherapy can be administered to that patient.
5.3.4.3 Scope frequencies are related to behavior frequencies

The frequencies of the scopes and the behaviors are related to one another, as is evidenced by the fact that there are certain scope-behavior combinations that occur more often than others in the case studies, for some of the reasons discussed in previous sections. While the combination of a Between scope and an Existence behavior make up 26% of all the formal property specifications in the case studies, it is not the most common scope-behavior combination. The most common combination is Between Precedence, with 28% of the formal property specifications, followed by the Between Existence combination, the Global Precedence combination with 24%, and the Global Response combination with 13%. There are six scope-behavior combinations that each appear in less than 5% of the formal property specifications: Before Response, Before Precedence, Between Absence, and Between Response, After Absence, and After Response. The other six scope-behavior combinations do not occur at all in the current set of formal property specifications for the case studies.

The frequencies of the scope-behavior combinations in the pattern survey differ significantly from those in the case studies. The Global Response scope-behavior combination was the most common in the pattern survey, appearing in 43% of the properties. The next most common combination was Global Universality and it appeared in 20% of the properties. This latter scope-behavior combination illustrates one of the possible reasons why the pattern survey differs from the case studies on this point: the pattern survey included more behaviors than are currently supported by PROPEL, and thus it is likely that more of those properties could be specified with a Global scope, since the behaviors used covered additional types of restrictions. For example, in addition to the Universality behavior, the pattern system included a Chain Precedence behavior and a Chain Response behavior. Each of the ordered AND decompositions in these case studies could probably be expressed with a version of one of those Chain behaviors. Where PROPEL’s limitations forced us to specify the ordering with several properties, it may have been possible to express the same information with only one Chain property using the pattern system. If that is the case, it should be taken into consideration when comparing the frequencies of these scope-behavior combinations in these case studies to those in the pattern survey.

5.3.5 Observations about option-setting frequencies

Figure 5.4\textsuperscript{8} shows the distribution of the option settings for all the scopes and behaviors. For each option, the figure gives a brief textual description, the space of possible settings for that option, the total number of times each setting occurred in the case studies, and the percentage that that total number represents.

\textsuperscript{8}The percentages in Figure 5.4 sometimes do not sum to 100% because of rounding imprecision.
<table>
<thead>
<tr>
<th><strong>Scope Options</strong></th>
<th>Option</th>
<th>first or last start?</th>
<th>restricted interval needs end?</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>first start</td>
<td>76</td>
<td>scope needs end</td>
</tr>
<tr>
<td></td>
<td>last start</td>
<td>71</td>
<td>105</td>
</tr>
<tr>
<td></td>
<td></td>
<td>48%</td>
<td>67%</td>
</tr>
<tr>
<td></td>
<td></td>
<td>52%</td>
<td>scope w/o end</td>
</tr>
<tr>
<td></td>
<td></td>
<td>52</td>
<td>33%</td>
</tr>
<tr>
<td></td>
<td>is the scope repeatable?</td>
<td>multiple intervals</td>
<td>B can precede</td>
</tr>
<tr>
<td></td>
<td></td>
<td>136</td>
<td>32</td>
</tr>
<tr>
<td></td>
<td></td>
<td>94%</td>
<td>64%</td>
</tr>
<tr>
<td></td>
<td></td>
<td>9</td>
<td>18</td>
</tr>
<tr>
<td></td>
<td></td>
<td>6%</td>
<td>36%</td>
</tr>
<tr>
<td><strong>Behavior Options</strong></td>
<td>Option</td>
<td>is A required?</td>
<td>can B precede A?</td>
</tr>
<tr>
<td></td>
<td>A req’d</td>
<td>44</td>
<td>B can precede</td>
</tr>
<tr>
<td></td>
<td>A not req’d</td>
<td>139</td>
<td>32</td>
</tr>
<tr>
<td></td>
<td></td>
<td>24%</td>
<td>64%</td>
</tr>
<tr>
<td></td>
<td></td>
<td>76%</td>
<td>B cannot precede</td>
</tr>
<tr>
<td></td>
<td></td>
<td>76%</td>
<td>36%</td>
</tr>
<tr>
<td></td>
<td>secondary events?</td>
<td>none</td>
<td>prohibited</td>
</tr>
<tr>
<td></td>
<td></td>
<td>59</td>
<td>51</td>
</tr>
<tr>
<td></td>
<td></td>
<td>32%</td>
<td>28%</td>
</tr>
<tr>
<td></td>
<td></td>
<td>allowed</td>
<td>73</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>40%</td>
</tr>
<tr>
<td></td>
<td>how many A’s before B?</td>
<td>at most one</td>
<td>how many B’s after A?</td>
</tr>
<tr>
<td></td>
<td></td>
<td>41</td>
<td>at most one</td>
</tr>
<tr>
<td></td>
<td></td>
<td>22%</td>
<td>75</td>
</tr>
<tr>
<td></td>
<td></td>
<td>78%</td>
<td>41%</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>59%</td>
</tr>
<tr>
<td></td>
<td>can behavior repeat?</td>
<td>can repeat</td>
<td>can A occur after B?</td>
</tr>
<tr>
<td></td>
<td></td>
<td>105</td>
<td>A can occur</td>
</tr>
<tr>
<td></td>
<td></td>
<td>57%</td>
<td>181</td>
</tr>
<tr>
<td></td>
<td></td>
<td>43%</td>
<td>99%</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>1%</td>
</tr>
<tr>
<td></td>
<td>how many A’s can occur?</td>
<td>at most one</td>
<td>A cannot occur</td>
</tr>
<tr>
<td></td>
<td></td>
<td>10</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td></td>
<td>16%</td>
<td>99%</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>1%</td>
</tr>
<tr>
<td></td>
<td></td>
<td>multiple A’s</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>54</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>84%</td>
<td></td>
</tr>
</tbody>
</table>

**Figure 5.4.** Scope and Behavior Option Settings Distribution Summary
There are some interesting observations to be made about the relative frequencies of those scope and behavior option settings. With regards to the scope option settings, we observed that each of the case studies, when looked at individually, tends to be lopsided on the option setting that is used for the “first or last start?” option: within a given case study, most of the formal property specifications would use one option setting almost to the exclusion of the other. The aggregation of all the case studies, however, led to a nearly even split between the two option settings, as is shown in Figure 5.4. We suspect that the individual differences between the case studies are due to differences in the domain boundaries and to whether or not the Between scopes, which account for almost all occurrences of this option, are being used primarily for enforcing repeatability or for expressing state information. In the former case, “first start” is more likely to be used, whereas in the latter case, “last start” is more likely to be used. In the other two scope options, however, there is an obvious difference between the relative frequencies of the option settings. For example, the “is the scope repeatable?” option has an overwhelming majority of the instances of the Between scope set to be repeatable. The only ones that are not repeatable are in the Chemotherapy study, where the domain boundaries are set to be within a single course of chemotherapy, which is defined by the creation of a single treatment plan. Once the treatment plan has been created—and approved, if necessary—those events are not expected to occur again, and thus the scope does not need to be repeatable.

In addition to these observations about the Between scope’s options, we also observed an interesting correlation between the Existence bound and the use of the Between scope. As was mentioned earlier, the second most common behavior in the case studies is Existence, and this behavior has one option, “how many A’s can occur?”, which has two option settings. We found that one of those settings is far more prevalent than the other, and that which setting is used is correlated with the type of Between scope that each Existence behavior is combined with. In most of the formal property specifications that use an Existence behavior, there is no upper bound on the number of times that the primary event can occur. We observed that, with the exception of one property, the properties that have an upper bound on the number times that the primary event can occur are the same properties where the Between scope is not repeatable.

With regards to the options in the properties’ behaviors, there is a general trend towards restricting as little as possible. That is, for each behavior option, the setting where no restriction is made is more common than the setting where a restriction is made. In general, we intentionally limited the number of restrictions enforced by any single formal property specification to just the main point of what that property was intended to restrict. In doing so, we aimed to avoid distractions that can make a property unnecessarily complex and can thus create difficulties for the domain experts. This issue is discussed in more detail in Section 5.4.4. Despite our attentiveness to this issue, we still found that we were restricting the Nullity option too much. After some preliminary analyses of the properties with respect to the associated medical processes, we realized that
many of the exceptional situations described in those processes meant that the Nullity option in the properties should not be set to “A required”, because there are valid ways to step through the processes that do not include an occurrence of A.

5.3.6 Summary of the quantitative results

During the course of these case studies, we were able to use PROPEL to formalize 95% of the informal property specifications that we encountered. This overall coverage seems high, and indeed when we look at the results in more detail, we find a number of limitations on PROPEL’s expressibility that cause us to temper our conclusions. Certain concepts cannot be expressed in PROPEL at this time, such as multivariate state information that defines a scope, N-1 Chain Response behaviors [81], event conjunctions, and event bounds. In addition, 56% of the properties’ formal specifications in PROPEL are specified somewhat awkwardly, since they make reference to event disjunctions, single-variable state information, a variable number of both allowed and prohibited secondary events, or real-time constraints. We are able to specify these types of information in a limited fashion, and it often involves using multiple formal specifications for a single informal specification, to capture all the necessary details. In fact, 51% of the informal specifications that could be expressed in PROPEL map to a conjunction of two or more formal specifications. This 1-to-many mapping plays a part in explaining the differences between the pattern survey [82] results and the case study results, most notably with respect to the sharp differences in the number of Between scopes and the number of Precedence and Existence behaviors. The Between scope and Existence behavior are of particular use in specifying single-variable state information, which always required multiple formal specifications in PROPEL, and the Precedence behavior is used heavily for specifying ordered AND decompositions, in addition to specifying the ubiquitous type of medical constraint, “a safety check must be completed before an intervention can be performed for a patient.” As evidenced by this type of constraint, it is likely that the other major factor to play a part in explaining the differences between the pattern survey results and the case study results is the different domains from which the two sets of properties are drawn.

While there are a number of reasons (see Section 5.5 for a more detailed discussion of them) to be cautious about coming to any strong conclusions about PROPEL based on these case studies, these results do give some indication of the value of our approach to property specification. It should be noted that the purpose of the case studies is not so much to demonstrate the breadth of PROPEL’s expressibility, but rather to begin to address the issue of whether this approach can be used to specify enough of the properties encountered in some example real-world situations to be worth considering in more depth (i.e., Proposition 1 in Section 5.1). Given these promising results, we do think further exploration of this approach will yield valuable results.
5.4 Qualitative Observations

During the course of conducting the interviews with the domain experts and collecting quantitative information, the computer scientists recorded qualitative observations about the entire team’s experiences elucidating the properties. There was an initial learning stage for all the team members, both the computer scientists and the domain experts, before all of us were able to make forward progress on the property specifications. Once we started to make progress, we gained insights about the four major types of property specification issues that the case study methodology described in Section 5.2 aims to address:

- identifying an abstract (i.e., high-level), informal specification of a property;
- stating an informal property specification clearly;
- formalizing a property specification; and
- organizing a set of property specifications.

The development of the property specifications went through various stages in which one or more of these types of issues was at the forefront. As was mentioned in Section 2.3.3, the progression through these issues was not necessarily linear because they are interrelated, and often changes made to the properties prompted the team members to revisit these issues multiple times. Each of these issues is discussed in more detail in the following sections.

5.4.1 Initial learning stage for both computer scientists and domain experts

One of the first observations that was made about these case studies was that the computer scientists and the domain experts needed to learn enough about each other’s areas of expertise to communicate effectively. The domain experts did not initially understand what the computer scientists meant by the concept of a property, and the computer scientists’ limited understanding of the domains made it difficult at first to offer specific guidance on what the properties in the domains might be. As was mentioned in Section 5.2, these case studies exist in the context of a larger medical safety project. Because of this larger context, the property elucidation activities in three of the case studies (BT, Chemo, and VPID) were sometimes proceeding in parallel with the elucidation of other types of information, such as the control and data flow in various processes and the resources needed to perform the tasks in those processes. A decision was made early in the development of the larger project to elicit at least part of the processes and resources first, before elucidating the properties. The process elicitation was thus part of the means that the team members used to build a common understanding of the domain. Although the work reported here is based on eliciting process
information before property information, alternatives to this order are possible and are discussed in more detail in Chapter 7.

Given the constraints of this initial learning stage, the elicitation activities ended up occurring in roughly the following order:

- The computer scientists started by asking the domain experts to describe the processes they used.
- The domain experts started by telling detailed stories about their daily experiences.
- The computer scientists encouraged the domain experts towards generalizing the detailed stories into process descriptions.
- Throughout the previous steps, the computer scientists listened for any possible statements the domain experts might unknowingly make concerning abstract goals.
- The computer scientists encouraged the domain experts towards consciously identifying abstract goals themselves.

The computer scientists observed early on that the domain experts would often state properties as they described why they performed various tasks in their processes. This also happened in casual conversations outside of explicit process-elucidation activities, when the domain experts were discussing recent “war stories” in their experience. In these stories, there was usually some undesirable outcome that violated an implicit abstract goal, and this led the computer scientists to ask what that goal was, to understand why the outcome was undesirable. This allowed the computer scientists to indirectly elicit some initial properties and to use them as examples to aid the domain experts in understanding the concept of a property, so that the domain experts could identify properties themselves. Similar findings about this type of initial learning stage are reported in [16, 128, 254].

5.4.2 Issues involved in identifying an abstract informal specification of a property

There were two main challenges to arriving at an abstract informal property specification: identifying the abstract goal that a property should specify and finding a suitable level of abstraction for the property specification. One of the main reasons that these challenges existed was that the team members were coming from three different perspectives: scenario-based, process-based, and goal-based. All three perspectives can be brought to bear when identifying an abstract goal, but the differences among the perspectives can also lead to confusion about the appropriate level of abstraction, since the different levels of abstraction that the three perspectives provide have different strengths and weaknesses. With experience, it became clear that the type of property specifications that the team was aiming for can be more easily created from a process-based
or a goal-based perspective than from a scenario-based perspective. In addition to understanding these three perspectives, finding a suitable level of abstraction for a property specification required mastering two skills: understanding the domain experts’ goals for a case study and managing the domain complexity. All of these issues are discussed in detail in the remainder of this section.

5.4.2.1 Three different perspectives influenced abstract goals and levels of abstraction

Since the team members saw the case studies from three different perspectives, much of the challenge was in learning how to leverage all three perspectives effectively. The scenario-based perspective focuses on the specifics of one example situation in which the domain experts do a task. In the process-based perspective, however, the focus is instead on generalizing from the specifics of that one example and perhaps several other examples, to derive a description of how that task should be done in most or all situations. In contrast to these previous two perspectives, the goal-based perspective focuses on the underlying reasons (i.e., the abstract goals), behind why that task is being done in the first place. We found that all three perspectives contribute towards identifying an abstract goal and identifying the appropriate level of abstraction for its specification.

Since the team members’ aims in the larger medical safety project were slightly different for each case study, the resulting property specifications were at different levels of abstraction. These perspectives and their influence is discussed in detail in the following sections.

5.4.2.1.1 Scenario-based perspective: “What is an example situation in which we do this task?”

The scenario-based perspective is the most concrete, in terms of the daily experiences of the domain experts, and thus it was a perspective primarily held by the domain experts, especially at first. The main strength of this scenario-based perspective is an awareness of the exact details of a wide breadth and depth of possible scenarios in the domain. In the first few interviews, the domain experts mainly told stories about individual scenarios, and these stories were based on their actual experiences. Although the domain experts’ focus eventually shifted to include the other two perspectives to some degree, the domain experts’ main strength remained their ability to vet the process and property specifications developed later against this detailed knowledge of the domain.

To illustrate the scenario-based perspective, let’s look at an example scenario from the Chemo study. This scenario is described in a narrative form, as a domain expert might have described it.9

*So just last week I had a patient—let’s call her Mrs. Schlobotnik—who came in to get her chemo, but her chart wasn’t in the pile that morning, and there she was on the schedule. I tracked down*

---

9This is a reconstruction of a scenario narrative, not a verbatim quote from a domain expert. All names are fictitious.
her chart—Alice had it—and then I got her all ready, had her sit in the chair, made sure she had an ID band, and then I did a review of systems. When I flipped through the chart, I realized that she hadn’t signed a consent form. I asked her about it, and she said that she’d come in for a chemotherapy teaching the week before but that no one had asked her to sign anything. This made me upset, because here Mrs. Schlobotnik was, sitting right in the clinic, and now I was going to have to make her wait even longer to get her chemo, because I was going to have to do her chemotherapy teaching right there on the spot. I called Alice to find out what was going on, since she was the nurse practitioner listed on the schedule for Mrs. Schlobotnik’s teaching, and it turned out that Mrs. Schlobotnik had signed the form, but it must have fallen out of the chart, because it was still there on Alice’s desk. Mrs. Schlobotnik must have been confused. I don’t think English was her first language. Anyway, after Alice brought the consent form down to me, I drew blood for the lab tests, to make sure Mrs. Schlobotnik was well enough to get chemo...

As can be seen from this example, there is a great deal of detail provided about the specific experiences of this person while they tried to perform this task, including frustrations and references to particular people involved. While this level of detail is valuable, it also presents some difficulties. Experiences differed among the various domain experts who were interviewed, and a single person’s account could also change over time. These differences often occurred because the domain experts would use what they learned during the interviews to make changes to their daily practice in between interview sessions. Such changes were made to address one defect, but they often introduced other defects elsewhere, which prompted further changes to their processes, and so on. Given the enormous amount of information that was available at this level of detail and the degree to which that information changed as often as it did, the team encountered conflicts between the scenarios that were provided and thus found it difficult to describe the processes and identify abstract goals using only this scenario-based perspective. Upon realizing that these types of changes were counter-productive for the larger medical safety project, the team’s focus shifted more towards the other two perspectives.

5.4.2.1.2 The process-based perspective: “How do we do this task?” The process-based perspective is more abstract than scenarios, in that its focus is on aggregating the scenarios that the domain experts describe and finding generalizations about what steps the domain experts are performing to accomplish their tasks. The main strength of this perspective is an awareness of domain processes as instances of abstract control and data flows. This is an area in which the computer scientists have expertise, so they tended to start from this perspective. In addition, the domain experts started with some intuitive understanding of the basic concept of a process and were able to help the computer scientists generalize from example scenarios.
To illustrate the process-based perspective, let’s return to the example given in the previous section, but this time frame it in terms of a process description:

*The clinic nurse checks whether or not a signed consent form is in the patient’s chart.*

1. *If there is no signed consent form, the clinic nurse checks when the teaching was scheduled by looking at the computer scheduling system records to find out which nurse practitioner was assigned to do the chemotherapy teaching for the patient.*
   
   (a) *If the nurse practitioner who was assigned to do the chemotherapy teaching is available and has access to the signed consent form, the nurse practitioner either brings the signed consent form to the clinic nurse or has a clinic medical assistant do it.*
   
   (b) *If the nurse practitioner is available but does not have access to a signed informed consent form, then either the clinic nurse or the nurse practitioner does the chemotherapy teaching for the patient.*
      
      i. *After the chemotherapy teaching, the clinic nurse or the nurse practitioner has the patient sign a consent form.*
   
   (c) *If the nurse practitioner is not available, the clinic nurse calls the doctor for further instructions.*

2. *If there is a signed consent form, the clinic nurse goes to the next step in the pre-chemotherapy checklist.*

This process specification differs from the earlier scenario narrative in several important ways. It was created from multiple scenarios and it focuses on defining a series of steps in a precise order, such that these steps can be followed to complete the chosen task. All references to particular people have been removed and replaced by abstract descriptions of the roles that those people played in the narrative, to indicate that anyone fulfilling that role is a resource that can contribute to the completion of the task. The process description also includes directions about how to recover if exceptions, such as a missing resource, occur.

As was mentioned in Section 5.4.1, this process elucidation occurred somewhat in parallel with the property elucidation, and this both supported and challenged effective property elucidation. Section 5.4.1 describes how the process elucidation enabled the team members to begin to communicate effectively. In addition to that benefit, the property specifications that were identified through the process-based perspective were easily operationalizable, since they were clearly related to activities in the process. This latter benefit was of particular interest to the computer scientists, since it addressed some of their goals for the larger medical safety project. There were also some problems that this emphasis on the process-based perspective
introduced, however. The early property specifications that were identified during the process-elucidation activities tended to focus too much on process implementation details and could be viewed as only slightly generalized restatements of the process. As an example of this problem, here is an early property specification for the example task:

*Before the clinic nurse can administer chemotherapy to a patient, either the clinic nurse, the nurse practitioner, or a clinic medical assistant must put the patient’s signed consent form in the patient’s chart.*

This property specification shows a great deal of attention paid to the details of who is putting the consent form in the chart, but at this level of detail, there are three problems. One problem is that such property specifications changed too often to be used as generalizations of the scenarios. In addition to the changes being made to the processes in between interview sessions, these property specifications also changed because the domain experts’ understanding of their processes grew deeper over time. The more the domain experts thought about the issues involved, the more scenarios, exceptional situations, and alternative ways of completing the task were introduced, and the property specifications needed to be modified to capture the new information. For example, that property specification initially only referred to a nurse practitioner, but the domain experts realized that there were scenarios where other people might be putting the consent form in the chart. Another problem with specifying properties at this level of detail was that there were so many that doing so was prohibitive. It became apparent that there would need to be a large number of properties to cover all the scenarios that the domain experts could imagine, even when they were limited by the chosen domain boundaries (see Section 5.4.2.2). At this level of detail, there would be so many properties that it would not be possible to elucidate them all effectively within the practical limitations of our case studies. A more essential problem with specifying properties at this level of detail, however, is the lack of abstraction in the resulting property specifications. The example property specification given above focuses on specific actions taken by specific people, but not on why those actions are necessary in the first place. In other words, this type of property specification does not address the real patient-safety issue that the domain experts wanted to focus on.

**5.4.2.1.3 The goal-based perspective: “Why do we do this task?”** The goal-based perspective is the most abstract of the three, in that its focus is on eliciting and organizing the abstract goals that motivate the scenario-based and process-based perspectives: these abstract goals apply to all relevant scenarios and versions of the processes. This goal-based perspective is central to the development of the property specifications, since its focus is on the essence of what a property is (see Chapter 1). The main strength of this perspective is an awareness of abstract event-sequence patterns that are independent of any particular
domain. This is an area in which the computer scientists have expertise, but they did not begin to focus on this perspective until they had spent some time with the domain experts and had begun to develop an understanding of what some of the abstract goals in the domain might be. One issue that arose when the team was starting to identify abstract goals was whether it would be better to capture abstract goals for an ideal process or for the existing process. It was sometimes helpful to identify abstract goals for existing processes, but the team members had to be careful to avoid implementation bias: the team did not want the shortcomings of the existing process to hobble the development of a clear specification of the abstract goals. To avoid this problem, the team decided to work based on the assumption of an ideal world, because this assumption made it easier to focus on the abstract goals.

Given this assumption of an ideal world, it was sometimes easy to become too abstract when creating property specifications. Specifications that are too abstract are of limited value, because they need enough detail to be operationalizable. This was important for the purposes of the larger medical safety case study and as a sanity check when the domain experts vetted the property specifications. To illustrate this balance, consider the following example property specifications, Chemo.B.2 and Chemo.B.7:

- **Before chemotherapy can be administered to a patient, that patient must sign a consent form for chemotherapy.**
- **Before a patient can sign a consent form for chemotherapy, that patient must have chemotherapy teaching.**

These two property specifications provide an example of three benefits of identifying properties from this goal-based perspective. At this level of detail, the property specifications tended to be more stable over time, since they were concise generalizations of the scenarios and they were not tied to the implementation details of particular versions of the processes. In addition, fewer properties are needed to cover all the scenarios that the domain experts could imagine within the chosen domain boundaries. This made the resulting set of properties easier to specify and to organize clearly, as is discussed in Sections 5.4.3 and 5.4.5, respectively. Perhaps the most significant benefit of specifying properties from a goal-based perspective, however, is that the resulting property specifications expose the underlying goals that the domain experts have in their domain, and it becomes easier for the domain experts to evaluate their processes and daily experiences in light of these explicitly-stated goals.

5.4.2.1.4 A summary of the three perspectives. As the running example in these sections shows, the three perspectives are complementary and they all contribute towards a deeper understanding of the abstract goals in the domain. The scenario-based perspective provides a narrative that is a rich, real-world source of
information about a task. The process-based perspective provides a detailed set of ordered steps to complete that task and enumerates the possible cases and which steps should be performed in each case. These are invaluable types of information for daily medical practice, since the focus of these two perspectives is on what the task is and how it is to be done, respectively. By contrast, the focus of the goal-based perspective is on the underlying patient safety issues that motivate why the steps in the task are being done, and perhaps also why they are being done in that particular order. The value of separating out the property specifications from the process descriptions is that doing so draws attention to the abstract patient safety goals in the domain, and an awareness of these goals is a necessary step in evaluating and improving the effectiveness of any medical process that aims to adhere to them. The property specifications must be expressed at the appropriate level of abstraction, however, or they can end up being of only limited use. It is important to make sure that the properties neither constrain the implementation of the processes nor abstract so much information away that they are too vague to be of practical use. This issue is discussed in more detail with respect to property decompositions, in Section 5.4.5.1.1. These three perspectives matured over the course of this case study, in that the team members came to a shared understanding of the perspectives’ strengths and weaknesses. This shared understanding enabled the team to specify properties at a level of abstraction that the domain experts could use to describe their domains in a succinct fashion and that the computer scientists could use to apply some additional Software Engineering techniques, with the overall aim of improving patient safety in these domains.

5.4.2.2 Domain boundaries influenced a property specification’s level of abstraction

Many of the considerations concerning a suitable level of abstraction for property specifications are discussed in the previous sections, but there remains another major consideration: the problem of how to manage the sheer size and complexity of the domains. As the domain experts thought critically about their processes and properties, they often realized that activities that they used to think of as simple tasks were actually very complex processes in their own right. It quickly became clear that the domains in these case studies were so large that it seemed possible to find exceptions to almost any property specification except for those that were specified with so much abstraction that they were effectively useless because they were too vague. This awareness prompted the team members to carefully consider their expectations for the case studies, and the decision was made to explicitly assume certain limitations on the domains’ boundaries. There was clearly value in trying to describe the domains within these boundaries, because these circumscribed domains still provided more than enough complexity to be a challenge. The domain boundaries were also meant to codify the expectations of both the computer scientists and the domain experts, and it was thus necessary to agree on the domain boundaries before a level of abstraction could be found that satisfied the expectations of everyone.
involved. These domain boundaries are stated explicitly for each of the case studies, as can be seen by the following example from the BT study:

*All the properties in the Blood Transfusion study are assumed to be in the context of:*

- one patient and one medical professional (except where explicitly noted that two medical professionals must be present) in an in-patient situation; and
- concerned with carrying out only one physician order for blood transfusion, where it is possible that a single physician order may specify that multiple units of blood product are to be transfused.

Limiting the domain boundaries in this fashion helped to narrow the set of abstract goals that needed to be identified and provided a general framework for defining the relative levels of abstraction internal to each case study. The different purposes that each case study has in the larger medical safety project means that each case study has different internal notions of what the most- and least-abstract types of property specifications are, depending on what is valuable from the perspectives of the particular domain experts involved. Without such explicitly-defined domain boundaries, it is difficult to identify the most- and least-abstract types of property specifications, but with these domain boundaries, it becomes easier to shift from the more abstract to the more detailed, and vice versa, because the domain boundaries provide a shared understanding of where the focus is in each case study.

The domain boundaries were not necessarily static during the course of conducting these case studies, however. The domain boundaries sometimes expanded or contracted as details in the domain became clearer to the team. For example, consider the Chemo study, where the domain experts initially limited the domain boundaries to just the activities leading up to the first administration of chemotherapy. This simplification was necessary when the team members were in the initial learning stage, but it obscured a key cardinality relationship between the chemotherapy orders (many) and the treatment plan (one) until late in the study. This led the computer scientists to an incomplete understanding of the relationship between the approval of the treatment plan and the verification of the chemotherapy orders, since the computer scientists were working under the misconception that a single event occurred in which both documents were approved for an entire course of chemotherapy. Because of this misunderstanding, it was not clear to the computer scientists why the domain experts kept stating that the approval of the treatment plan was a very different activity than the verification of the chemotherapy orders. The confusion disappeared when the domain experts stated that the domain boundaries had expanded as the project matured, and that they now included all administrations in a course of chemotherapy, not just the first administration. The domain boundaries of Chemo and all the other case studies are given in Appendix C.
5.4.3 Issues involved in stating an informal property specification clearly

Once an abstract informal property specification was initially identified, another challenge was stating it clearly. There were a number of terminology problems at first, which led to confusion. It thus became essential to develop a glossary of terms in each domain as quickly as possible, and to use those terms consistently. The domain experts responded very positively to these glossaries and they quickly adopted the use of agreed-upon terminology. It was clear to all the team members that carefully distinguishing terms led to a better understanding of the processes and properties in each of the domains. In addition to developing glossaries, the team also used several simple techniques to improve the clarity of the informal property specifications, such as assigning unique identifiers to each one, annotating them with clarifying information, and minimizing word repetition. In this section, we discuss each of these issues in more detail.

5.4.3.1 Addressing terminology problems

5.4.3.1.1 Terminology problems. One issue that arose early in the property and process elucidation activities was the problem of incomplete terminology. The domain terminology was considered incomplete when there was no abstract term to describe a concept, only an instance-based term. For example, in one study there was a member of the administrative staff who coordinated an important flow of information, and the domain experts knew this person was an essential part of enforcing patient safety concerns. This person’s role had grown to its present responsibilities and capabilities largely because of the competence of the individual filling that role, and the only term that the domain experts used to describe that role was the person’s actual name. Since there was no abstract term that was chosen to refer to that person’s role, when the person left, the term to describe the role then changed to be the name of the next person to fill that role. After discussing the confusion that these types of changes in terminology can introduce, the domain experts adopted abstract terms to describe such instance-based concepts.

We also encountered non-minimal and inconsistent terminology while elucidating the properties and processes. Non-minimal terminology was used when the domain experts knew that multiple terms referred to the same concept and thus used those terms somewhat interchangeably. For example, in the Chemo study, “infusion-suite nurse,” “chemotherapy nurse,” “infusion nurse,” and “clinic nurse” all referred to the same role. In the BT study, “unit,” “blood bag,” and “unit of blood product” all referred to the same artifact. Inconsistent terminology was used when the domain experts had a single term that meant multiple distinct things. The domain experts knew that the meaning of the term changed based on the context of its use, but such changes led to confusion, especially when trying to develop properties that needed to distinguish among multiple senses of a term in a single specification. An example of inconsistent terminology in the BT study is the use of the verb “transfuse.” In the early property and process specifications in this study, “transfuse blood”
was used both to refer to the abstract activity of carrying out a physician’s order for a blood transfusion and to refer to just the specific task of physically infusing a unit of blood product into a patient’s body. To reduce the incidence of non-minimal and inconsistent terminology, the computer scientists encouraged the domain experts to choose a single concept to associate with each term and vice versa.

There was another subtle terminology issue that arose, when all the domain experts implicitly assumed that a very commonly-used term was associated with only one concept and vice versa, but upon closer inspection, this assumption was found to be false. This was the problem of undefined terminology. Often, the domain experts were not initially aware of the ambiguity in their commonly-used terms. A prominent example was the ubiquitous use of “verify patient identity.” This term is used in training manuals, textbooks, institutional policies, and statements of best practices, but it is vague and masks hidden complexity. During the course of these case studies, the domain experts observed that in practice there is a wide variety of interpretations of this term, many of which do not satisfy patient safety concerns.

Related to this issue of implicit assumptions about commonly-used terms is the occurrence of common terms that are present in multiple medical domains, or are used in both computer science and medicine, with very different meanings. It is important to be as consistent with terminology across the different medical domains as possible, especially when there are shared concepts, such as the verification of some artifact. More than one domain uses “verification” as an abstract term that refers to a set of more detailed checks, where each check is either (a) a confirmation of the existence of some datum, (b) a test for an exact string or numerical match between two pieces of data, or (c) a test for consistency between two pieces of data, where this consistency requires professional medical judgment to determine. Although this particular term was relatively simple to unify across the domains in the case studies, there were some cases where it was not possible to unify the terminology. For example, the meaning of the word “algorithm” differs significantly between computer science and medicine, and the use of the term is central and widespread in both areas. In such situations, it was important that all of the team members were aware that the term was confusing and that they should endeavor not to use it if alternatives were available.

5.4.3.1.2 Glossary development. It should be noted that the use of the word “glossary” here does not refer so much to words associated with definitions—although that is a component of our glossaries—as it does to a simple ontology of concepts in each of the domains. In addition to the terms and the associated definitions, the glossaries also contained a record of known synonyms and other notes intended to clarify such possible sources of confusion. We actually created two different types of glossaries: one type was oriented around the nouns or noun phrases (i.e., terms) that were deemed important enough concepts to be included in each glossary, and the other type was oriented around the roles of the different medical professionals involved.
In a given domain, the terms were associated with all the verbs or verb phrases that they occurred with in the property specifications, and this arrangement was valuable in summarizing what artifacts, both physical and virtual, were essential components of the domains and how they could be manipulated. In the second type of glossary, the various medical professionals’ roles were associated with the same verbs or verb phrases, and this arrangement provided a clear summary of the capabilities and responsibilities of each role. Since the Chemo study was the only one where multiple different roles were important to the specification of the properties, it was the only study that included this second type of glossary.

The domain experts responded positively to the development and use of these glossaries. They quickly understood that confusion was introduced by the terminology problems discussed in previous sections and they worked carefully not only to arrive at terminology that was clear and unambiguous, but also to change their own use of terms during subsequent interviews. They responded well to having specific questions asked about how to clarify or define terms, especially when the questions highlighted how a term could be misinterpreted. The domain experts often went beyond the computer scientists’ questions and thought of other things that needed to be considered about the terms. One additional consideration that they pointed out is the need to ensure that the property specifications are based on the word-order conventions that the domain experts are most comfortable with. For example, the domain experts preferred a particular order when describing checks where two pieces of data were being compared. In the phrase “A is compared to B,” the domain experts assumed that more weight was given to B in the event that the two items did not match (or were not consistent with each other), and A would thus be more likely to be questioned than B. A concrete example of this can be found in the Chemo study, where the chemotherapy orders must be consistent with the treatment plan, not vice versa.

Another consideration that the domain experts pointed out is the importance of paying very careful attention to the terminology used in the event names that are in the formal property specifications, since even a slight misinterpretation of the event names can lead the domain experts to think that an incorrect property specification is correct. For example, in an early version of the Chemo property specifications, there was one specification that stated,

An Attending MD must not enter a patient’s chemotherapy orders into the computer system until after that Attending MD confirms a cancer diagnosis by reviewing that patient’s pathology report.

The meaning of the phrase “confirms a cancer diagnosis” is that the Attending MD decides that there is a cancer diagnosis for that patient, not just that the Attending MD checks whether or not there is a cancer diagnosis for that patient. At first, the computer scientists incorrectly implied that the latter interpretation
was the meaning and the domain expert started to answer the questions from that perspective. After trying to resolve some of the subtle details, however, the domain expert realized that the property specification did not make sense given that incorrect interpretation. The domain expert quickly caught the fine difference and this changed how this person resolved the subtle details in the property specification. This experience taught the computer scientists to be more careful about clarifying the event names used in the property specifications whenever there is potential for confusion, and to make sure to note precisely which possible interpretation is the correct one when the terms used in the event names are stored in the glossary.

In addition to changing their use of terms during interviews, the domain experts also reported that focusing on terminology issues affected their daily practice outside of the project meetings. They reported that it heightened their awareness of ambiguities and common pitfalls. For example, the Nursing professor reported that it prompted her to make changes in the terminology that she used when she taught her students, such as separating the use of “transfuse blood,” which she used as an abstract term for the entire process of carrying out a physician order for blood transfusion, from “infuse blood,” which she used to refer to just the physical transmission of a unit of blood product into a patient’s body. Through the focus on the glossaries, she also worked to be more consistent in her use of terms while teaching. In another case, a pharmacist reported that he had begun to notice that some of the posted instructions in the pharmacy were ambiguous. When the domain experts noticed these common pitfalls, it often led to a heightened awareness of potential safety hazards in common practice. For example, after carefully examining the term “suspect a transfusion reaction,” one domain expert said that the volume-based suspected transfusion reactions are probably commonly under-reported because often a nurse chooses to just pause the blood infusion and monitor the patient (and then resume the infusion later if there are no further problems) instead of completely stopping the blood infusion and reporting the suspicion. The domain expert did not know if there is anything noted in the official definitions of “suspect a transfusion reaction” that eliminates this ambiguity by, for example, explicitly including all clinical parameters of a possible transfusion reaction. This example highlights how the heightened awareness of ambiguities in the medical domain prompted a deeper understanding of possible reasons why certain medical errors occur, and thus part of why the domain experts so quickly embraced this close examination of their domain terminology.

Besides the positive responses of the domain experts to clarifying their terminology, there was another benefit that resulted from carefully developing the glossaries in these case studies. Often an examination of the terms led to finding new major areas of process and property complexity that needed to be explored. An example that has been used several times in this discussion is the task of verifying a patient’s identity. The attempt to define this term eventually led to a much richer appreciation of the complexity and importance of understanding exactly what this task entails. This is a foundational, common task performed in medical
Table 5.5. Formalization Status Marking Convention

<table>
<thead>
<tr>
<th>Status</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>No</td>
<td>No formal specification of this property has been created yet.</td>
</tr>
<tr>
<td>?</td>
<td>A formal specification of this property has been created, but it is not yet vetted by domain expert(s).</td>
</tr>
<tr>
<td>MP</td>
<td>The formal specification of this property has been vetted by domain expert(s).</td>
</tr>
<tr>
<td>X</td>
<td>A formal specification of this property is not possible in PROPEL.</td>
</tr>
<tr>
<td>- -</td>
<td>Formalization does not apply to this chunk of text.</td>
</tr>
</tbody>
</table>

practice, but until this point it has not been carefully examined. The computer scientists’ focus on this and other terms in the case studies’ domains have prompted the domain experts to begin to look at previously-unexplored threats to patient safety that are rooted in problematic terminology.

5.4.3.2 Techniques for improving the clarity of informal property specifications

5.4.3.2.1 Make the document easily navigable. In addition to clarifying the terminology used in these case studies and developing the glossaries, the team also employed a number of other simple techniques to improve the clarity and readability of the informal property specification documents. One important aspect of making these documents clearer is supporting the ability to easily find and refer to the information recorded in them. There are three main techniques that were used to achieve this. One technique is to give each property specification a unique numerical identifier and to use a numbering style that supports property decomposition and grouping (see Section 5.4.5 for more details on those concepts). Another simple technique is to use font, vertical spacing, indentation, and capitalization consistently, not only within a single case study’s document, but also across all the documents. For example, it is helpful to put the focus on important noun and verb phrases by using boldface font for them. This formatting drew attention to important differences in similar property specifications, and helped to ensure that all important terms were included in the glossaries. The glossary format differs from that used in the informal property specification documents, since it is a table rather than a series of numbered sentences, but wherever the formatting could be kept similar, it is. This consistent formatting makes it easier to quickly identify similar information that is present in multiple case studies and to standardize terminology where appropriate.

A simple marking technique helped the team to keep track of the formalization status of each informal property specification, throughout its lifecycle in a case study’s development. The convention that was used is shown in Table 5.5. In this convention, the “MP” status mark represents the domain experts because, in our case studies, all the domain experts who vetted the formal property specifications were medical professionals (MPs).10 In addition, the “- -” mark is used because not all numbered text in the informal specification

---

10While the software development manager vetted early drafts of 3 of the DBSS property specifications, this person did not yet the versions that are included in Appendix C.5.
documents is a property specification, and it is necessary to distinguish this status from the one where no formal specification of the property has been created yet. Section 5.2 has a discussion of how an informal property specification is defined, and thus which types of text would use this status mark. Examples of the use of this convention can be seen in Appendix C, in the sections that contain the informal property specification documents. As can be seen in those examples, the status mark for a given chunk of text is shown to the left of the text’s unique identifier. The use and placement of these status marks made it easy to visually scan the documents and quickly determine which property specifications to focus on at any given point in the case studies’ development.

5.4.3.2.2 Focus on making the text concise and consistent. Another important aspect of making the informal property specification documents clearer is to focus on making the text concise and internally consistent. A key part of making the text concise is minimizing word repetitions. This was done in a couple of ways. One common source of repetition is when there are multiple checks that must precede (i.e., guard) a single important event. To minimize this text, the guarded event is stated once, and all property specifications that provide guards are listed under the event. In addition to minimizing word repetitions, this type of restructuring also helps to group related properties together. More on grouping is discussed in Section 5.4.5. Another way to reduce word repetition is to state common sets of resources in just one place. For example, if several informal property specifications only differ by agent roles, the informal property specification is stated once and the set of agent roles is listed under it, along with a note intended to clarify how those agent roles fit into the specification. Examples of both of these ways to make the text more concise can be seen in the following examples, Chemo.C.6 and Chemo.C.9:

Before a patient’s treatment plan can be approved:

- that patient’s chemotherapy orders must be consistent with that patient’s treatment plan.
- all the patient data (i.e., height, weight, laboratory results, consult note) in that patient’s treatment plan must not be stale or disparate.

Approvals required: Attending MD (if Fellow MD created the new treatment plan), Practice RN or second Clinic RN (Practice RN preferred), Pharmacy, and Clinic RN.

In the interests of making the text internally consistent, the team also adopted the convention of referring to “a patient” first and to “that patient” for all subsequent occurrences. An example of this is shown above, but this convention is used in multiple structures in the informal property specification documents. It is used in standalone property specifications that refer to the patient multiple times, in guarded-event property
groupings like the one shown above, and in property-decomposition groupings (see Section 5.4.5 for more details).

5.4.3.2.3 Include clarifying notes in the document. Another important aspect of making the informal property specification documents clearer is to include clarifying notes where necessary. There are several types of these annotations on the informal property specifications. One type of clarifying note explicitly states the decomposition relationship when properties are decomposed into sub-properties. Even if the relationship can be understood from carefully inspecting the set of sub-properties given, stating the relationship explicitly makes it easier to grasp. An example of this is given in Section 5.4.5. Another type of clarifying note is a summarization of a group of properties. This type of note can be found under several of the major sections in the Chemo and BT informal property specification documents, but it is not limited to those places. For example, the note “To summarize: that patient’s ID band must be verified by performing the following activities” summarizes properties VPID.A.1 through VPID.A.k+1. This type of note can be used to summarize any group of properties, and it is used whenever a domain expert indicated confusion about a particular grouping. Another type of clarifying note is intended to include relevant glossary content in the informal specification document, when there is value in using an abstract term, but the domain experts wanted to qualify it. There are several examples of this in the case studies, and one of them, “patient data,” can be seen in the example in the previous section, 5.4.3.2.2, with its clarifying parenthetical. Another type of clarifying note, which is also shown in that example, is one that specifies restrictions on resources. In that particular example, there are restrictions listed on when particular agent roles are needed, such as the information that the Attending MD only needs to provide approval if a Fellow MD (i.e., a trainee) was the one who created the chemotherapy treatment plan.

There were two other types of notes, which were not so much intended to clarify the informal specifications as to record important references. For example, in the DBSS study the computer scientists drew heavily on source documents to obtain informal property specifications, and references to precise locations in the source documents are thus included in the informal property specification documents. This can be seen in Appendix C.5. This type of reference was valuable when the computer scientists interviewed the domain experts, because the computer scientists reworded all of the informal property specifications to clarify the text and when the domain experts vetted the specifications, they wanted to see the original text in the source documents, to make sure that the rewording was appropriate. The other type of reference note was used in earlier versions of the case study documents, while they were under development. This type of reference was to similar property specifications in other documents or in other sections of the same document, and was used when the focus was on standardizing terminology or on making the documents more concise and consistent.
Such references helped to guide the discussions so that the team would not leave relevant properties out of consideration. Although these references are not present in the versions of the informal property specification documents that are included in the Appendices, all of the other types of clarifying notes can be found there.

5.4.3.3 Clearly stating a set of properties requires still more work

This section focused on terminology and on a few simple techniques to further clarify informal property specifications, both at the level of an individual property and at the level of an entire document. As can be seen by all the references to Section 5.4.5 that were made in this discussion, the organization of the properties into groups also aids a great deal in improving the clarity of the informal property specifications and their context. Although those organizational issues are discussed in that section, organization alone is not sufficient to unambiguously specify the properties, because these informal property specifications are expressed in natural language. Carefully stating these properties in natural language helped the team to clarify a large number of issues surrounding the properties, and thus enabled both the team and the properties to be ready for more precise specification using PROPEL.

5.4.4 Issues involved in formalizing a property

One of the goals of the property elucidation activities in these case studies was to arrive at a formal specification of the properties in each domain, and the act of transitioning from an informal specification to a formal one introduced a number of challenges. The computer scientists used PROPEL as the tool support for these formal specifications, and these experiences using it enabled the computer scientists to begin to understand both its strengths and its weaknesses when doing property elucidation and specification. PROPEL’s exposure of each the of subtle details required some work on the part of the computer scientists for those details to be clearly understood by the domain experts. Just as the domain experts initially struggled with the concept of a property, to all but one of the domain experts, the concept of specifying a property with a mathematical degree of precision was also outside their area of expertise, and they responded in a variety of ways to the formalization activities. Different domain experts were comfortable using different property views in PROPEL, and they sometimes interpreted those views in different ways. The computer scientists gained new insights into how clear or ambiguous the wording in the QT and DNL template property views was. In addition, the computer scientists learned that it was often difficult for the domain experts to use PROPEL directly, because its design sometimes requires non-intuitive mappings from informal specifications to formal specifications. In addition to learning lessons about the use of PROPEL, the computer scientists also gained a greater appreciation for choosing the appropriate amount of detail in the formal property specifications, since what is
necessary can depend on the property’s intended use. Each of these issues is discussed in more detail in the remainder of this section.

5.4.4.1 Challenges and benefits in formalization

Experiences in formalizing property specifications with the various domain experts gave the computer scientists some insight about where this approach fits into the larger field of RE. It takes a lot of information and iteration to get to a place where a property specification is refined enough that it is ready to be formalized, and even then, sometimes examination of other property specifications or issues in the domain requires that formalization be postponed until the set of properties has re-stabilized. All of the issues about how to identify a property, how to state it clearly, and how to organize the properties’ relationships to one another have to be considered before formalization can be completed successfully. When formalization began too early in the case studies’ development, it tended to focus the domain experts on minute details before the bigger picture had been clearly defined, and thus it became easy for both the computer scientists and the domain experts to “lose the forest for the trees.” Another sign that formalization had begun too soon was when, as is discussed in Section 5.4.2.1.1, the processes and properties changed rapidly early in the project, and the domain experts were sometimes uncertain during that time about how to resolve all the detailed options and became frustrated. It is thus important to consider delaying formalization until the activities involved in identifying, stating, and organizing the properties has begun to stabilize.

The concept of specifying a property at the level of detail required for the specification to be mathematically precise introduced two points of confusion for the domain experts. For them, the activities involved in identifying, stating, and organizing the properties carefully introduced more rigor than they were accustomed to when reasoning about the abstract goals in their domains, and when the computer scientists described going even further, the domain experts were initially taken aback. In some cases, they actually resisted being more precise, because of considerations in the domains’ social and professional cultures. For example, there are several properties that involve the medical professionals who can act in the role of Attending MD, and those property specifications express restrictions on the order in which an Attending MD can perform certain tasks. The domain experts said that they were uncomfortable mandating such restrictions on the order, even if that order could arguably ensure a higher degree of patient safety. The main objection was a cultural one: the Attending MDs expected a certain level of autonomy and such precision in the restrictions might threaten that sense of autonomy. Along these lines, there was also resistance among the Attending MDs to defining such standardized regimens of care. One of the domain experts observed that this cultural resistance was not so much of a problem in other countries, where closer peer review and the use of standardized regimens are generally expected. After some discussion about these issues, it was decided that the abstract goals would
specify the restrictions on the ordering of the Attending MDs’ tasks, from the perspective that these property specifications are meant to describe an ideal world where patient safety was the first concern. The other source of confusion occurred when the domain experts initially responded to the request for additional precision in the property specifications with a return to their original, scenario-based perspective, instead of just an increased focus on the details of the abstract goals. Once this issue was explained, however, the domain experts were for the most part able to return to the goal-based perspective very quickly.

In addition to satisfying some of the computer scientists’ goals for these case studies, formalization of the property specifications also provided significant value to the domain experts. When the computer scientists began to discuss the formalization details with the domain experts, the discussions prompted the discovery of additional properties, implicit assumptions, and in some cases a new understanding of the domain. For example, when selecting precisely which way to interpret certain subtle details in the BT properties, the Nursing professor said that she had begun to see exactly which subtle misinterpretations her students and fellow domain experts sometimes made, at a level of detail that she had not been conscious of before. One property specification where this occurred was BT.D.5, which is stated as the following:

After receiving a blood bank or physician order to obtain a specimen from a patient and immediately before obtaining any specimen from that patient, it must be made sure that the first and last name and medical record number on that patient’s ID band match the first and last name and medical record number on the specimen container label.

One subtle detail that needed to be resolved in the formal specification of this property was what to do in a situation where a nurse or medical technician is drawing multiple blood specimens. A common error in this situation is that the medical professional would check that the first label matched the patient’s ID band, and then would fill all the specimen containers and apply all the labels to them. What the Nursing professor actually meant in this property was that the person should check that each label matched the patient’s ID band before that label is applied to a specimen container. After carefully examining this and other properties at this level of detail, the Nursing professor observed that she was not aware of anyone having officially stated which precise interpretations of these subtle details are correct, and she said that considering these details carefully was very important to arriving at a better understanding of the abstract goals that should be adhered to in any in-patient blood transfusion situation. Such discoveries would often delay the formal specification of the details so that the team could carefully assimilate the new information and make sure that all related property specifications were consistent with any changes that were made. As mentioned previously, this reinforced the need to make sure that the other three issues (i.e., identifying, stating, and organizing the properties) were addressed before the formal specification could be considered complete.
In addition to contributing to an improved understanding of the domain, creating formal property specifications can also be used to aid domain experts in several other ways. The resulting property specifications in these cases studies are a step towards more clearly stating the guidelines and policies to support patient safety in these domains. They also enable the domain experts to communicate the value of precise requirements to colleagues, since the more precise property specifications explicitly state subtle details that are often overlooked or misinterpreted. Because of this, they have also been used to aid in the training of new domain experts. Finally, from the perspective of the larger medical safety project that these case studies are part of, these formal specifications play an essential role in ensuring that the processes in these domains support patient safety.

5.4.4.2 Lessons learned about the use of PROPEL

5.4.4.2.1 Exposure of details required some work for the use of PROPEL to be clearly understood.

The computer scientists used PROPEL as the tool support for formalizing the property specifications, and these experiences provided some valuable lessons about the use of this tool for property elucidation and specification. One important lesson is that the computer scientists need to be careful to focus the domain experts’ efforts on the real core issues that a particular property specification is supposed to describe, and not let every subtle detail that is being exposed by PROPEL divert their attention to side issues. It was noted that PROPEL’s exposure of these details, while necessary, does not guide the specifier in determining how to resolve different interpretations of those details, because it gives no explicit clues about what information is relevant or irrelevant. The computer scientists learned that to reduce confusion among the domain experts on this issue, it is important make sure that each of the individual formal property specifications restricts as little as possible, so that the only restrictions that are described are directly related to enforcing event-orderings that would preserve patient safety with respect to that particular property. All other properties that shared one or more of the same events are considered related, but are not the focus of the current property. The computer scientists therefore made sure to note what restrictions might be related to those being specified in the current property that was in focus, and if there is no property that imposes those other restrictions, both formal and informal property specifications must be added as necessary, to cover those cases.

Depending on the complexity of the properties, there could be over a dozen separate, subtle details that were needed to specify each property precisely using PROPEL, and the domain experts responded to the resolution of these details in different ways. One domain expert said that being required to resolve all the details for precisely how to interpret each property made her feel like she was “being poked by a small child,” in that there were many issues to consider, they were often very subtle, and they made her question “common-sense” things that she normally took for granted in her understanding of the domain. She said
that she had been unaware of how many details had to be carefully explained, but that she had begun to
understand why stating them explicitly was valuable, because there were so many ways to misunderstand the
informal property specifications. Over time, we observed that she became more comfortable resolving the
properties’ details, sometimes anticipating what detail would need to be resolved next when similar property
specifications were being formalized. Another domain expert reported that when she was later working with
her colleagues to more clearly define evaluation guidelines for student nurses, she found herself asking her
colleagues to state precisely what they thought the guidelines should mean, in terms of some of the subtle
details that she had become familiar with while interacting with PROPEL.

The computer scientists also learned that they could not always ask the domain experts to resolve every
subtle detail in the formal property specifications, because understanding the correct interpretation of each
detail was sometimes not based on the domain experts’ knowledge, but on expertise in computer science.
For certain properties, some of the subtle details of temporal constraints did not seem to translate well to the
domain experts, since those details required an abstract understanding of how to characterize sequences of
events. On occasion, when the computer scientists tried to explain why a particular interpretation was correct
or incorrect, the domain experts indicated that the reasoning about events seemed nonsensical, because of
their intimate knowledge of the domain and what they understood as being practically possible in it. For ex-
ample, in the Chemotherapy study, there are two properties, Chemo.D.22 and Chemo.E.2, that are concerned
with stabilizing a patient’s condition. Chemo.D.22 states that if a patient has an adverse reaction to the pre-
chemotherapy supportive care medications, then that patient must be stabilized, and the Chemo.E.2 states
the same thing, except that it replaces the pre-chemotherapy supportive care medications with chemotherapy
drugs. One subtle detail that needed to be resolved in the first property is whether or not a patient’s condition
can be stabilized before that patient has an adverse reaction to the pre-chemotherapy supportive care medica-
tions. The correct interpretation of this detail is that yes, that ordering of the events is possible. Consider the
following example scenario: a patient receives their pre-chemotherapy supportive care medications, does not
have an adverse reaction to them, then receives their chemotherapy drugs and has an adverse reaction to them,
at which point that patient’s condition would has to be stabilized. The computer scientists attempted to ex-
plain to the domain experts how that occurrence of stabilizing that patient qualifies as “before that patient has
an adverse reaction to the pre-chemotherapy supportive care medications;” but were unable to communicate
it in a way that the domain experts understood. Given such awkward interactions, the computer scientists
thus had to learn when to ask the domain experts questions about the subtle details, and when to let such
reasoning be amongst themselves.
5.4.4.2.2 **Different domain experts were comfortable using different property views.** The domain experts responded in a wide variety of ways to the property views shown in PROPEL. All of the domain experts, except for the one with software programming experience, seemed much more comfortable with the informal views, (i.e., the QT property view and the DNL template property view), than with the formal view, (i.e., the FSA template property view). Three of the domain experts (two in the Chemo study and one in the DBSS study) were trained in how to interpret the FSA template view, and all three were able to quickly learn the basics of that formalism. Two of those domain experts seemed to enjoy using this formal view, but the third indicated that the formal view was not a preferred method for reasoning about the properties. The domain expert with previous software programming experience was the only one who used PROPEL without a computer scientist functioning as an intermediary. This domain expert preferred to continue using the QT property view to develop the first draft of the property specifications, once a scope and behavior had been chosen. After developing the first drafts of the property specifications, this person then used the FSA template property view to review and revise those drafts. This domain expert ignored the DNL view for the most part, but looked at it at the end of creating the property specifications. This person said that the DNL template property view might be helpful to show someone who had not participated in creating a property specification, but that it was unlikely that she would ever want to create a property specification using the DNL template property view, because it seemed too informal. Another domain expert had exactly the opposite reaction to the property views: this person did not want to be trained in how to interpret the formal FSA template property view and preferred to only work with the informal DNL template and QT property views.

5.4.4.2.3 **The wording of concepts in the QT and DNL can be confusing.** In addition to seeing how the domain experts preferred to interact with PROPEL, the computer scientists also gained a new appreciation for precision with which they worded questions or named the events used in the formal property specifications. There were some subtle wording pitfalls that made them aware of certain assumptions they had made about the clarity of the QT and DNL template property views. For example, rather than asking the question, “Can A precede B?,” it was instead better to ask, “Should A be allowed to precede B?” The first way to phrase that question often prompted the domain experts to focus on the specific ways that they performed their daily tasks, and they would answer that question by discussing whether they thought that the ordering of the events in the question was practically possible or common. This was an inappropriate focus on the scenario-based perspective, however, and the computer scientists realized that they needed to rephrase that question so that the domain experts could focus their attention on the correctness of the property specifications, independent of any particular scenario or the estimated frequency of that scenario occurring. This and several other places
in the QT and DNL template property views that had awkward or confusing wordings were brought to light through these interactions with the domain experts.

5.4.4.2.4 **PROPEL’s design sometimes required non-intuitive mapping from informal to formal specifications.** Another subtlety in the property specification formalization activities that was sometimes not obvious to the domain experts was the relationship between the informal property specifications and the formal property specifications. As is discussed in Section 5.3, a single informal property specification often mapped to more than one formal property specification. This was one reason why the domain experts could not be expected to use PROPEL directly, because doing so often required expertise in knowing how the various types of mappings should be specified. One such challenge is that there is some overlap in the scope and behavior combinations and thus there sometimes appears to be more than one possibility that looks reasonable. Examples that occurred during the development of the case studies include whether to use an After scope with an Existence behavior or a Global scope with a Response behavior, or whether to use a Between scope with an Absence behavior or a Global scope with a Precedence behavior, where certain secondary events were prohibited from occurring. While there are clear reasons for deciding between these choices, understanding those reasons and applying them appropriately requires some computer science expertise.

Another case where computer science expertise was often required to handle informal-to-formal property specification mappings is when there were multiple logical cases implied by the informal specification. For example, whenever the informal property specification is of the form, “a variable must be in a good state before the guarded event is allowed to occur,” there are two separate formal property specifications required:

- **Case 1:** An occurrence of the bad state must be followed by an occurrence of the good state before the guarded event is allowed to occur.
- **Case 2:** The good state must occur at least once before the guarded event is allowed to occur.

This example is a finer-grained type of AND decomposition than is discussed in Section 5.4.5. This type of AND decomposition was usually not at the conceptual level where the domain experts were working: they tended to work at the level of “a variable must be in a good state before the guarded event is allowed to occur,” defining the AND decomposition of all the “good states” for a given guarded event and expecting that the computer scientists would handle the details required to formally specify that higher-level AND decomposition correctly. The computer scientists decomposed the properties as necessary and vetted the sub-properties with the domain experts by explaining the reasoning behind the decomposition, but the computer scientists did not belabor the point, since the domain experts sometimes seemed confused by the need for such details.
There is another type of formalization subtlety that was not initially obvious to the domain experts: there is sometimes a need to introduce an event into the formal property specification that is not mentioned in the informal property specification. In PROPEL, a property is composed of two parts: a behavior and a scope. The behavior specifies the restrictions on the occurrences of events, and the scope specifies the interval(s) in the event sequences within which the behavior restrictions are required to hold. Certain properties require that an additional event, which is not mentioned in the informal property specification, be used as the scope end delimiter. To illustrate this subtlety, consider the BT.D.1 property:

*After receiving a blood bank or physician order to obtain a specimen from a patient and before obtaining any specimen from that patient, it must be confirmed that that patient has at least one appropriate ID band.*

This example property has two characteristics that require the introduction of an end delimiter event for the property’s scope:

- the behavior must be able to be enforced repeatedly (i.e., it might be required to hold more than once in any given sequence of events).
- the behavior is not able to enforce that repetition by itself.

Since this property’s behavior states that the presence of a patient’s ID band must be confirmed before any specimens can be obtained from that patient, this property demonstrates the first characteristic. The behavior must hold after each time the nurse receives a blood bank or physician order to obtain a specimen from that patient. This property also has the second characteristic, in that after the nurse has confirmed the presence of a patient’s ID band once, the nurse can obtain as many specimens as needed from that patient, and given the limitations of specifying that behavior detail in PROPEL, there is no way for the behavior by itself to enforce the need for a new confirmation at any future point in the sequence of events.

Given such a behavior, the only way to enforce it repeatedly in a formal specification in PROPEL is to compose it with a Between scope. Appendix C.1 shows the formal specification of this example. Recall that the Between scope requires a pair of events, distinct from one another, which are the start and end delimiters that define the restricted intervals within which the behavior is required to hold. The informal specification of this type of property mentions the start delimiter explicitly, but it does not mention the end delimiter because that event would possibly cause confusion by drawing the attention away from the real focus of the property specification, which is that the behavior has to hold after each “important” occurrence of the start delimiter. In the example property here, the domain expert wanted the informal property specification to put the emphasis on how the behavior must hold whenever the nurse receives a blood bank or physician order to
obtain a specimen. From the point of view of the domain expert, there is no particular event that this behavior has to hold before. The main criterion used to select an end delimiter for this type of property is that the event is consistent with the domain expert’s understanding of when the “influence” of an occurrence of the start delimiter “times out” (though this is usually not a well-defined period), and a new occurrence of the start delimiter is important enough start a new restricted interval. The end delimiter that is used in this example property is not interesting in itself, since it mainly functions just as a way to define when a subsequent occurrence of the start delimiter is important enough to start a new restricted interval. This end delimiter is not guaranteed to occur subsequent to an occurrence of the start delimiter: there are actually a variety of possible end delimiters for this property, depending on the clinical situation, but the event chosen is the most common one encountered in the in-patient context that is assumed in the BT study’s domain boundaries.\textsuperscript{11}

For all these reasons, it is sometimes necessary to introduce an event into the formal property specifications that was not mentioned explicitly in the informal property specifications. It is at best an inelegant way to enforce the repetition of a behavior within the current limitations of PROPEL, and such subtleties as these meant that the domain experts could not be expected to interact directly with PROPEL when creating many of these property specifications.

5.4.4.3 A formal specification can have more or less detail, depending upon its intended use

Another observation that can be made about formal property specifications is that having more than one semantically-equivalent version of the formal specification is sometimes important. Property specifications might need to be more verbose to improve their clarity for system developers, but can be minimized for automated-analysis purposes. For example, one of the subtle details that sometimes must be resolved in PROPEL is how to handle events of secondary interest in a property’s behavior. One of the goals of the larger medical safety project is to produce property specifications that can be used in automated analyses of various aspects of the case studies’ domains. Because of this, one of the issues taken into consideration in the property specifications is how to minimize the properties’ alphabets (i.e., the set of events that each property is concerned with) so that the analyses can run more efficiently. Since inclusion of unrestricted secondary events in a formal property specification can make analysis models larger without improving analysis accuracy, the computer scientists know that completely unrestricted secondary events can be elided from the formal property specifications without loss of information. This is not the only perspective on this issue, however. This analysis-efficiency consideration has to be weighed against the domain experts’ desire to specify the properties as clearly and as carefully as possible, and that desire sometimes results in alphabets that are not

\textsuperscript{11}It is possible that an event disjunction could be better used as the end delimiter in this property specification, but that is not necessarily the clear choice, given other interactions with the domain expert.
minimal. This non-minimality occurred when the domain experts wanted to include secondary events that were not prohibited from occurring at any point in the sequence of events. Having such “extra” events is not ideal for doing analyses, but from the perspective of the domain experts, explicitly stating that events are always allowed to occur is an important part of clearly describing some of the properties. For example, in the BT study, there are ordered AND decompositions where the sub-properties are described as a series of checks that have to be performed. The domain expert who vetted the following property specifications wanted to make sure that all the checks are allowed to occur. Consider the following example decomposition for properties BT.D.1-6:

After receiving a blood bank or physician order to obtain a specimen from a patient and immediately before obtaining any specimen from that patient, the following activities must be performed:

1. it must be confirmed that that patient has exactly one ID band.
2. that patient’s stated first and last name and birth date must be obtained.
3. it must be made sure that the first and last name and birth date on that patient’s ID band match that patient’s stated first and last name and birth date.
4. it must be made sure that the first and last name and medical record number on that patient’s ID band match the first and last name and medical record number on the order to obtain a specimen.
5. it must be made sure that the first and last name and medical record number on that patient’s ID band match the first and last name and medical record number on the specimen container label.
6. the activities described above must occur in that order.

Let’s focus on just the first check described in this decomposition. The domain expert working on this BT.D.1 property specification wanted all of the subsequent checks in the decomposition to be explicitly allowed to occur between the primary events in BT.D.1’s behavior. In PROPEL, this information is specified by making the subsequent checks be secondary events in the property specification. See Appendix C.1 for the formal specification of this property. From the computer scientists’ knowledge of the formalism used to specify this property, not including those secondary events in the property’s alphabet would be the same as explicitly specifying them and then not restricting them at all, but from the domain expert’s perspective, it is not clear that unmentioned events are allowed to occur without restriction with respect to this property. Thus, the domain expert wanted this lack of restriction to be stated explicitly. Since the domain experts should not be required to have expertise in interpreting the formalism used in the property specifications, and
since removing these unrestricted secondary events automatically is not difficult, these events are kept in the relevant specifications and can be temporarily removed from the alphabets when the properties are used for analysis. For a related discussion on this topic, see Section 5.3.2.2.3.

5.4.5 Issues involved in organizing a set of properties

As the case studies progressed, the team identified multiple properties in each domain and it quickly became necessary to impose some organizational structure on the sets of property specifications. One reason for doing this was that organizing a set of property specifications made navigating through it easier. We also observed that seeing similar or related properties grouped together helped us to see if a property was missing from the group, if there were duplicate properties, or if there were important assumptions that were not explicitly specified. In addition, a structured organization was sometimes helpful for seeing similarities between the properties in the different domains, so that both the terminology and the subtle details in the property specifications could be compared and perhaps made to agree, if that was appropriate. There were several types of organizational structures found in these case studies, such as the decomposition of properties into sub-properties, and property groupings based on artifacts, guarded events, medical professionals’ roles, process flow, and the domain experts’ preferences for organizing their domain. It was common to see that a single group of properties was characterized by multiple grouping types. Since these informal property specification documents evolved over the course of these case studies, the organizational structure of the documents sometimes changed, and different grouping types were visible, depending on the level of abstraction that the properties were specified at. All of these issues are discussed in more detail in the remainder of this section.

5.4.5.1 Property groupings

5.4.5.1.1 Groupings based on decomposing a high-level property into finer-grained sub-properties.

There were three different types of property decompositions observed in these case studies, where one relatively abstract property would be decomposed into at least two more detailed sub-properties. The three types of decompositions are an unordered AND decomposition, an ordered AND decomposition, and an unordered OR decomposition. These property decompositions resemble the goal-based AND/OR refinements described in [72]. An unordered AND decomposition of a property means that if all the sub-properties are adhered to, then this is a sufficient condition for concluding that the more abstract property is being adhered to. The following group of properties from the DBSS study, DBSS.A.1-2, is an example of an unordered AND decomposition:
Abstract Property

Before a unit of blood product can be assigned to a patient, that unit of blood product must be successfully crossmatched against that patient’s blood specimen.

Decomposition

Before a unit of blood product can be assigned to a patient:

- an ABO / Rh identification and an antibody screen must be performed on an up-to-date blood specimen from that patient.
- that unit of blood product must be identified as compatible with that patient’s up-to-date blood specimen.
- the activities described above are allowed to occur in either order.

By contrast, an ordered AND decomposition of a property means that all of the sub-properties must be adhered to and, if the events described in all those sub-properties occur in a particular order, then this is a sufficient condition for concluding that the abstract property is being adhered to. The following group of properties from the BT study, BT.D.1-6, is an example of an ordered AND decomposition:

Abstract Property

After receiving a blood bank or physician order to obtain a specimen from a patient and immediately before obtaining any blood specimen from that patient, that patient’s identity and that order must be verified.

Decomposition

After receiving a blood bank or physician order to obtain a specimen from a patient and immediately before obtaining any blood specimen from that patient, all of the following activities must be performed:

- it must be confirmed that that patient has exactly one ID band.
- that patient’s stated first and last name and birth date must be obtained.
- it must be made sure that the first and last name and birth date on that patient’s ID band match that patient’s stated first and last name and birth date.
- it must be made sure that the first and last name and medical record number on that patient’s ID band match the first and last name and medical record number on the order to obtain a specimen.
• it must be made sure that the first and last name and medical record number on that patient’s ID band match the first and last name and medical record number on the specimen container label.
• the activities described above must occur in that order.

In the third type of property decomposition, an unordered OR decomposition, if at least one of the sub-properties is adhered to, then this is a sufficient condition for concluding that the abstract property is being adhered to. The following group of properties from an earlier version of the Chemo study is an example of an unordered OR decomposition:

Abstract Property
Before chemotherapy can be administered to a patient, an Attending MD must certify a treatment plan for that patient.

Decomposition
Before chemotherapy can be administered to a patient, at least one of the following must occur:

• an Attending MD must create a treatment plan for that patient.

• a Fellow MD in training must create a treatment plan for that patient and then an Attending MD must explicitly approve that treatment plan.

Incidentally, it should be noted that while it may be possible to have an ordered OR decomposition relationship between properties, this type of decomposition was not present in any of the sets of properties in these case studies.

We observed one other small variation on the concept of a property decomposition: a generalized decomposition that can be instantiated in any one of a number of more specific contexts. This appears in the VPID study, since the task of verifying a patient’s identity is an essential component of every intervention involving a patient in the medical domain. This study contains an ordered AND decomposition that has a spot where an arbitrary set of additional checks can be inserted, depending on the type of intervention that is to be performed. In all interventions, the checks required to verify a patient’s identity are always required to take place before any of the additional, intervention-specific checks. The following example decomposition, for properties VPID.A.1 - VPID.A.k+1, illustrates this concept:

Abstract Property
Before each intervention involving a patient, that patient’s identity must be verified and all the
intervention materials must be verified.

Decomposition

Before each intervention involving a patient, the following activities must be performed:

- it must be confirmed that that patient has at least one institutional ID band.
- that patient’s stated first and last name and birth date must be obtained.
- it must be made sure that the first and last name and birth date on that patient’s ID band match that patient’s stated first and last name and birth date.
- it must be made sure that the first and last name and medical record number on that patient’s ID band match the first and last name and medical record number on the physician order.
- (one or more additional activities, included as necessary for each particular intervention, can be inserted here)
- these activities described above must be performed in that order.

There are examples of interventions in the other case studies where there are checks that can be inserted into the ordered AND decomposition given above. For example, the following set of checks from the BT study, BT.B.5-6, can be inserted there and these checks are required when preparing to administer an infusion of a unit of blood product into a patient:

- all the information for that unit of blood product must be verified:
  - it must be made sure that the first and last name and medical record number on that patient’s ID band match the first and last name and medical record number on the tag affixed to the unit of blood product.
  - it must be made sure that the blood type and blood product unit number on the tag affixed to the unit of blood product match the blood type and blood product unit number on the unit of blood product.
  - it must be made sure that the expiration date on the unit of blood product has not been exceeded.
  - these verifications described above must occur in that order.
- a baseline single-unit assessment must be done for that patient to make sure that that patient is well enough to receive an infusion.
This set of checks also demonstrates that decompositions can be nested. In this example, a BT-specific ordered AND decomposition is nested within the larger ordered AND decomposition that is in the VPID study.

5.4.5.1.2 Groupings based on artifacts. In addition to the property decompositions, there are several other types of property groupings that occur in these case studies. One type of grouping is centered around particular physical or virtual artifacts in the domains, and these artifacts tend to be represented as the noun phrases in the glossaries. For example, in the BT study, there is a grouping of the properties that involves specimen container labels. There are also examples in the Chemo study, where there is a grouping of all the properties related to the development of the treatment plan and the chemotherapy orders. There are also two separate groupings in that study around the checks necessary for the chemotherapy drugs and those necessary for the pre-chemotherapy supportive care medications, where the sets of properties in each group have a parallel structure.

5.4.5.1.3 Groupings based on guarded events. In the groupings that are based on particular guarded events, those sets of properties function as a kind of checklist for the domain experts. The following example property grouping from the Chemo study, Chemo.B.6-8, is centered around a guarded event:

*Before a patient can sign a consent form:*

- *that patient must have a consult with an Attending MD.*
- *that patient must have chemotherapy teaching.*
- *the activities described above must occur in that order.*

Such guarded events tend to be represented in the glossaries as the verb phrases that are associated with the artifacts and the medical professionals' roles. The domain experts who have expertise in the appropriate roles can use these groups to make sure that no patient safety concerns related to the guarded events are being left out. In the case of guarded events that are part of an ordered AND decompositions, the domain experts can also use these groupings to make sure that the order in which the patient safety concerns are being addressed is correct.

5.4.5.1.4 Groupings based on medical professionals' roles. Closely related to the guarded-event groupings are the groupings that are based on the medical professionals' roles. In these groupings, the focus is explicitly on providing the domain experts, each of which have expertise in at least one of the roles, with a cohesive summary of the properties that that person is best suited to work on. This approach helped to narrow the focus of each domain expert as necessary, and it often led to the domain experts finding missing properties
or noticing implicit assumptions that needed to be examined. This type of grouping is not as visible in the current versions of the case studies (though it does exist to some limited extent in the Chemo study where, for example, Chemo.D.8-13 is primarily the focus of the pharmacists, though the nurses do some of the checks as well, as backups), but it was more common in earlier versions, when it was helpful to sort out the properties based on which domain expert had helped to generate them.

5.4.5.1.5 Groupings based on high-level flow through the process. By contrast, rather than being based on the specific artifacts, guarded events, or medical professionals’ roles, the other two groupings are based on more abstract concepts. These more abstract groupings usually include the previous three types of groupings, nested inside. One of these more abstract groupings is based on a kind of temporal ordering of the properties, such that progressing linearly through the properties roughly corresponds with progressing through the associated processes. For example, the Chemo study includes the groupings “activities required right before chemotherapy is administered,” “activities required while chemotherapy is being administered,” and “activities required after chemotherapy has been administered.” Also, in the BT study, the properties are separated into four main groups, the first three of which are given in this order: “checks that are done before obtaining unit(s) of blood product,” “administration of blood product,” and “the handling of a suspected transfusion reaction.”

5.4.5.1.6 Groupings based on domain experts’ understanding of the domain. The fourth main group in the BT, “checks that are done before obtaining specimen(s) from a patient” contains properties that have events that occur both before and after the administration of a unit of blood product, so this group does not follow the temporal grouping paradigm, but rather fits into this last grouping type. This type of grouping is based on domain-specific concepts chosen by the domain experts. The Chemo study also contains examples of this type of grouping, with some properties organized under “patient eligibility” and “legal constraints.”

5.4.5.1.7 A single group of properties is usually characterized by multiple grouping types. Sometimes a group of properties could be characterized by more than one organizational structure. For example, the following properties from an earlier version of the Chemo study exhibit four different grouping types:

Abstract Property

_Before chemotherapy can be administered to a patient, it must be confirmed that the right drugs are going to be administered._

Decomposition

_Before chemotherapy can be administered to a patient:_

160
• the chemotherapy drugs on the chemotherapy drug labels must match the chemotherapy drugs on the chemotherapy orders.

• the chemotherapy drugs’ doses on the chemotherapy drug labels must match the chemotherapy drugs’ doses on the chemotherapy orders.

• the first and last name, birth date, and medical record number on the chemotherapy drug labels must match the first and last name, birth date, and medical record number on that patient’s ID band.

This early property grouping is an unordered AND decomposition, an artifact-based grouping (where the artifacts are the chemotherapy drug labels), a role-based grouping (where the roles are the Pharmacy and the Clinic RN), and a guarded-event grouping (where the guarded event is the administration of chemotherapy).

5.4.5.2 Organizational structure can change

The organizational structure of the sets of properties in these case studies often changed based on the level of abstraction used in the property specifications. Some of these structures could be seen at the more detailed levels, but were not present in the more abstract sets of properties. For example, in the more detailed set of Chemo properties, the role-based groupings were common, whereas in the more abstract sets of properties the role-based groupings largely disappeared and the guarded-event groupings became more common, since the focus at that point was not so much on who performed a particular check, but rather that the critical guarded events were not allowed to occur unless all the necessary checks had been performed, independent of who was performing them. The following example property grouping, Chemo.D.8-13, shows that the team eventually arrived at property specifications whose level of abstraction is somewhere between the abstract property and the very detailed sub-properties given in the previous section.

Before chemotherapy can be administered to a patient, all of that patient’s chemotherapy drugs must:

• be physically suitable for administration.

• be consistent with that patient’s cancer diagnosis.

• match that patient’s approved treatment plan and verified chemotherapy orders.

• be in doses that are consistent with that patient’s data (i.e., height, weight, laboratory results, previous chemotherapy administration data).

• be correctly prepared for that patient.
• be assigned to that patient.

As can be seen here, some aspects of the detailed sub-properties were retained when the properties were specified at a more abstract level, but the focus is placed on the purpose behind those checks, including new checks that had not been identified earlier. When the focus of this property grouping is shifted in this fashion, it is still an unordered AND decomposition of the original abstract property and it is still a guarded-event grouping that is based on the same event as before, but there is no longer special attention paid to any particular medical professional’s role.

5.4.6 Summary of qualitative observations

While working with the domain experts to elucidate the properties for these case studies, there were four major property specification issues that needed to be addressed: how to identify an abstract informal specification of a property, how to state that informal property specification clearly, how to formally specify that property, and when dealing with multiple properties, how to organize them. These four types of issues were strongly interrelated, and often changes made to the properties prompted the team to revisit each of these issues throughout the course of the case studies.

Within each of these four major issues, there were multiple challenges that needed to be resolved. There were two main challenges to identifying a suitably abstract informal property specification: how to identify the abstract goal that the property should specify and how to find a suitable level of abstraction for that property specification. Effectively addressing both of these challenges depended on having a clear understanding of the three different perspectives that the team members held. The domain experts primarily held a scenario-based perspective at first, and were able to provide detailed examples from a wide breadth of daily experiences in their domains. All of the team members held a process-based perspective, which enabled the domain experts to generalize from their scenario examples and to describe their domain processes to the computer scientists. Using the shared understanding garnered from this process-based perspective, the computer scientists were able to help guide the domain experts towards a goal-based perspective, which was essential to being able to identify important patient-safety properties in the domains. A property can be specified at different levels of abstraction, and awareness of those three different perspectives enabled the team to find a suitable level, one which contained enough detail to be operationalizable but not enough detail to constrain how a process should be implemented. Another key to identifying a suitable level of abstraction for a property specification was the ability to establish domain boundaries, because explicitly stating these boundaries codified many of the domain experts’ expectations for the case studies.

In addition to the challenges of identifying an abstract informal specification of a property, there were also challenges to addressing the other three issues. For example, to state an informal property specification
clearly, it was necessary to address a number of terminology problems by carefully creating a glossary. The team also found it helpful to use several simple techniques to improve the readability and navigability of the informal property specification documents. These techniques also contributed to the development of various types of property groupings. These property groupings provided an organizational structure to the informal property specification documents and helped the domain experts to identify missing properties, duplicate properties, and implicit assumptions that needed to be stated explicitly.

All of these challenges needed to be addressed to be able to formally specify the properties in these case studies effectively. The computer scientists used PROPEL to aid with the formalization, and several lessons were learned about the use of PROPEL and about how to avoid common pitfalls. For example, different domain experts were comfortable using different property views in PROPEL, and providing multiple property views was helpful when communicating with the different members of the team. While there was some initial resistance to formally specifying the properties, it quickly became clear to the domain experts how valuable these formalization activities are. The domain experts reported coming to a new understanding of their domains, which gave them deeper insights into some of the common issues that undermine patient safety efforts in daily practice. Formalizing the property specifications in these case studies also provided the computer scientists with new insights about the use of this approach to property elucidation and specification, and as described in Chapter 7, these insights suggest several possible directions that future improvements might take.

5.5 Threats to Validity

There are several factors that prompt us to view these results with caution. At a high level, it is likely that the larger goals behind the work done for the evaluation of this approach had an influence on the results. Recall that the purpose of these case studies was not only to evaluate our approach, but also to use various Software Engineering techniques in a larger medical safety project to improve patient safety. The two goals competed somewhat, because the more properties that the domain experts were able to formally specify, the more properties would be available for use in other analyses being done in the project. In addition, the process- and resource-elucidation activities, which were occurring in parallel with this evaluation, certainly influenced the aims of the property-elucidation activities. Given this larger context for the work, the expressibility data described in Section 5.3 must be viewed with caution, because we undoubtedly biased the domain experts towards properties that PROPEL could express.

It should also be noted that the case studies are limited in two major ways. Although we worked on five different processes, all of the case studies are in the context of the medical domain, and it is possible—
perhaps even likely, given the differences between the case study results and the results observed in the pattern survey—that evaluating our approach in the context of other domains would yield different results. These case studies are also limited by the number of people who participated in them. The overwhelming majority of the properties were specified by just one computer science graduate student, though some review was done by other computer scientists. Three of the properties were specified by a domain expert, who did so based on the training given by that same graduate student. It is likely that the results are biased by how that student conceptualized the properties.

Likewise, the pool of domain experts is limited in several ways. In total across all of the case studies, we worked with two medical doctors (M.D.’s); two registered nurses (R.N.’s), one of whom was also a Nursing Ph.D.; one pharmacist, who was a Pharm.D.; one administrative staff member; one software development manager; and two laboratory technicians. Only three of those domain experts worked with us closely over the long term, however, and two of those three worked together when interacting with us: they usually did not each interact with us independently. None of the domain experts, with the possible exception of the software development manager, had any prior training in doing RE. As is discussed in Section 5.4, the domain experts with medical expertise had been trained to think about their processes from a very detailed, scenario-based perspective, such that they initially seemed to not be aware of many of the abstract properties in their domains. Though they did quickly learn how to adopt both process-based and goal-based perspectives, those new perspectives were heavily influenced by our interactions with them, and thus are likely to be somewhat biased towards our point of view, which is in turn influenced by our knowledge of the limitations of PROPEL.

In addition to the discussion in Section 5.3.2 about the types of information in the properties that could not be expressed in PROPEL, it should be noted that the tool has several fundamental limitations that may have an impact on the validity of these results. All properties in PROPEL must be specified using an event-based paradigm, which excludes some behaviors that are only meaningful in a state-based paradigm and which makes specification of state information awkward at best and impossible at worst. The formal language that PROPEL currently supports is regular, which imposes certain limits on its expressibility. The only properties that can be expressed in PROPEL are ones that can be built via a composition of one of the scopes and one of the behaviors, and the tool only supports a small number of scopes and behaviors. In addition, PROPEL imposes constraints on the behavior and scope alphabets: the two sets must be disjoint, and the start and end delimiters in the scope must be distinct events. It is likely that our knowledge of these limitations influenced the directions we pursued with the domain experts when we elucidated the properties in their domains. For example, we steered the domain experts away from structural specifications for their domains (e.g., which types of patient data need to be included on an ID band) and instead focused them on behavioral specifications (e.g., what should be done with an ID band and in what order those activities should be done).
Another direction that we took with the domain experts was in using explicit domain boundaries to help determine an appropriate level of abstraction. When the domain experts initially had difficulty answering our questions about the properties, they often said that there were too many variations and exceptions in their domains and thus they couldn’t give us a firm answer about the subtle details in many of the properties. Given this feedback from them, we encouraged them to simplify the domain boundaries. This undoubtedly affected the set of properties that they identified for us, because some exceptional situations and some of the need for resource instance disambiguation were removed from consideration. For example, in the Chemo, BT, and ED studies, the domain experts initially assumed that multiple patients were involved in the properties, and this significantly increased the complexity of the property specifications. Defining the domain boundaries for the case studies such that only one patient was in view made the property specifications much simpler, and thus more likely to be expressible in the current implementation of PROPEL. We tried to mitigate this effect somewhat by eliciting the exceptions that could occur, even within the simplified domain boundaries.

5.6 Summary

Although there are reasons to be cautious about coming to strong conclusions about our approach, there are good indications in both the quantitative results and qualitative observations that this approach does provide effective support for enabling specifiers, including those who do not have expertise in property-specification formalisms, to develop precise and understandable property specifications that they can be confident correctly capture the desired behavior of their intended system. The quantitative results of these case studies show that a high percentage (95%) of the informal property specifications could be formalized using PROPEL, and this promising finding bears out Proposition 1 from Section 5.1. As for Proposition 2, although most of the domain experts did not work directly with PROPEL, we observed that the subtle details that PROPEL required the domain experts to resolve helped to clarify many issues that they had not examined closely before their participation in these case studies, and they carefully vetted each formal property specification. Overall, the domain experts responded very positively to the property elucidation activities and the use of our approach.

There were four major types of property-specification issues that we encountered during the course of these case studies. We encountered challenges while identifying an abstract informal specification of a property, stating that informal property specification clearly, specifying it formally in PROPEL, and when dealing with multiple properties, we encountered challenges while organizing them. These four types of issues were interrelated, and changes made to address one type of issue often prompted changes that addressed the other
types of issues. We found that these issues were not necessarily addressed in one particular order, since some feedback and iteration occurred.

We learned several valuable lessons about the use of PROPEL during these case studies. We found that having the three property views is valuable for communicating with different domain experts, because they were each comfortable with different property views. A few domain experts learned how to reason using the property specification formalism, though only at a surface level. Although most of the domain experts were more comfortable with the QT and DNL property views, we learned that there are several places where we need to improve the wordings of their natural-language text, to avoid confusion. Another important lesson is that we need to be careful to focus the domain experts’ efforts on the real core issues that a particular property specification is supposed to describe, and not let every subtle detail that is being exposed by PROPEL divert their attention to side issues. It was noted that PROPEL’s exposure of these details, while necessary, does not guide the specifier in determining how to resolve different interpretations of those details, because it gives no explicit clues about what information is relevant or irrelevant. In addition, we also found that the domain experts were, for the most part, not able to use PROPEL directly. To address this issue, we need to provide clearer guidance to specifiers for how to select among the scopes and behaviors, and also for how to map between informal and formal property specifications. For a more detailed discussion of these issues, see Chapter 7.

Finally, we found that the domain experts’ understanding of their domains changed significantly through their participation in these case studies. The domain experts initially had a fractured picture of their overall processes, because each person was primarily focused on their own subset of a process and often made incorrect assumptions about other subsets of the process. Over the course of these case studies, the domain experts reported developing a better understanding of problem areas in the processes, as they learned the process- and goal-based perspectives. They learned these perspectives quickly and were able to generalize process information from detailed scenarios and to identify abstract goals behind their processes. Throughout these case study evaluations, the domain experts provided invaluable feedback to us and they often stated that they were getting a lot of value from critically thinking about the properties. These responses are encouraging and indicate that our approach to property specification is a promising one.
CHAPTER 6
DNL TRANSLATION STUDY EVALUATION

In addition to the case studies, we also performed a small study to evaluate how well people understood the DNL template property view. By providing a non-formal English description of a formal view, the DNL view is intended to help PROPEL specifiers to understand the formal specifications of their properties without needing extensive expertise in the formalism. In developing the DNL template property view, we tried to select phrases that were precise enough to describe the corresponding FSA with little ambiguity. There is an obstacle to the DNL template property view being able to achieve this goal, however. The DNL template property view is designed to describe all the property details as precisely as possible in natural language, and because of the level of detail that seems to be necessary to express each property detail carefully, a DNL property specification can be long and legalistic. The DNL template property view can thus be a challenge to read, so we wanted to see if other people interpreted the DNL template property view in the way we intended. It is hard to get a definitive answer on a question regarding the human understanding of language, especially with limited resources, but we did this study to at least get a sense of how understandable the DNL template property view is.

To evaluate in measurable terms how well we achieved the goal of understandable DNL, we conducted a study in which we asked a group of people with a background in Computer Science to translate DNL property specifications into equivalent FSAs. Then, we measured the effectiveness of the DNL property specifications by comparing the FSAs that the study participants created to the FSAs that those DNL property specifications were intended to describe. The underlying assumption of this experimental design was that if people can use a DNL property specification to build an accurate FSA property specification, then that DNL property specification is achieving its goal of being understandable.

It should be noted that the DNL that was used in this study is not the DNL that is described in Appendix A.3. The DNL described in that appendix is a newer version that was created in large part because of the results of this study. The problematic parts of the older version of the DNL that was used in this study are discussed in this section, and each problematic part was revised when the newer version of the DNL was created. Not all differences between that older version of the DNL and the newer version were motivated by the results of this study, however. A discussion of the reasons for the other changes is given in Section 3. In
Table 6.1. Property Complexity Categories

<table>
<thead>
<tr>
<th>Complexity category</th>
<th># option settings</th>
</tr>
</thead>
<tbody>
<tr>
<td>Global scope, 1-event behavior</td>
<td>2 or 3</td>
</tr>
<tr>
<td>After/Before scope, 1-event behavior</td>
<td>3, 4, or 5</td>
</tr>
<tr>
<td>Between scope, 1-event behavior</td>
<td>6, 7, or 8</td>
</tr>
<tr>
<td>Global scope, 2-event behavior</td>
<td>8 or 9</td>
</tr>
<tr>
<td>After/Before scope, 2-event behavior</td>
<td>9, 10, or 11</td>
</tr>
<tr>
<td>Between scope, 2-event behavior</td>
<td>12, 13, or 14</td>
</tr>
</tbody>
</table>

this chapter, we discuss our methodology and the results of performing this evaluation of the DNL template property view, and we end with a summary of our observations.

6.1 Methodology

The research question that we were interested in addressing in this study can be expressed by the following proposition:

**Proposition.** The study participants will be able to read a DNL property specification and, based solely on that description of the property, create an FSA that accepts exactly the same language as the FSA that the DNL property specification is intended to describe.

In addition, we were also interested in finding out which parts of the DNL template property view seemed to be clear enough to allow the participants to create an accurate formal specification of the described concepts and which parts of the DNL template property view needed to be improved.

For every PROPEL FSA, there is an associated DNL property specification that was designed to express the same selection of scope, behavior, and option settings. We thus assumed that the language accepted by the PROPEL FSA was the one that the associated DNL property specification was designed to describe. To evaluate whether our proposition held without making unreasonable demands on our study participants’ time, we decided to give every person in our study four DNL property specifications to translate. The current implementation of PROPEL supports over 1,900 properties with its property views, but given the limitations on the number of people who would participate, we were not able to cover every possible property in our study. We thus had to carefully choose which combinations of scopes, behaviors, and their option settings to give the participants to translate, so that the study would cover a representative sample of the available DNL property specifications.

To identify this representative sample, we first assigned each scope-and-behavior combination a property-complexity category based on the number of option settings in the combination. Table 6.1 shows the categories and the number of option settings in each. The “or” in the “# option settings” column for a category
indicates that the property templates that fit into the category can have different numbers of option settings. It should also be noted that the choice of scope and the choice of behavior were counted as “options” for the purposes of this complexity categorization. The assumption in this categorization system is that property templates with more option settings would be considered “more complex” than property templates with fewer option settings, and thus we would expect the more complex property templates to be more difficult to express correctly as FSAs than less complex property templates. Once we had categorized the property templates in terms of potential translation difficulty, we selected a representative sample by systematically choosing fully-instantiated properties from each of the different property-complexity categories such that we covered each type of scope, each type of behavior, and every option setting at least once.

Another issue we needed to address was how to measure the translation accuracy of a participant’s FSA. For each FSA that a participant created, we estimated how closely that FSA matched the PROPEL FSA that was associated with the DNL property specification that the participant was given.\(^1\) We considered a participant FSA to be “close” to the PROPEL FSA if the participant FSA could be transformed into the PROPEL FSA by changing no more than one transition\(^2\) (i.e., adding it to the FSA if it was missing, changing its label, or changing its destination state) or changing the accepting status of one state. While it is possible to compute the difference between the two FSAs’ languages, it is not clear how that difference can be used to measure closeness, so we used this rough transformation measure instead.

Once we had selected a representative sample of properties and had defined how we would measure translation accuracy, we randomly assigned every participant four of the DNL property specifications from the representative sample, each from a different property-complexity category. In particular, every participant was given a DNL property specification from the simplest category, where the property had a Global scope and a 1-event behavior. This simple type of property was only given to the participants to make sure that they understood the translation task, so the results given in Section 6.2 do not include the results for this simple type of property. In addition to the simple property, the remaining three DNL property specifications assigned to each participant varied in complexity. Each property was assigned to only one participant, and the overall set of properties included in the study covered all of the option settings, although obviously not all possible combinations of those settings. If participants could not translate the simplest kind of property with 100% accuracy, then it would be likely that, due to reasons other than the accessibility of the DNL (e.g., they did not understand FSAs or English well enough), the participants were unable to translate the DNL property specification into an equivalent FSA. For such participants, their results would likely be biased

---

\(^1\) These two types of FSAs are hereafter referred to as a “participant FSA” and a “PROPEL FSA,” respectively.

\(^2\) Multiple transitions from one state to another are treated as one transition with multiple labels.
towards low translation accuracies as compared to the participants for whom FSAs and English were not a problem. Conversely, the nature of the participants’ prior familiarity with PROPEL could also introduce variability into the results, because those familiar with our approach would likely have results biased towards high translation accuracies.

To control for these potential sources of variability in the results, we instituted several measures in the experimental protocol. First, we drew our participant population sample only from graduate students and technical staff in our Computer Science department, which provided some commonality in the participants’ familiarity with English and FSAs. Second, every participant was given a translation packet which contained (1) a short survey that asked the participant to rate their familiarity with English, FSAs, and PROPEL on a five-point scale; (2) a formal definition of FSAs and a description of the alphabet that would be used for all the DNL property specifications given in the packet; (3) an example property translation that showed the graphical components of an FSA; (4) four DNL property specifications for the participant to translate (see Appendix D.3 for a sample DNL property specification); and (5) three blank pages for comments or questions. Items (2) and (3) (see Appendix D.2) were included in the packet in the hopes that this information would help to standardize the FSA notation the participants used and thus make our interpretations of their responses less error-prone. Similarly, item (5) was included in the packet so that the FSA translations would not be confused with any comments or questions that the participants might write down.

The participants’ experimental protocol was as follows. Participants were asked to first answer the three questions in their familiarity survey and then read items (2) and (3), which were provided for information purposes only and did not require any response. Participants were asked to work on their four DNL property specifications in the order in which the specifications were given in the translation packets. For each DNL property specification, the participant’s task was to carefully read the specification and then draw the FSA that they thought was being described. The participants were allowed to re-read the DNL property specifications at any time. Once the study had begun, the participants were not allowed to speak either to us or to any of the other participants until after the time allotted was completed. The participants were asked to write their comments or questions only on the three blank pages in the back of the translation packet. The study was conducted for 75 minutes.

6.2 Observations

The experimental results of this study, which are shown in Table 6.2, indicated that 40% (17 of 42) of the participant FSAs were an exact match for the PROPEL FSAs and 64% (27 of 42) were close to the PROPEL FSAs. For each property complexity category, Table 6.2 gives the total number of DNL property
Table 6.2. DNL Study Experimental Results

<table>
<thead>
<tr>
<th>Complexity Category</th>
<th>Total # of Properties</th>
<th>Exact Match</th>
<th>1 Error</th>
<th>% Close</th>
</tr>
</thead>
<tbody>
<tr>
<td>After/Before scope, 1-event behavior</td>
<td>6</td>
<td>5</td>
<td>1</td>
<td>100%</td>
</tr>
<tr>
<td>Between scope, 1-event behavior</td>
<td>8</td>
<td>1</td>
<td>2</td>
<td>38%</td>
</tr>
<tr>
<td>Global scope, 2-event behavior</td>
<td>14</td>
<td>9</td>
<td>3</td>
<td>86%</td>
</tr>
<tr>
<td>After/Before scope, 2-event behavior</td>
<td>8</td>
<td>2</td>
<td>3</td>
<td>63%</td>
</tr>
<tr>
<td>Between scope, 2-event behavior</td>
<td>6</td>
<td>0</td>
<td>1</td>
<td>17%</td>
</tr>
<tr>
<td><strong>TOTAL</strong></td>
<td><strong>42</strong></td>
<td><strong>17</strong></td>
<td><strong>10</strong></td>
<td><strong>64%</strong></td>
</tr>
</tbody>
</table>

specifications that the 14 participants were given to translate, how many of the participant FSAs were an exact match for the PROPEL FSAs, and how many of the participant FSAs were close to the PROPEL FSAs.

Although these percentages are low, the inherent difficulties of accurately translating from natural language to FSAs, even for simple properties, would likely bias the results. The more complicated properties, specifically those with a Between scope, had particularly low translation accuracy rates. Only 1 of those 14 FSAs was an exact match, and only 3 of them were a close match. Although the DNL description of the Between scopes could probably be improved, it is likely that the subtleties of this complex scope make translating any description of it very difficult. For example, 5 of the participants’ 11 Between scope FSAs that were deemed not close omitted a state that even we initially missed when we developed the PROPEL FSA template property view. What these observations tell us is that finding DNL that expresses the Between scope clearly is difficult, but that for the rest of the scopes and behaviors, in most cases this group of study participants interpreted the DNL in the way we intended.

There were three notable places, however, where there may have been serious problems with the DNL template property view. The worst-faring option setting required a scope’s restricted interval to begin after the last occurrence of the starting delimiter: only 5 of the 12 DNL property specifications with this option setting were translated in the way we had intended. The second problematic scope option stated that a restricted interval did not exist if the ending delimiter did not occur: only 6 of the 8 DNL property specifications with this option setting were translated in the way we had intended. The results for these two option settings were not a surprise because these were the two most difficult option settings to describe in the DNL property view. There was one other place where participants noted an issue they had with translating the DNL property specifications into FSAs. The DNL phrase that we used to express the core concept of the Response behavior, “A causes B” was noted as unclear by four of the participants. Even having considered that phrase unclear, all four participants translated it as we had intended.

In evaluating these problematic parts of the DNL template property view, it should be noted that the precision of the DNL template property view may not be the only cause of translation errors. First, some of the
more complex properties can be very difficult to express correctly because the combinations of those scopes and behaviors may introduce subtle interactions that may not be immediately evident even to a participant with high familiarity with both English and FSAs. Second, a participant might understand the DNL template property view, but make an unintended error in their FSA translation. To be conservative, however, we assumed that all errors were due to a failure to understand the DNL template property view. In all three of the problematic parts of the DNL template property view that were described above, it was clear that the DNL template property view of those concepts needed to be improved, and we tried to address these issues when we created the current version of the DNL template property view, which is given in Appendix A.3.

6.3 Summary

While the DNL template property view is intended to help PROPEL specifiers to understand the formal representations of their properties without needing extensive expertise in the formalism, we do not advocate that the DNL template property view be used by itself, without the FSA template property view also available. This is because, while we have strived to reduce the ambiguity in the DNL phrases, there remains an inherent ambiguity in the natural-language descriptions that might not be possible to avoid. In addition, if a person is not an expert in the natural language used, the simpler formal language of the FSA template property view may prove helpful for clarification purposes, assuming of course that the person is familiar with FSAs.

It is because of these inherent difficulties in translating from natural language to a formal specification, even for simple properties, that PROPEL provides multiple property views that are automatically kept linked together whenever changes are made to the property specification. It is important to remember that specifiers are not expected to manually translate from the DNL template property view into the FSA template property view when developing a property in PROPEL. The tool automatically keeps the DNL template, FSA template, and QT property views consistent with each other and it is designed to help specifiers resolve misunderstandings by making it possible to compare the property views. Given all the potential pitfalls to accurate translation and the supportive context in which the DNL template property view is meant to be used, we believe the percentage of close FSAs that we observed in this small study indicates that the DNL template property view is a promising approach for supporting precise and understandable property specifications.
CHAPTER 7
CONCLUSIONS

Property specifications are often used in RE to describe important aspects of what a system is supposed to do. These specifications can then be used as the basis for system development and validation. Ideally, property specifications should be precise enough to support automated analysis and clear enough to be readily understood by system developers. Automated validation tools typically accept property specifications that are represented in mathematical formalisms, such as temporal logic. Such formalisms have not been widely adopted by system developers, however, because their use requires significant expertise. In practice, requirements engineers tend to write requirements specification documents in natural language. While natural language may offer understandability, properties written with such informality are often ambiguous and thus are of limited value when doing rigorous analysis of the system. In addition, no matter what notation is used to describe properties, it is surprisingly difficult to represent even relatively simple properties correctly, since there are often subtle, but important, details that need to be considered. Overlooking these details often leads to inaccuracies that are not revealed until verification or testing, or perhaps even deployment. System developers may invest considerable effort trying to make sure that the system conforms to a property, only to later determine that the property has been specified incorrectly.

We have developed an approach that aims to guide specifiers through the process of creating property specifications that are both understandable and mathematically precise, by providing property pattern templates that leverage the experience of property specification experts. These templates explicitly indicate the variations that must be considered, thereby ensuring that important subtle details in the properties are not overlooked by specifiers. While developing their intended property, specifiers can work with three different property views that are together designed to provide precision, understandability, and guidance. To support this approach, we have developed the PROPEL tool, which keeps all three property views and a scope view synchronized as specifiers make changes to their intended property, and enables specifiers to manage multiple property specifications.
7.1 Observations

We have completed two different types of evaluations of this approach and the results are promising. In one type of evaluation, we used PROPEL to specify properties in five case studies in the medical domain. The results from our case studies showed that PROPEL could handle 95% of the properties that we encountered. It is interesting to note that of the five case studies, only one was specifically chosen to evaluate PROPEL. The other four case studies were part of a larger investigation into ways to reduce medical errors. Although the property patterns covered 92% of the properties examined in [82], those properties were mostly selected from the finite-state verification (FSV) literature, which was the domain used to help develop the property patterns. In our case studies, however, the medical professionals who acted as our domain experts had no knowledge of FSV or RE. Thus, it is somewhat surprising that we achieved similarly promising results, although we currently support only four of the property pattern behaviors with our templates.

For most of the properties in these five case studies, computer scientists formulated them based on discussions with the domain experts and then the domain experts vetted the resulting property specifications. In a few cases, one domain expert worked with PROPEL directly, while a computer scientist looked on. Although most of the domain experts did not work directly with PROPEL, we observed that the subtle details that PROPEL required the domain experts to resolve helped to clarify many issues that they had not examined closely before their participation in these case studies. The domain experts were taken aback by the amount of detail required to precisely capture medical guidelines. In fact, one nursing faculty member remarked that working with us on this project significantly changed the way she views and teaches medical procedures. She said that the subtle details that PROPEL exposes have helped her to better understand why certain medical guidelines are commonly misinterpreted. Similarly, a pharmacist reported that he had become more aware of how ambiguities in the guidelines posted in his workplace could be misinterpreted and could lead to errors. We also learned that having multiple property views was valuable for communicating with a variety of domain experts, because they were each comfortable with different property views. Overall, the domain experts involved in these case studies responded very positively to the property elucidation activities and the use of our approach.

In addition to gathering data about how many of the properties PROPEL could handle and how the domain experts responded to the use of our approach, we also encountered four major types of property-specification issues during the course of these case studies. We learned valuable lessons about identifying an abstract informal specification of a property, stating that informal property specification clearly, specifying it formally in PROPEL and, when dealing with multiple property specifications, what to consider when organizing them. Most of these lessons are not just limited to the use of PROPEL, but are likely to be applicable in many situations where RE approaches are employed.
There are several factors that cause us to view these encouraging results with caution, however. Due to the larger goals of the medical safety project, of which our work was a part, and due to process-elucidation activities that were being done in parallel with these case study evaluations, it is likely that we biased the domain experts towards properties that PROPEL could express. These case studies were also limited in two major ways. They are all in the context of the medical domain, and only a small number of people participated in them. In addition, although the overall coverage of the properties in the case studies is high, there are a number of limitations on PROPEL’s expressibility that prevented complete coverage of the properties and that influenced the types of properties that we encouraged the domain experts to explore. Certain concepts cannot be expressed in PROPEL yet, such as multivariate state information to define a scope, N-1 Chain Response behaviors [81], event conjunctions, and event bounds. Other concepts can only be expressed somewhat awkwardly in PROPEL, such as event disjunctions, single-variable state information, a variable number of both allowed and prohibited secondary events, and real-time constraints.

In the other type of evaluation, the small study involving the DNL template property view, the results were also promising. For 64% of the DNL property specifications in this study, the 14 computer scientists could exactly or almost exactly formulate an FSA representation of the intended property based on the DNL property specifications used in PROPEL. Mistakes tended to be made when the Between scope was involved. Although we have since reexamined the phrasing for this scope, there is no doubt that it is a difficult concept that results in a complex FSA. We may just have to acknowledge that the Between scope cannot be represented succinctly or quickly comprehended in its entirety. Although we have strived to reduce the ambiguity in the DNL phrases, there remains an inherent ambiguity in the natural-language descriptions that might not be possible to avoid. It is partly because of these inherent difficulties that PROPEL provides multiple property views that are automatically kept synchronized, and it is important to remember that the DNL template property view is designed to be used in this supportive context.

7.2 Future Work

Although the results of these evaluations are promising, it is also clear that there are several areas where additional work is needed. There are a number of issues with how our approach supports the property-elucidation process and how specifiers interact with the property and scope views, especially as they are currently implemented in PROPEL. There are also many possibilities for how to expand the expressibility of what our approach currently supports, such as additional property pattern templates, different property-specification paradigms, and various kinds of event compositions. In addition, more evaluation of this approach is needed, both in terms of case studies in other domains and in terms of different types of experiments.
to test the effectiveness of all aspects of our approach. We discuss each of these issues in the following sections.

7.2.1 User-Interface Issues

7.2.1.1 Property-decomposition guidance

One challenge that we observed during the case study evaluation is that the use of our approach sometimes requires a non-intuitive mapping from an informal property specification to a formal property specification. This was one reason why most of the domain experts could not be expected to use PROPEL directly, because doing so often required expertise in knowing how the various types of mappings should be specified. There are two aspects to this expertise that warrant further exploration. One aspect involves how to decompose an informal property specification into the appropriate scope and behavior templates. The other aspect involves how to decompose an informal property specification into multiple formal property specifications. Both types of knowledge are necessary to use PROPEL effectively, and PROPEL does not currently provide any explicit guidance in these areas; specifiers are expected to understand these issues before they begin specifying a property in PROPEL. Some possibilities for future work in this area are discussed below.

7.2.1.1.1 It can be challenging to decompose an informal property into a scope and a behavior. One reason why this can be challenging to specifiers is that there is some overlap in the scope and behavior combinations and thus there sometimes appears to be more than one possibility that initially looks reasonable. Although PROPEL allows specifiers to explore different combinations, it is possible for specifiers to associate their events with the parameters in a way that makes it difficult to find the appropriate scope and behavior combination. In addition to addressing that particular weakness (see Section 7.2.1.4), there may be alternative ways to elicit the appropriate scope and behavior templates from specifiers. For example, it may be possible to use the events that specifiers define to automatically generate event-sequence scenarios and then ask specifiers to indicate whether each scenario should be accepted or rejected, similar to the work presented in [68]. These generated scenarios could be designed such that specifiers’ responses distinguish between the different possible similar scope-and-behavior combinations until specifiers arrive at just one combination, at which point they could be shown the current property and scope views supported by PROPEL and could be asked to resolve the options in the selected combination.

7.2.1.1.2 It can be challenging to decompose a single informal property specification into multiple, more focused, formal property specifications. To support the creation of precise and understandable property specifications in PROPEL, one key aspect of our approach is to limit each formal property specification to describing a very focused subset of system behavior. One outcome of this focus, however, is that
it is common for a single informal property specification to be decomposed into multiple formal property
specifications in PROPEL. One major reason why this decomposition can be challenging to specifiers is that
knowledge of how to employ the multi-property patterns discussed in Section 5.4.4.2.4 requires both a deep
understanding of the expressibility of each of the property pattern templates available in PROPEL and a deep
understanding of the space of event sequences that are relevant to a given informal property specification.
Although PROPEL does not currently provide any explicit support for how to do this type of decomposition,
we expect that it is possible to provide specifiers with additional guidance regarding these multi-property
patterns. Section 5.3.3 describes four types of multi-property patterns: multiple resources, single-variable
state information, ordered AND decompositions, and scope constraints. The case studies’ property specifi-
cations, which are given in Appendix 5, include many concrete examples of these multi-property patterns.
With the exception of scope constraints, it was usually obvious when these multi-property patterns occurred
in the informal property specifications, because consistent phrasing, organization, and annotations, such as
those mentioned in Section 5.4.3.2.3, indicated the appropriate decomposition. Although it is not clear at
this time how PROPEL should offer guidance in this area, since the tool is not designed to do any automatic
analysis of the informal property specifications, it may be possible to provide guidelines for how to spot such
multi-property patterns and to offer specifiers the ability to select a multi-property template, which would
generate the needed set of property pattern templates, treat them semantically as a group, and associate the
specifier-defined events with the appropriate parameters in all the property pattern templates in that group. In
addition, any support that might be added to indicate unresolved options (see Section 7.2.1.2) would need to
be extended to handle this notion of a multi-property group.

There may also be value in exploring domain-specific multi-property templates, to provide additional
guidance. It may be possible to identify common multi-property patterns that occur in a particular domain,
and to make more targeted recommendations for that domain. For example, in the case study evaluation, it
was common for guarded-event AND-decompositions to occur, where the guarded event is the act of per-
forming a medical procedure on a patient, and the guards in the AND-decomposition are the set of safety
checks required before the medical procedure can be performed. If such multi-property patterns can be iden-
tified for a given domain, it may be possible to make specific recommendations regarding which parameters
in the appropriate multi-property template should be associated with which types of domain-specific events.

7.2.1.2 Option-resolution guidance

Another user-interface challenge that we observed during the case study evaluation is that one of the
strengths of our approach can also become a weakness if the approach is not used carefully. The computer
scientists needed to be careful to focus the domain experts’ efforts on the core issues that a particular property
specification is supposed to describe, and not let every subtle detail that PROPEL exposes divert their attention to side issues. PROPEL’s exposure of these details via the options, while necessary, does not guide the specifier in determining how to resolve different interpretations of those details, because it gives no explicit clues about what information is relevant. One possibility for addressing this issue may be to use equivalence classes and boundary-case analysis [232] to provide more guidance to specifiers who are trying to specify a property and aren’t quite sure how to choose among all the options. For example, PROPEL might offer specifiers the ability to see different allowed-event sequences, one of which is made possible by choosing one setting for the given option, and the other of which is made possible by choosing a different setting for that option. Another avenue for exploration may be to give specifiers the ability to indicate that they can’t find the setting that they want, and to provide them with a wizard that would guide them towards choosing a different scope-and-behavior combination, a different association of the events with the parameters, or the creation of additional, separate property specifications.

Another issue that arose with respect to option resolution is that PROPEL provides no quick way to determine whether the options in a given property have been resolved, either at the gross (i.e., all options are / are not resolved) or the detailed (i.e., which options are / are not yet resolved) level. PROPEL currently offers two ways to determine a property’s resolution status. One way is to have specifiers open the property specification and do a careful visual inspection of its property views, which leaves room for human error and is tedious. Another way is to have specifiers try exporting the property specification: if there are unresolved options, PROPEL prevents the export and gives specifiers a list of the unresolved options in an error dialog. Since these two ways require specifiers to go through multiple steps, they do not scale well when specifiers are managing multiple property specifications. Providing specifiers with a one-step way of determining a property’s resolution status would enable them to more easily keep track of which properties need further elucidation. There are several possibilities for how to provide specifiers with a visual cue in the Project Tree View so that they can quickly see a property’s gross resolution status, such as the use of icons, colors, or font faces. There are also possibilities for how to show a property’s detailed resolution status, such as tool tips or inspectors that are accessible from the Project Tree View, or the use of additional icons, colors, or font faces inside the property views. We must be cautious about how we indicate a property’s resolution status, however. As discussed above, PROPEL’s exposure of the many subtle property details can already seem intimidating to specifiers, so it is important that any property-resolution cues that might be added to PROPEL avoid unnecessary visual clutter. It is clear that more work is needed to address these issues.
Table 7.1. Expanded Comparison of Property and Scope Views in Terms of Design Goals

<table>
<thead>
<tr>
<th>Design Goal</th>
<th>Reason for Importance</th>
<th>QT</th>
<th>DNLT</th>
<th>FSAT</th>
<th>ST</th>
<th>SST</th>
</tr>
</thead>
<tbody>
<tr>
<td>Based on natural language</td>
<td>Provides an understandable representation of the property; does not require specifiers to learn a formalism</td>
<td>✓</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Based on a formalism</td>
<td>Provides a precise representation of the property; eliminates ambiguity and can be used in automated analyses</td>
<td></td>
<td></td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>Supports selection of a scope and/or behavior template</td>
<td>Provides guidance through the property-elucidation process</td>
<td>✓</td>
<td></td>
<td>✓</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

7.2.1.3 Scope-focused property views

One area where further work is needed is the ST scope view. Although it provides guidance for selecting a scope template and it enables specifiers to reason about the scope independent of the behavior, it has three major shortcomings. Unlike the property views, the ST scope view does not indicate when the scope options are unresolved, and can thus mislead or confuse specifiers when it appears inconsistent with the scope information displayed in the property views. In addition, the ST scope view does not yet support the single/multiple restricted intervals scope option. Finally, even for those options that are expressed in the ST scope view, not all of the optional components that represent them are clearly indicated as such. Appendix A.4 proposes an alternative version of this scope view that tries to address these three shortcomings by introducing changes and additions to the notation.

Another area where further work is needed involves how PROPEL presents a formal representation of scope information to specifiers. Currently, there is only one view of a property’s scope that is based on an established formalism: the FSA template property view. One limitation of this view is its inability to display a property’s scope independent of its behavior. A property’s scope is conceptually orthogonal to its behavior and it should thus be possible for specifiers to reason about the two parts independently. For complex properties, such as those with a two-event behavior and a Between scope, it can be difficult to distinguish which parts of the FSA are related to the scope and which are related to the behavior, and this can impede specifiers’ attempts to understand and modify their property’s scope. To enable specifiers to reason about their property’s scope independently of the behavior in a view that has the precision of a mathematical formalism, we can consider looking at other specification formalisms.

One possibility for precisely representing a property’s scope might be to introduce another scope view, the Scope Statechart Template (SST), which is briefly sketched out in Appendix A.5. This new scope view is compared to the other views via Table 7.1. Like the ST scope view, the SST scope view also allows specifiers to reason about their property’s scope independent of the behavior, but it has four advantages over the ST
scope view. Its formal notation is an extension of the widely-accepted Statechart formalism [124], so a fullyresolved specification of a scope in this notation might be useful as an input formalism for automated analysis tools. In addition, since the SST scope view is a state-machine formalism, repetition (i.e., the single/multiple restricted intervals option) is easily represented in it. Another advantage is that by using a notation that is very similar to what is used in the FSA template property view (i.e., dashed lines for optional transitions, multi-label items, different colors for changeable components vs. unchangeable components), specifiers can reuse knowledge from their FSA template property view experience when learning how to change option settings in the SST scope view, and vice versa. Finally, there may also be the possibility of representing the behavior templates using statechart templates and exploring the combination of scope and behavior using superstates, allowing specifiers to “zoom in” to view the details of the behavior template and “zoom out” to view the details of the scope template. This nascent scope view is just an initial sketch, however, and more work is needed on it before we could consider including it in our approach.

7.2.1.4 Alphabet Views

In addition to the work that should be done to address issues regarding the property and scope views’ user interfaces, there are also several areas where the Alphabet Views could be improved. One of the assumptions that the Parameter Associations View currently makes is that once specifiers have associated their events with the available parameters, they will be able to use the property views to select the appropriate property pattern template for their intended property. Aside from the property-decomposition guidance issues described in Section 7.2.1.1, the problem of how to select the appropriate property pattern template is made more difficult by the inflexibility of this assumption about the parameter associations. There have been situations where specifiers were unable to find the property pattern template that they wanted because the parameter associations confused them. We need to look into ways to make it easier for specifiers to quickly modify the parameter associations and we need to consider how to provide guidance for which parameter should be associated with which specifier-defined event, perhaps by showing example parameter associations to specifiers, based on the selected property pattern template, and letting them select the one they prefer. Another source of confusion in the Alphabet Views is found in the Alphabet Manager. The way that the list of events in a property’s alphabet is shown in the Alphabet Manager does not distinguish between the primary and secondary events. Distinguishing between these two types of specifier-defined events would be helpful in situations where specifiers have defined a set of secondary events and need to make decisions about the option settings based on that set. In addition to these user-interface issues, there are a number of ways that the Alphabet Views might be expanded to provide support for more expressibility. These expansions are discussed in Section 7.2.2.
7.2.1.5 Project Tree View

Although the Project Tree View provides PROPEL specifiers with the ability to see and change the hierarchical structure of a project, there are some limitations on both capabilities. In the current version of the Project Tree View, specifiers are shown a project- and alphabet-oriented layout of the project structure. Figure 4.3 in Section 4.4.1 shows this layout, where alphabets are displayed as the children of the projects that contain them, and where properties are displayed as the children of the alphabet that they are associated with. This layout can present problems for specifiers when the tree structure under a project or alphabet is collapsed, because specifiers have difficulty finding their desired property if they do not remember which project and alphabet that property is associated with. This problem also exists to a lesser extent when the tree structure is expanded, because the Project Tree View displays the nodes in alphabetical order, and thus the properties’ order is constrained by the projects’ and then the alphabets’ relative order. Since specifiers’ focus is more often on the properties than on the projects or alphabets, presenting only this one inflexible project- and alphabet-oriented layout is not sufficient. Appendix B.1 gives an alternative design for the Project Tree View, which allows specifiers to toggle between an alphabet-oriented layout and a property-oriented layout. In this latter type of layout, properties are shown as the children of the projects, and the alphabet that a property is associated with is shown as the child of that property. This layout is not without its drawbacks, however, since the same alphabet can appear as the child of multiple properties and this might be confusing to specifiers. In addition, the projects still remain the dominant organizing concept. Rather than just providing this strict hierarchical view, it might be desirable to be able to organize properties in a more flexible way, such as with tags, so that a property could be found without requiring knowledge about its project or alphabet. In addition, such tagging functionality might enable the set of properties to be more easily viewed from multiple perspectives. It is clear that more work is needed on this issue.

7.2.1.6 Summary Views

The Summary Views also support the management of multiple property specifications and although these views provide specifiers with the ability to see summaries of the project structure from multiple perspectives, they lack several features that specifiers have requested. Although it is possible to browse the project structure, the tabular format that the Summary Views are displayed in requires a lot of screen real estate and it can thus be difficult for specifiers to quickly grasp the structure of what is being displayed. As was described in Section 4.4.2, a common use for the Summary Views is to enable specifiers to search through a set of properties. Currently, that search must be manual, however, since there is no explicit search support. In addition to adding search capabilities, it may also be important to enable specifiers to create custom Summary Views beyond the five pre-defined Summary Views that we currently support. Some specifiers have indicated
that they want to minimize the need to search through a large project structure and they want to show or hide particular information that they commonly refer to. In addition to providing more effective search and display capabilities, there may also be need to make these read-only Summary Views editable, since many specifiers’ tendency upon first seeing the Summary Views is to try to add comments or to change event, alphabet, project, or property names there, especially since the display makes missing information or inconsistencies more visible than the other PROPEL views do.

7.2.2 Expressibility Issues

Although there is work to be done in addressing the shortcomings of how specifiers interact with PROPEL, there are also a number of intriguing directions for how to add support for new types of properties. Our initial work explored the scopes and the four most common behaviors encountered in the property pattern survey [82], and even though most of the properties that we encountered in the case study evaluation can be covered by those four behaviors, there are many properties that cannot be covered. There are several different ways to expand to support new types of property pattern templates, such as exploring new behavior templates, loosen the restrictions on the scope templates, and exploring other property-specification paradigms beyond the event-based one that PROPEL currently supports. In addition to exploring more property pattern templates, there are also possibilities for new ways of interacting with them, such as providing richer support for how their parameters are associated with specifier-defined events. Beyond just exploring these possibilities, it is also important to widen the applicability of this approach by doing further work on how to translate these property pattern templates into other specification formalisms beyond just finite-state automata. We briefly discuss these ways to expand PROPEL’s expressibility in the following sections.

7.2.2.1 Exploring new behavior templates

When considering new types of properties to support, one area where there is room to expand is into new types of behavior templates. For example, one common issue that PROPEL specifiers have struggled with is how to create a property that draws on both the Response and Precedence behavior templates. Although it is possible to do this to a certain extent using the current Response behavior template, not all combinations of these two concepts are available for specifiers to experiment with. There are also other two-event behaviors that have been observed in practice and should thus be considered, such as two-event Alternation. These behaviors are only partially covered by the current Response and Precedence behavior templates. This begs the question of whether a more generalized two-event behavior template should be considered, one that covers all the two-event behaviors that specifiers can currently express in PROPEL and that introduces some new possibilities. We briefly explored a generalized two-event behavior template using the FSA template.
property view, and the initial results of this exploration are given in Appendix A.2.1.5. In addition to the 96 behaviors that correspond to those supported by the current Response behavior template and the 48 behaviors that correspond to those supported by the current Precedence behavior template, there are 46 new behaviors made possible by this generalized two-event behavior template. 30 of these new behaviors each focus on a different combination of option settings for a subset of the options that are available in the current Response and Precedence behavior templates, but do not correspond to either of those unifying concepts. It is interesting to note that the 16 remaining new behaviors might qualify as a type of two-event Alternation, and they all require that the sequence of events must be started by only one of the parameters (i.e., A). There are other types of two-event Alternation behaviors possible, however, that this generalized two-event behavior template does not cover. There are also a variety of new options to consider for two-event Alternation, such as whether just one or both of the events must strictly alternate, or whether the alternation must be started by one or the other of the two events involved.

In addition to the challenges of understanding the space of behaviors that this generalized two-event behavior template covers, there are also challenges regarding how it should be represented in the property views and how PROPEL’s architecture needs to be made more flexible to support it. For example, although it is relatively easy to expand an FSA template’s structure via the addition of new optional states and transitions, it is not as clear how these additions can be incorporated into the DNL template property view. DNL templates are structured such that they always have a core phrase that describes the unifying concept for all of the behaviors possible based on a given behavior template. Remember that one of the primary design goals for the DNL template property view is to provide an understandable representation of the property. This core phrase is an essential part of supporting understandability, since it concisely states the unifying concept around which the corresponding FSA template and QT property views are structured. One challenge with this new generalized two-event behavior template is that it has no clear unifying concept. It allows specifiers to create a new set of behaviors that have nothing in common save for the fact that they all rely on two primary events. What this challenge implies is that further work is needed to understand how PROPEL can make this generalized two-event behavior template understandable to specifiers. In addition to considering how the property views might be impacted by new behavior templates, there is also the question of how PROPEL should be re-architected such that it can more flexibly handle the behaviors that it supports. Right now, it is rigidly structured so that behavior templates have little to no relationship with each other. As new behavior templates and new combinations of them, such as the generalized two-event behavior template, are explored, the PROPEL architecture needs to be able to adapt more easily to the possible changes in the behaviors’ relationships.
Beyond a generalized two-event behavior template, there are several other behavior templates that we have encountered but that we do not yet support in PROPEL, and there might be value in exploring them as well. One possible direction involves putting more complex constraints on the events in the behavior templates. The behavior templates currently supported in PROPEL enable specifiers to either not constrain event occurrences at all or to constrain events to occur zero or one times. Finer-grained constraints may be necessary in many real-world domains, however. The simplest example of a finer-grained constraint is the Bounded Existence behavior described in the property pattern approach [81, 82]. This behavior requires a given event to occur at least once in the sequence of events and it also puts an upper limit on the number of times that the event is allowed to occur. A behavior template based on this behavior might extend the concept of bounds beyond just a maximum bound to also include a minimum bound, or some kind of conditional bound, such as not requiring the given event to occur, but if it occurs, then requiring it to occur between m and n times.

In addition to finer-grained constraints on event occurrences, another possible direction to pursue is behavior templates that use three or more primary events. One example of such a behavior template is Locking, where an event, A, can only occur between an occurrence of acquiring the lock, C, and the first subsequent occurrence of releasing the lock, D. Although this pattern can be specified by a conjunction of two separate properties in PROPEL:

- A is not allowed to occur before the first occurrence of C
- A is not allowed to occur between an occurrence of D and the first subsequent occurrence of C

this way of specifying it seems overly difficult to understand. Using only the scopes and behaviors currently supported by PROPEL, this conceptually simple pattern has to be specified in terms of a negation: the two properties given above only specify the parts of the event sequence where A is not allowed to occur. Thus, specifying the pattern this way obfuscates the simple Locking concept for anyone who reads the property specifications, because it requires specifiers to go through two mental “hoops” to understand what is being said: thinking in terms of a negation and breaking the concept up into two separate parts. In addition, it is not possible to use secondary events in any of the behavior templates that PROPEL currently supports to express this Locking concept, and even if it were possible, it is not clear that such an approach would provide the right emphasis in the resulting property specification. In Locking, A is not a secondary event; it has an equal primary-event focus with C and D. Specifically, the wording in the DNL template and QT property views would need to be revised to put more of an emphasis on A. Since this Locking pattern is relatively simple and common, it presents an interesting direction that we can explore.
Other examples of behavior templates that use three or more primary events are the variations on Chaining, a concept that was introduced in the property pattern approach [81, 82]. There are two different types of Chaining, $1-N$ and $N-1$, where $N \geq 2$. In $1-N$ Chaining, one event causes or precedes an ordered chain of events. For example, in $1-N$ Chain Response, if event $A$ occurs, then $B$, $C$, and $D$, etc., are required to occur subsequently, in that order. Likewise, in $N-1$ Chaining, an ordered chain of events causes or precedes one event. For example, in $N-1$ Chain Precedence, event $D$ cannot occur until after $A$, $B$, and $C$, etc., occur, in that order. It is also possible to have unordered conjunctions of event occurrences as part of a behavior template, but unordered conjunctions are not considered part of Chaining. Those possibilities are discussed briefly in Section 7.2.2.3. Although it is possible to specify an $N-1$ Chain Precedence using $2N - 1$ Global Precedence properties, doing so can produce a large number of properties, many of which appear redundant. This increases the complexity that PROPEL specifiers have to manage and it thus impedes the understandability of the resulting property specifications. Given the fact that 43% of the properties that we encountered in our case study evaluation are involved in specifying an ordered chain of events, exploring these Chaining behavior templates is of particular interest to us.

### 7.2.2.2 Loosening the restrictions on the scope templates

Although we have not encountered new types of scope templates as we have new types of behavior templates, there are still a number of ways in which we might expand our support for scopes. We have placed two restrictions on the scope templates’ alphabets, to simplify how scope templates are applied to behavior templates: the scope template alphabet must be disjoint from the behavior template alphabet, and if a scope template has both starting and ending delimiters, these must be associated with distinct specifier-defined events. Although there are important reasons why we are reluctant to relax the first restriction (see Section 3.2.1), there have been several situations where specifiers requested that the second restriction be relaxed, so that some form of Alternation could be expressed, for example. Whether this is the right approach to handle Alternation is a separate question, but the implications of relaxing this restriction should be explored. Another direction to consider involves adding support for more scope options beyond the current three. To preserve the distinction between a property’s scope and behavior, we do not yet allow the scope to violate the property, but if we did, it would open up several new possibilities for scope options. There are also open questions regarding whether interscope intervals, the parts of the event sequence when the behavior does not have to apply, could be handled differently. For example, the current assumption is that secondary events are allowed to occur without restriction in those interscope intervals. There may be value in exploring whether specifiers

---

1 See Section 5.3.3.3 for a discussion of why this requires $2N - 1$ Global Precedence properties.
should be given the choice of whether to prohibit secondary events in those intervals, and perhaps even in which ones. Finally, the issue of whether and how to support nested scopes remains open. There are a small number of properties in the case study evaluation where being able to specify multiple nested scopes might have enabled us to specify multivariate state information. It is not clear if this is a helpful direction to pursue, however, since nested scopes and the variations on nesting can grow complex. There is some question as to whether the complexity that might be introduced would come at the expense of understandability. Multivariate state information might be more understandable in a state-based paradigm rather than in an event-based one. These questions all bear further examination.

7.2.2.3 Exploring event patterns and compositions

Beyond just the behavior and scope template possibilities, there are also new areas to explore with respect to a property’s alphabet. One issue that occurred several times in the case study evaluation was the question of how to handle event compositions. One simple type of event composition is when a higher-level event corresponds to a disjunction of more detailed events. Although it is possible to specify something this simple in PROPEL right now, there is no explicit support in the tool for defining an event disjunction. Although we expect that adding such functionality to PROPEL would amount to just adding syntactic sugar, that is not the case for other types of event compositions. One example of an event composition that would not be as simple to add to PROPEL is an event conjunction. Unless it is done carefully, an event conjunction can present problems when it is associated with the parameters in the current behavior and scope templates, because it is not necessarily clear what the conjunction means in an event-based paradigm where none of the events can overlap. The Chaining behavior templates can be thought of as handling one case of an ordered event conjunction, but there are other types of event conjunctions, such as some of the Composite Propositions described in [204], that may need to be explored as well.

In addition to event compositions, there is also the question of how to handle event parameterization, where a single event might refer to multiple instances of an artifact, such as a datum or a physical object, and those instances must be distinguished. We encountered this issue a few times in the case study evaluation and we handled it in a very limited fashion by adding the keyword “\(i\)” to parameterize the event name, but this approach should be more closely examined. There are clear possibilities for using event parameterizations in automated analyses, and there may be value in providing more explicit support for event parameterization semantics in PROPEL.

One possible new direction for specifying restrictions on secondary events was mentioned in the previous section, but there are several others that are also of interest. In addition to the questions regarding whether secondary events are allowed to occur in interscope intervals, there are also questions regarding whether to
restrict secondary events before and after particular occurrences of the primary events. Beyond that kind of finer-grained control over where the secondary events are restricted, there are also issues concerned with specifying two different sets of secondary events. There were a few properties in the case study evaluation where the domain experts wanted to be able to specify both prohibited and allowed sets of secondary events, but this is not possible in the current implementation of PROPEL: the entire set of secondary events in a given property must be either prohibited or allowed. We also encountered the issue of how to specify an abstract description of a set of secondary events when the exact enumeration of the events in that set is not yet known, but the definition of which events belong in the set is known. This occurred when the set of events was still in development as the medical processes were being specified, but the domain experts knew what types of secondary events should be prohibited from occurring and they wanted to specify that set in abstract terms in the properties. This is an interesting direction that needs to be explored as support for properties’ alphabets is expanded.

7.2.2.4 Exploring other property-specification paradigms and languages

We have thus far limited ourselves to exploring behavior and scope templates in an event-based paradigm, using just one property specification formalism, FSAs. There is value, however, in exploring other paradigms and formal languages, for two reasons. One reason, as was shown by the case study evaluation, is that there is a wide variety of common property patterns that we cannot yet express in PROPEL, or that we can only express somewhat awkwardly. It is very likely that expanding to support other property-specification paradigms can address many of these issues. Another reason to consider other paradigms and languages is that specifiers may want to use the property specifications that they create in PROPEL as inputs to automated analysis tools that work with formalisms other than FSAs. For example, there are a number of finite-state verification tools that take the state-based LTL [220] or CTL [88] as their input formalism. In addition to exploring just event-only or state-only paradigms, there are also some intriguing possibilities in a combined event-and-state paradigm. Although such formalisms (e.g., Fluent-LTL [108]) introduce an additional level of complexity, they might also offer a means to reason more naturally about properties that put constraints on both types of information.

Although we encountered some of those specification-paradigm issues in the case study evaluation, an area that is especially relevant in the medical domain is supporting a real-time paradigm. We encountered several properties with real-time constraints on the delay and duration of treatments, and it is essential that such properties are captured accurately if the safety of those treatments is to be ensured. Although work on real-time property patterns has begun [165], there is still room for further exploration of the options in those property patterns and further exploration of which real-time formalisms would best suit specifiers’ needs.
Another type of paradigm shift to explore may be closer to what PROPEL currently supports. As was discussed in Sections 7.2.1.1 and 7.2.1.2, it can sometimes be difficult to identify the appropriate scope and behavior templates, and then to correctly resolve the options once the templates have been selected. It may be possible to make those decisions easier in situations where the intended property is more simply or more naturally specified using nondeterminism or negation (i.e., a “none” property rather than an “all” property), and thus these are areas that should be explored further. Alternatively, there may be value in looking at ways to remain in an event-based paradigm, but to allow multiple events to occur simultaneously. For example, we can go beyond the current assumptions in PROPEL’s event-based paradigm and consider property pattern templates in a non-interleaved model of concurrent computation. Although we have not yet encountered properties where this type of expansion is necessary, it may still be useful to explore these possibilities to see what they may offer.

7.2.3 Evaluation Ideas

Although the initial evaluations of this work have given us valuable feedback about the effectiveness of our approach, there are still several areas where further evaluation is necessary. There is some question as to what types of experiments would exercise different parts of the approach. At the very least, another study on how well people understand the DNL template property view, perhaps by again having study participants translate DNL property specifications into FSAs, would be a valuable way of evaluating the effectiveness of the revised DNL, as compared to the first study. This type of experiment tends to be limited to participants who are familiar with FSAs, however, so it would also be useful to try other types of experiments where familiarity with a property-specification formalism is not a prerequisite for participation. Perhaps one way of measuring the effectiveness of the property and scope views is to show participants sample event sequences and ask if the property will accept or reject them. It may be possible to do this separately for each of the property or scope views, or to do this with multiple views shown together. Of course, as we found in the case study evaluation, it often requires computer science expertise to be able to reason about the subtleties of event sequences, so the prerequisites for this type of experiment might also be too limiting. We might be able to construct the experiment such that the prerequisites are relaxed or mitigated somewhat, for example by separating scope concerns from behavior concerns, but this would need to be done carefully, and it is not clear how generalizable the results would be.

One challenge with the evaluations proposed above is that they focus on the individual property views, and not on the larger process of interacting with our approach as a whole, in real-world situations. As an alternative to those more focused types of evaluations, case studies have been a useful way to evaluate our approach in this larger context. Our case studies thus far have been limited to the medical domain and it
would be valuable to explore other domains as well. In addition to moving into other domains, it would also be valuable to explore a different mix of computer scientist and domain expert participation. It may be necessary to address some of the user-interface issues mentioned in Section 7.2.1 before it would be realistic to expect domain experts to use PROPEL extensively themselves, but case studies of this type are likely to more directly address our essential research questions. Beyond just the use of PROPEL, there are also a number of questions to answer about how we developed the properties and the processes in parallel during the larger medical safety project. For example, should processes be examined first or should properties be examined first? Which order makes it easiest for people to learn how to identify the appropriate properties in their domains? Is the order not the distinguishing factor, but rather how the two types of activities are interleaved? Do people identify properties more effectively after they learn abstract control- and data-flow patterns, or do they need concrete, domain-specific examples of properties instead? These are intriguing questions, and since interesting alternatives to the way that we ran the case studies are possible, these alternatives ought to be explored.

7.2.4 Summary

As the systems that we depend on grow ever more complex, the challenge of creating property specifications that are both understandable and precise is an essential one to address. Property specifications should support communication among system developers, and should also enable them to automatically ensure that the system conforms to the system stakeholders’ needs and expectations. We have developed an approach that addresses this challenge by providing template-based guidance for how to specify properties and by making multiple property and scope views available, such that when the views are used together, they support understandability, precision, and guidance. We used this approach to specify properties in five real-world case studies drawn from the medical domain, and the domain experts who worked with us indicated that this approach offered them valuable guidance as they considered the subtle details of their properties. Although more work is needed, our initial evaluations indicate that this approach to property specification is quite promising.
APPENDIX A

PROPERTY VIEW DETAILS
A.1 Question Trees

See Section 3.1.2 for an introduction to the QT property view, its notation, and its interaction model. Like the other property views, the QT can be used to resolve the options that are associated with the selected property pattern template. Thus, the same options that must be resolved in the FSA template and DNL template property views, and in the ST and SST scope views, are options in the QT property view as well.

A.1.1 Behavior Question Tree

This appendix gives the details of how all four behavior templates can be selected and how their options can be resolved using the Behavior Question Tree (BQT). Central to this discussion is the fully-expanded version of the BQT, given in Figure A.1. There are two aspects of the how the BQT is portrayed in this figure that differ from its presentation in PROPEL. For the purposes of the discussion in this appendix, each of the questions and answers are assigned a unique line ID, although those IDs are not part of the actual BQT. In addition, the BQT cannot be fully expanded in PROPEL, because when an answer to a question is selected and the child questions under it are revealed, the child questions under the unselected sibling answers are simultaneously hidden. For the purposes of the discussion in this appendix, however, all the BQT questions and their answers are shown simultaneously. For a more realistic example of how specifiers interact with the BQT to specify a property, see Section 3.1.2.2.

As can be seen in line 1 of Figure A.1, the first question the BQT asks specifiers about the intended property is, “How many events of primary interest are there in this behavior?” Lines 2 and 7 show the two possible answers to this question. If specifiers select “One event” on line 2, they will be shown lines 3-6, where they have the choice of selecting either an Absence behavior template or a variation on an Existence behavior template. If specifiers instead select “Two events” on line 7, they will be shown lines 8, 9, 13, 17, 21, 24, and 37. Lines 9, 13, and 17 allow specifiers to select either a Precedence behavior template or a variation on a Response behavior template, and lines 21, 24, and 37 allows specifiers to resolve all but one of the options available in those behavior templates. The following sections discuss these choices in more detail.

A.1.1.1 The Response and Precedence Behavior Templates in the BQT

Lines 7-42 in Figure A.1 show the fully-expanded subset of the BQT that allows specifiers to select either the Response or the Precedence behavior template and to resolve each of the selected behavior template’s options. These two behavior templates are concerned with restrictions on the interaction of two primary events, A and B. Specifiers can select between the Precedence and the Response behavior templates by choosing one of the answers under the question, “Which of the following choices best describes how A and
Which of the following choices best describes how A and B interact?

<table>
<thead>
<tr>
<th>Both statements describe how A and B interact: if A occurs, B is required to occur subsequently, and B is not allowed to occur until after A occurs</th>
</tr>
</thead>
<tbody>
<tr>
<td>Is A required to occur?</td>
</tr>
<tr>
<td>Yes, A is required to occur</td>
</tr>
<tr>
<td>No, A is not required to occur</td>
</tr>
<tr>
<td>B is not allowed to occur until after A occurs</td>
</tr>
<tr>
<td>Is A required to occur, whether or not B eventually occurs?</td>
</tr>
<tr>
<td>Yes, A is required to occur, whether or not B eventually occurs</td>
</tr>
<tr>
<td>No, A is not required to occur and, if it does not occur, B is never allowed to occur</td>
</tr>
<tr>
<td>After A occurs, is A allowed to occur again before the first subsequent B occurs?</td>
</tr>
<tr>
<td>Yes, A is allowed to occur again, zero or more times, before the first subsequent B occurs</td>
</tr>
<tr>
<td>No, A is not allowed to occur again before the first subsequent B occurs</td>
</tr>
<tr>
<td>After A and the first subsequent B occur, is A allowed to occur again?</td>
</tr>
<tr>
<td>Yes, A is allowed to occur again</td>
</tr>
<tr>
<td>No, A is not allowed to occur again</td>
</tr>
<tr>
<td>Is B allowed to occur again?</td>
</tr>
<tr>
<td>Yes, B is allowed to occur again, but not until after another A occurs. If another A does occur, the situation is the same as when the first A occurred, meaning that the restrictions described on any events that take place after that occurrence would again apply</td>
</tr>
<tr>
<td>Yes, B is allowed to occur again, zero or more times, whether or not another A occurs</td>
</tr>
<tr>
<td>If another A does occur, is the situation the same as when the first A occurred, meaning that the restrictions described on any events that take place after that occurrence would again apply?</td>
</tr>
<tr>
<td>Yes, if A does occur again, the restrictions described on those events would again apply</td>
</tr>
<tr>
<td>No, even if A does occur again, there are no restrictions on the occurrences of any future events</td>
</tr>
<tr>
<td>No, B is not allowed to occur again</td>
</tr>
</tbody>
</table>

Are there any events of secondary interest in this behavior?

Yes, there is an event of secondary interest in this behavior: it is C1; : they are C1, C2, Cn-1, and Cn

After A occurs, is the event of secondary interest in this behavior allowed to occur before the first subsequent B occurs?

Yes, after A occurs, the event is allowed to occur zero or more times before the first subsequent B occurs

No, after A occurs, the event is not allowed to occur before the first subsequent B occurs

No, there are no events of secondary interest in this behavior

**Figure A.1.** Fully-Expanded Behavior Question Tree (with line numbers added)
B interact?”, on line 8. The answer on line 13 is the one specifiers can select to get the Precedence behavior template, and specifiers can select either of the other two answers (on lines 9 or 17, respectively) to get the Response behavior template. There are two answers that lead to the Response behavior template because that is how the BQT represents the two different settings for the Precedency option in the Response behavior template. There is no Precedency option in the Precedence behavior template, but the remaining six options are shared between the two behavior templates and are represented by the questions and answers in the rest of the BQT.

The Nullity option is in both of the behavior templates, but it is represented in the BQT in three places: on lines 10-12, lines 14-16, and lines 18-20. This multiple placement is necessary in the BQT because, to reduce confusion that was observed when specifiers tried to resolve this option (see Chapter 5), the phrasing of this option’s question and answers differs depending on whether it is being shown as part of the Response behavior template or as part of the Precedence behavior template. The phrasing of the Nullity child question, “Is A required to occur?” and its answers are the same on lines 10-12 and lines 18-20, because those are both part of the Response behavior template. The phrasing of the Nullity child question on line 14 is different however, instead reading, “Is A required to occur, whether or not B eventually occurs?” Likewise, that question’s answers on lines 15-16 are changed to match the different phrasing.

The BQT represents the remaining five shared options in the Response and Precedence behavior templates as follows. The Pre-arity option is represented on lines 21-23 with the question, “After A occurs, is A allowed to occur again before the first subsequent B occurs?” and its two answers. The three options Post-arity, Finalization, and Repeatability can constrain each other, depending on their respective settings, and thus the BQT represents their questions and answers as nested together, on lines 24-36. The Finalization option is represented on lines 24, 25, and 33 with the question, “After A and the first subsequent B occur, is A allowed to occur again?” and its two answers. Under both of those answers, the Post-arity option is represented on lines 26-28, 32, and 34-36 with the child question, “Is B allowed to occur again?” Making this a child question under the Finalization question on line 24 allows the possible combinations of Repeatability and Post-arity to be revealed in the only place where the Repeatability option makes logical sense: the context where the specifiers have chosen that after A and the first subsequent B occur, A is allowed to occur again (i.e., when specifiers select the answer on line 25). In the other setting to the Finalization option, on line 33, where A would not be allowed to occur again, it does not make sense to ask specifiers to resolve the Repeatability option, since that option is only concerned with what happens after A occurs again. Returning to the context where we assume that specifiers have selected the answer on line 25 and have thus revealed the child question on line 26, we can see the four possible combinations of the Repeatability and Post-arity options: those combinations are the four leaf nodes in that subtree, on lines 27, 30, 31, and 32. The question
One event

Which of the following choices best describes the restriction on A?

- A is never allowed to occur
- A is required to occur at least once
- A is required to occur exactly once

Figure A.2. Absence and Existence Behavior Templates in the BQT

on line 26, “Is B allowed to occur again?” has three answers, on lines 27, 28, and 32. The answers on lines 27 and 32 are two of the possible combinations of the Repeatability and Post-arity options. The answer on line 28 has another child question (see line 29), which itself has two answers (lines 30 and 31), and those are the other two possible combinations of the Repeatability and Post-arity options.

The Immediacy option is represented on line 39 in the BQT, as a child of the question on line 37 “Are there any events of secondary interest in this behavior?” and its answer on line 38, “Yes, there \{is an event / are events\} of secondary interest in this behavior \{it is C_1 / they are C_1, C_2, ..., C_{n-1}, and C_n\}.” Although the question on line 37 about the existence of secondary events does not itself correspond to an option in either of the behavior templates, its presence is designed to improve the usability of the BQT. We observed (see Chapter 5) that it can be confusing to specifiers to be asked to resolve the Immediacy option, which is concerned with whether or not secondary events are allowed to occur between the primary events, when there are no secondary events in the intended behavior. Assuming that specifiers do indicate that there is at least one secondary event in the behavior (i.e., by selecting the answer on line 38), the BQT then asks the Immediacy child question on line 39, “After A occurs, \{is the event / are the events\} of secondary interest in this behavior allowed to occur before the first subsequent B occurs?” Specifiers can select an answer from either line 40 or line 41 to resolve the Immediacy option.

The “\{ / \}” notation used in this part of the BQT in Figure A.1 is the same as that described in Section 3.1.4.1, and it denotes how the phrasing of these questions and answers changes based on how many secondary events are in the behavior. If there are no secondary events or just one secondary event, then the left side of the “\{ / \}” is displayed. If there is more than one secondary event, then the right side of the “\{ / \}” is displayed. In the case where there are no secondary events, the default name of the parameter C_1 is displayed. If there are n secondary events, where n can be any positive integer, then the specifier-defined names of the secondary events are displayed in alphabetical order in a list of length n. See Section 3.1.2.2 for a detailed example illustration of instantiating the Response scope template by using the BQT.
A.1.1.2 The Absence and Existence Behavior Templates in the BQT

In addition to the Response and Precedence behavior templates, the BQT also provides a representation of the Absence and Existence behavior templates, which are concerned with the restrictions on just one primary event, A. Figure A.2 focuses on lines 2-6 from Figure A.1 and shows how these two behavior templates are represented in the BQT. Specifiers can choose between the Absence and Existence behavior templates by selecting one of the answers under the question, “Which of the following choice best describes the restriction on A?” The Absence behavior template has no options and is represented as the first answer, “A is never allowed to occur.” The Existence behavior template has one option, Bounded, which determines whether A is allowed to occur more than once. The BQT represents the two option settings as the two answers, “A is required to occur at least once” and “A is required to occur exactly once.”

A.1.2 Scope Question Tree

This appendix gives the details of how all four scope templates can be selected and how their options can be resolved using the Scope Question Tree (SQT). Central to this discussion is the fully-expanded version of the SQT, given in Figure A.3. There are two aspects of the how the SQT is portrayed in this figure that differ from its presentation in PROPEL. For the purposes of the discussion in this appendix, each of the questions and answers are assigned a unique line ID, although those IDs are not part of the actual SQT. In addition, the SQT cannot be fully expanded in PROPEL, because when an answer to a question is selected and the child questions under it are revealed, the child questions under the unselected sibling answers are simultaneously hidden. For the purposes of the discussion in this appendix, however, all the SQT questions and their answers are shown simultaneously. For a more accurate example of how specifiers interact with the SQT to specify a property, see Section 3.2.2.1.

A.1.2.1 The Global Scope Template in the SQT

As can be seen in line 1 of Figure A.3, the first question the SQT asks specifiers about the intended property is, “Is the behavior only required to hold within restricted interval(s) in the event sequence?” Lines 2 and 22 show the two possible answers to this question. If specifiers select “Yes, the behavior is only required to hold within restricted interval(s) in the event sequence” on line 2, they will be shown lines 3, 4, 8, and 12, where they have the choice of selecting among the After, Before, and Between scope templates. The following sections discuss those choices in more detail. If specifiers select “No, the behavior is required to hold throughout the entire event sequence” on line 22, they thus choose the Global scope template, which has no options.
Is the behavior only required to hold within restricted interval(s) in the event sequence?

No, the behavior is required to hold throughout the entire event sequence.

Which of the following choices best describes the restricted interval(s)?

There can be at most one restricted interval in the event sequence and it has a starting delimiter, **START**: the behavior is required to hold from an occurrence of **START** through to the end of the event sequence.

What happens if there are multiple occurrences of **START** before the end of the event sequence?

- Only the first occurrence of **START** starts the restricted interval; later occurrences of **START** do not have an effect.
- Only the last occurrence of **START** starts the restricted interval; each occurrence of **START** resets the beginning of the restricted interval.

There can be at most one restricted interval in the event sequence and it has an ending delimiter, **END**: the behavior is required to hold from the start of the event sequence through to the first occurrence of **END**.

If **END** does not occur, is the behavior still required to hold, until the end of the event sequence?

- Yes, if **END** does not occur, the behavior is required to hold throughout the entire event sequence.
- No, if **END** does not occur, the behavior is not required to hold anywhere in the event sequence.

A restricted interval in the event sequence can have both a starting delimiter, **START**, and an ending delimiter, **END**. The behavior is required to hold from an occurrence of **START** through to the end of that restricted interval.

What happens if there are multiple occurrences of **START** without an occurrence of **END** in between them?

- Only the first of those occurrences of **START** potentially starts a restricted interval; later occurrences of **START** within that restricted interval do not have an effect.
- Only the last of those occurrences of **START** potentially starts a restricted interval; each of those occurrences of **START** resets the beginning of that restricted interval.

If an occurrence of **START** is not followed by an occurrence of **END**, is the behavior still required to hold, until the end of the event sequence?

- Yes, if **END** does not occur subsequently, then the behavior is required to hold until the end of the event sequence.
- No, if **END** does not occur subsequently, then the behavior is not required to hold for the remainder of the event sequence.

What happens if **START** occurs after the end of a restricted interval?

- That new occurrence of **START** would potentially start a new restricted interval. The situation would thus be the same as after the first occurrence of **START**, meaning that the restrictions described on any events that take place after that first occurrence would again apply.
- That new occurrence of **START** does not have an effect; no new occurrence of **START** will ever start a new restricted interval.

No, the behavior is required to hold throughout the entire event sequence.

---

**Figure A.3.** Fully-Expanded Scope Question Tree (with line numbers added)
A.1.2.2 The After Scope Template in the SQT

In addition to the Global scope template, the SQT provides a representation of the After scope template. Since the After scope template limits when the behavior is required to hold, focusing only on the part of the event sequence after an occurrence of \textit{START} through to the end of the event sequence, before specifiers can choose the After scope template, they must first select “Yes, the behavior is only required to hold within restricted interval(s) in the event sequence” on line 2. They will then be shown the question on line 3, “Which of the following choices best describes the restricted interval(s)?” and its three possible answers, on lines 4, 8, and 12. To choose the After scope template, specifiers select the answer on line 4. The After scope template has one option, \textit{first/last START}, which is concerned with what should happen if more than one \textit{START} occurs before the end of the event sequence: should only the first of those occurrences of \textit{START} be expected to start a restricted interval, or should only the last of those occurrences be expected to start a restricted interval? This option is represented in the SQT in two places: once in the After scope template and once in the Between scope template, discussed below. For the After scope template, this option is represented by the question on line 5, and its two answers, on lines 6 and 7, represent its two option settings.

A.1.2.3 The Before Scope Template in the SQT

The SQT also provides a representation of the Before scope template. As with the After scope template, the Before scope template limits when the behavior is required to hold. Thus, to choose the Before scope template, specifiers must first select “Yes, the behavior is only required to hold within restricted interval(s) in the event sequence” on line 2. They will then be shown the question on line 3, “Which of the following choices best describes the restricted interval(s)?” and its three possible answers, on lines 4, 8, and 12. To choose the Before scope template, specifiers select the answer on line 8. In the Before scope template, the behavior is required to hold from the start of the event sequence through to the first occurrence of \textit{END}, and this scope template has one option, \textit{missing END}, which is concerned (in the case of the Before scope template) with whether the behavior is required to hold even if \textit{END} never occurs in the event sequence. This option is represented in the SQT in two places: once in the Before scope template and once in the Between scope template, discussed below. For the Before scope template, this option is represented by the question on line 9, and its two answers, on lines 10 and 11, represent its two option settings.

A.1.2.4 The Between Scope Template in the SQT

In addition to the Global, After, and Before scope templates, the SQT also provides a representation of the Between scope template. As with the After and Before scope templates, the Between scope template limits when the behavior is required to hold. Thus, to choose the Between scope template, specifiers must
first “Yes, the behavior is only required to hold within restricted interval(s) in the event sequence” on line 2. They will then be shown the question on line 3, “Which of the following choices best describes the restricted interval(s)?” and its three possible answers, on lines 4, 8, and 12. To choose the Between scope template, specifiers select the answer on line 12. In the Between scope template, the behavior is required to hold from an occurrence of START through to the end of that restricted interval, which, depending on the setting that specifiers select for the missing END option, might be required to be delimited by a subsequent occurrence of END. This scope template has three options, first/last START, missing END, and single/multiple restricted intervals. The first/last START option is concerned with what should happen if more than one START occurs without an occurrence of END in between them: should only the first of those occurrences of START be expected to start a restricted interval, or should only the last of those occurrences be expected to start a restricted interval? This option is represented in the SQT in two places: once in the After scope template and once in the Between scope template. For the Between scope template, this option is represented by the question on line 13, and its two answers, on lines 14 and 15, represent its two option settings. The missing END option is concerned (in the case of the Between scope template) with whether the behavior is still required to hold even if an occurrence of START is not followed by an occurrence of END. This option is represented in the SQT in two places: once in the Before scope template and once in the Between scope template. For the Between scope template, this option is represented by the question on line 16, and its two answers, on lines 17 and 18, represent its two option settings. The single/multiple restricted intervals option is concerned with what happens after an occurrence of END ends a restricted interval. Does a subsequent occurrence of START begin a new restricted interval? This option is represented in the SQT by the question on line 19, and its two answers, on lines 20 and 21, represent its two option settings. See Section 3.2.2.1 for a detailed example illustration of instantiating the Between scope template in the SQT.

A.2 Finite-State Automata Templates

The FSA template property view notation that is used in this appendix is defined in Section 3.1.3.1. Like the other property views, the FSA templates can help specifiers to resolve the options that are associated with the selected property pattern template. Thus, the same options that must be resolved in the QT and DNL template property views, and in the ST and SST scope representations, are options in the FSA template property view as well.

A.2.1 Behavior FSA Templates

This appendix gives the details of how all four behavior templates are represented in the FSA template property view, and how all of the behavior templates’ options can be resolved using the behavior FSA tem-
plates. Central to this discussion are Figures A.4, A.5, A.6, and A.7, which each show one of the behavior FSA templates. There is one aspect of the how the behavior FSA templates are portrayed in these figures that differs from their presentation in PROPEL. For the purposes of the discussion in this appendix, each of the states in the FSA templates are assigned a unique ID, although those IDs are not part of the actual behavior FSA templates. For a more realistic example of how specifiers interact with a behavior FSA template to specify a property, see Section 3.1.3.2. The following sections discuss each of the behavior FSA templates in more detail.

A.2.1.1 The Response Behavior FSA Template

As was noted in Section 3.1.1, the Response behavior states that if some event $A$ occurs, it must be followed by some other event, $B$. We represent the Response behavior template with the Response FSA template, which is shown in Figure A.4. This figure shows the Response FSA template with all of its optional components displayed. The seven options in the Response behavior template are given in Table 3.2, and each of the options is represented in the Response FSA template. The Nullity option is determined by whether or not state 1 is an accepting state. If state 1’s inner circle exists, then the state is accepting and $A$ is not required to occur at all. If state 1’s inner circle does not exist, then the state is not accepting and $A$ is required to occur at least once. The Precedency option is determined by the multi-label item that is on state 1’s self-loop. If the item chosen is “$\neg(A,B)$”, then $B$ is not allowed occur before the first occurrence of $A$. If the item chosen is “$\neg A$”, then $B$ is allowed to occur before the first occurrence of $A$. The Pre-arity option is determined by the multi-label item chosen for the optional self-loop on state 2. If that self-loop does not exist because all the items in its associated multi-label have been removed from consideration, or if the item chosen on it is “$\neg(A,B)$”, then $A$ is only allowed to occur once before an occurrence of $B$. Any other item chosen for that multi-label will allow $A$ to occur multiple times before an occurrence of $B$. 
Since the property’s alphabet may include secondary events in addition to just the specifier-defined events that are associated with \(A\) and \(B\), the *Immediacy* option deals with whether or not these secondary events may occur at state 2. *Immediacy* is determined by the multi-label item that is chosen for the optional self-loop on state 2. If that self-loop does not exist because all the items in its associated multi-label have been removed from consideration, or if the item chosen for the self-loop is “\(A\)”, then the secondary events are not allowed to occur between an occurrence of \(A\) and an occurrence of \(B\). If the multi-label item that is chosen for the self-loop is “\(-(A,B)\)” or “\(-B\)”, then the secondary events are allowed to occur between an occurrence of \(A\) and an occurrence of \(B\).

The *Post-arity* option is determined by the multi-label item that is chosen for state 3’s self-loop. If the multi-label item chosen is “\(-(A,B)\)” or “\(-B\)”, then \(B\) is only allowed to only occur once after an occurrence of \(A\). The other two multi-label items allow \(B\) to occur multiple times after an occurrence of \(A\). The *Finalization* option is determined not only by the multi-label item that is chosen for state 3’s self-loop, but also by the existence of the transition on “\(A\)” from state 3 to state 2. If that transition exists or if the multi-label item chosen for state 3’s self-loop is “\(-B\)” or “\(\cdot\)”, then after an occurrence of \(B\) follows an occurrence of \(A\), \(A\) is allowed to occur again. If the transition on “\(A\)” from state 3 to state 2 does not exist and if the multi-label item for state 3’s self-loop is “\(-(A,B)\)” or “\(-A\)”, then \(A\) is not allowed to occur again. The *Repeatability* option is determined by the existence of the transition on “\(A\)” from state 3 to state 2. If that transition exists, then after an occurrence of \(B\) follows an occurrence of \(A\), further occurrences of \(A\) are again required to be followed by a further occurrence of \(B\), and the only multi-label items that are allowed to be chosen for state 3’s self-loop are “\(-(A,B)\)” and “\(-A\)”. If that transition does not exist, then further occurrences of \(A\) do not impose additional restrictions on the occurrences of any future events, and any of the four multi-label items on state 3’s self-loop can be chosen.

In addition to the relationships between the underlying Response behavior template option settings and the various parts of the Response FSA template, there are also some subtle constraints on the legal combinations of the *Finalization* and *Repeatability* options. The FSA template property view is the only one that makes the possibility of illegal combinations of those two options’ settings visible to specifiers, so although the constraints exist for all of the property views because they exist in the underlying Response behavior template, they are only explicitly enforced in the FSA template property view. In the QT and DNL template property views, the illegal combinations are simply not shown to specifiers: specifiers can only select from the set of legal combinations. In the scope-only property views, ST and SST, there are no constraints on the possible combinations of the scope template options. We thus only describe the constraints in this FSA template discussion, and we discuss the implications for how specifiers are constrained in their interaction with the Response FSA template property view.
### Table A.1. Options that Affect State 3’s Transitions in the Response and Precedence FSA Templates

<table>
<thead>
<tr>
<th>Description</th>
<th>Possible settings</th>
<th>Shorthand</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Repeatability</strong></td>
<td>After A and B occur, this option determines whether, if A occurs subsequently, the situation is the same as when the first A occurred, meaning that the restrictions described on any events that take place after that occurrence would again apply.</td>
<td>If another A does occur, the situation is the same as when the first A occurred, meaning that the restrictions described on any events that take place after that occurrence would again apply: it is allowed to repeat.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>A is allowed to occur again, zero or more times, and there are no restrictions on the occurrences of any future events: it is not allowed repeat.</td>
</tr>
<tr>
<td></td>
<td>Unresolved</td>
<td>Unresolved</td>
</tr>
<tr>
<td><strong>Finalization</strong></td>
<td>After A and B occur, this option determines whether A is allowed to occur again.</td>
<td>Yes, A is allowed to occur again.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>No, A is not allowed to occur again.</td>
</tr>
<tr>
<td></td>
<td>Unresolved</td>
<td>Unresolved</td>
</tr>
<tr>
<td><strong>Pre-Finalization</strong></td>
<td>When both Repeatability and Finalization are unresolved and specifiers select the ¬(A, B) or ¬A multi-label items on state 3’s self-loop, neither option can be resolved. This demi-option stores information necessary to resolve them later.</td>
<td>Both Repeatability and Finalization are currently unresolved, but if Repeatability is later set to “not repeatable”, then Finalization should be set to “A not allowed”.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>No relevant information is stored; ignore this demi-option.</td>
</tr>
<tr>
<td><strong>Post-arity</strong></td>
<td>After A and B occur, this option determines whether B is allowed to occur again.</td>
<td>Yes, B is allowed to occur again.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>No, B is not allowed to occur again.</td>
</tr>
<tr>
<td></td>
<td>Unresolved</td>
<td>Unresolved</td>
</tr>
</tbody>
</table>

The two options whose possible combinations are constrained are **Finalization** and **Repeatability**. Table A.1 gives both options, the space of option settings available for each of them, and our shorthand notation for referring to those settings. Table A.1 also gives a “demi-option”, **Pre-Finalization**, which only stores information about potential settings for the **Finalization** option, given how specifiers have made decisions concerning certain parts of the Response FSA template. The shorthand notation defined for the **Post-arity** option is discussed later in this section.

Using the shorthand notation defined in Table A.1, the constraints on the combinations of option settings are the following:

1. **Repeatability** = A → **Finalization** = A
2. **Finalization** = N → **Repeatability** = N
3. **Finalization** ≠ U → **Pre-Finalization** = I
4. \( \text{Repeatability} = N \land \text{Pre-Finalization} = N \rightarrow \text{Finalization} = N \)

Constraints 1 and 2 are included because they are required by the definitions of the \textit{Repeatability} and \textit{Finalization} options. Constraint 3 is included because once \textit{Finalization} has been decided, there is no longer a need to store information in \textit{Pre-Finalization}, and we thus want to avoid constraint 4 being applied incorrectly later.

Constraint 4 is necessary to handle two undesirable situations. First, if the \( \neg(A,B) \) multi-label item on state 3’s self-loop is selected when neither \textit{Repeatability} nor \textit{Finalization} has been resolved, the selection does not resolve either of those options. Selecting \( \neg(A,B) \) in this situation only changes the setting of the \textit{Post-arity} option. It is not possible to make the \( \neg(A,B) \) multi-label item be the only visible one on state 3’s self-loop in this situation, because it is not the only multi-label item that is compatible with those option settings. When \textit{Post-arity} is set to ‘B not allowed’ and the other two options are unresolved, this means that there are two multi-label items still available on state 3’s self-loop: \( \neg(A,B) \) and \( \neg B \). Leaving both multi-label items visible is unacceptable, however, since if specifiers have selected the \( \neg(A,B) \) multi-label item, they expect that item to be the only multi-label item visible on state 3’s self-loop. The second reason why constraint 4 is necessary is related to both the \( \neg(A,B) \) and \( \neg A \) multi-label items on state 3’s self-loop. Selecting either of them when neither \textit{Repeatability} nor \textit{Finalization} has been resolved does nothing to resolve those two options, but it does eliminate one possibility: \( A \) is not allowed to occur on state 3’s self-loop.

Although this selection does change the space of possibilities available in the FSA template, it is not reflected in any of the related options, \textit{Post-arity}, \textit{Repeatability}, or \textit{Finalization}. Preserving this new status requires the use of the \textit{Pre-Finalization} demi-option, in this case setting it to ‘N’. If either \( \neg(A,B) \) or \( \neg A \) is the selected multi-item on state 3’s self-loop when the \textit{Repeatability} option is resolved, the correct setting of the \textit{Finalization} option can be identified, because the fully-resolved FSA must be deterministic. If \( A \) is not allowed to occur on state 3’s self-loop, then either its occurrence causes the FSA to take the transition from state 3 to state 2 (i.e., when \textit{Repeatability} is set to ‘A’ and \textit{Finalization} is set to ‘A’) or its occurrence causes the FSA to take the transition from state 3 to the non-accepting trap state (i.e., when \textit{Repeatability} is set to ‘N’ and \textit{Finalization} is set to ‘N’).

What these constraints imply for how specifiers interact with the Response FSA template is given in Table A.2. There are 54 possible combinations of the \textit{Post-arity} (P), \textit{Repeatability} (R), \textit{Finalization} (F), and \textit{Pre-Finalization} (pF) option settings, but there are only 23 legal combinations, according to the four constraints given previously. There are 5 legal combinations that specifiers cannot reach through the property
Table A.2. Response and Precedence FSA Template Interaction Constraints

<table>
<thead>
<tr>
<th>Response or Precedence FSA template component changed</th>
<th>Original</th>
<th>New</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>P R F pF</td>
<td>P R F pF</td>
</tr>
<tr>
<td>state 3 → state 2 transition on A does not exist</td>
<td>U U U I</td>
<td>U N U I</td>
</tr>
<tr>
<td></td>
<td>U U A I</td>
<td>U N A I</td>
</tr>
<tr>
<td></td>
<td>U N U I</td>
<td>U N U I</td>
</tr>
<tr>
<td></td>
<td>U N A I</td>
<td>U N A I</td>
</tr>
<tr>
<td></td>
<td>U N N I</td>
<td>U N N I</td>
</tr>
<tr>
<td></td>
<td>A U U I</td>
<td>A N U I</td>
</tr>
<tr>
<td></td>
<td>A A A I</td>
<td>A N A I</td>
</tr>
<tr>
<td></td>
<td>A N U I</td>
<td>A N U I</td>
</tr>
<tr>
<td></td>
<td>A N A I</td>
<td>A N A I</td>
</tr>
<tr>
<td></td>
<td>A N N I</td>
<td>A N N I</td>
</tr>
<tr>
<td></td>
<td>N U U I</td>
<td>N U U I</td>
</tr>
<tr>
<td></td>
<td>N U A I</td>
<td>N U A I</td>
</tr>
<tr>
<td></td>
<td>N N U I</td>
<td>N N U I</td>
</tr>
<tr>
<td>All labels on state 3’s self-loop are enabled</td>
<td>? ? ? ?</td>
<td>U U U I</td>
</tr>
<tr>
<td>“¬A” multi-label item on state 3’s self-loop is selected</td>
<td>U U U I</td>
<td>A U U N</td>
</tr>
<tr>
<td></td>
<td>U A A I</td>
<td>A A A I</td>
</tr>
<tr>
<td></td>
<td>U N U I</td>
<td>A N N I</td>
</tr>
<tr>
<td></td>
<td>U N A I</td>
<td>A N N I</td>
</tr>
<tr>
<td></td>
<td>U N N I</td>
<td>A N N I</td>
</tr>
<tr>
<td></td>
<td>A U U I</td>
<td>A U U N</td>
</tr>
<tr>
<td></td>
<td>A A A I</td>
<td>A A A I</td>
</tr>
<tr>
<td></td>
<td>A N U I</td>
<td>A N N I</td>
</tr>
<tr>
<td></td>
<td>A N A I</td>
<td>A N N I</td>
</tr>
<tr>
<td></td>
<td>A N N I</td>
<td>A N N I</td>
</tr>
<tr>
<td></td>
<td>N U U I</td>
<td>A U U N</td>
</tr>
<tr>
<td></td>
<td>N U A I</td>
<td>A A A I</td>
</tr>
<tr>
<td></td>
<td>N A A I</td>
<td>A A A I</td>
</tr>
<tr>
<td></td>
<td>N N U I</td>
<td>A N N I</td>
</tr>
<tr>
<td></td>
<td>N N N I</td>
<td>A N N I</td>
</tr>
<tr>
<td></td>
<td>A U U N</td>
<td>A U U N</td>
</tr>
<tr>
<td></td>
<td>N U U N</td>
<td>A U U N</td>
</tr>
<tr>
<td>“¬B” multi-label item on state 3’s self-loop is selected</td>
<td>? ? ? ?</td>
<td>N N A I</td>
</tr>
<tr>
<td>“¬(A B)” multi-label item on state 3’s self-loop is selected</td>
<td>U U U I</td>
<td>N U U N</td>
</tr>
<tr>
<td></td>
<td>U A A I</td>
<td>N A A I</td>
</tr>
<tr>
<td></td>
<td>U N U I</td>
<td>N N N I</td>
</tr>
<tr>
<td></td>
<td>U N A I</td>
<td>N N N I</td>
</tr>
<tr>
<td></td>
<td>U N N I</td>
<td>N N N I</td>
</tr>
<tr>
<td></td>
<td>A U U I</td>
<td>N U U N</td>
</tr>
<tr>
<td></td>
<td>A A A I</td>
<td>N A A I</td>
</tr>
<tr>
<td></td>
<td>A N U I</td>
<td>N N N I</td>
</tr>
<tr>
<td></td>
<td>A N A I</td>
<td>N N N I</td>
</tr>
<tr>
<td></td>
<td>A N N I</td>
<td>N N N I</td>
</tr>
<tr>
<td></td>
<td>N U U I</td>
<td>N U U N</td>
</tr>
<tr>
<td></td>
<td>N U A I</td>
<td>N A A I</td>
</tr>
<tr>
<td></td>
<td>N N U I</td>
<td>N N A I</td>
</tr>
<tr>
<td></td>
<td>N N N I</td>
<td>N N N I</td>
</tr>
<tr>
<td>“*,” multi-label item on state 3’s self-loop is selected</td>
<td>? ? ? ?</td>
<td>A N A I</td>
</tr>
</tbody>
</table>
views\(^1\), so only the 18 reachable combinations are shown in Table A.2. When "?" is used in an option-setting column in Table A.2, it means that the option can be set to any of its possible values. Thus, "????" is a shorthand notation used here to represent all the reachable combinations of the options’ settings. When "?" is used in the option-setting columns on the right side of Table A.2, it indicates that whichever option settings would be in the columns on the left side are just copied over to the right side, for each reachable combination.

In addition to the four constraints on the option settings given above, there are two other principles that guide how we handle the specifiers’ changes to the Response FSA template. One rule that we adopted is that a resolved option trumps an unresolved option. If the specifiers’ selection takes the FSA template from a combination of option settings where either Repeatability or Finalization is resolved, but the other is not, the resolved option keeps its setting, and the unresolved option changes according to constraints 1 and 2, given the resolved option’s setting. This rule does not handle the situation where both Repeatability and Finalization are currently resolved and their settings are not compatible with the specifiers’ selection in the FSA template, however. Based on the aim of minimizing confusion to specifiers, we chose to keep changes visually localized to just the parts of the Response FSA template that specifiers directly select. For example, if specifiers select a multi-label item on state 3’s self-loop and both Repeatability and Finalization were resolved at the time, only Finalization changes, because changing the status of the state 3 to state 2 transition on A (i.e., changing the Repeatability setting) is visually farther away from specifiers’ direct selection than changing just the multi-label items on state 3’s self-loop (i.e., changing the Finalization setting).

---

\(^1\)Those unreachable combinations are unnecessary, because there are alternative ways to reach all the legal combinations where all four options are resolved.
A.2.1.2 The Precedence Behavior FSA Template

As was noted in Section 3.1.1, the Precedence behavior states that some event $A$ must occur before some other event $B$ is allowed to occur. The Precedence behavior template is similar to the Response behavior template, but it differs in one significant way: the Precedence behavior template has only six\(^2\) of the seven options that are in the Response behavior template. We represent the Precedence behavior template with the Precedence FSA template, which is shown in Figure A.5. This figure shows the Precedence FSA template with all of its optional components displayed. Since the Precedence FSA template is similar to the Response FSA template in most respects, including the constraints on the legal combinations of the Repeatability and Finalization options, in this discussion we focus solely on the part of the Precedence FSA template that differs from the Response FSA template. State 1’s self-loop in the Precedence FSA template has only one label, “$\neg(A,B)$.” There is no choice on this self-loop because the Precedence behavior template has no Precedency option: $B$ is simply not allowed to occur before the first occurrence of $A$.

A.2.1.3 The Absence Behavior FSA Template

As was noted in Section 3.1.1, the Absence behavior states that some event $A$ is not allowed to occur in the sequence of events. The Absence FSA template, which is shown in Figure A.6, does not have any options and thus it does not have any optional components.

\(^2\)The Precedence behavior template also slightly redefines the meaning of the Repeatability option in terms of the core concept of the Precedence behavior. In the Precedence behavior template, Repeatability concerns whether or not, after an occurrence of $B$ follows an occurrence of $A$, further occurrences of $A$ enable further occurrences of $B$. 

---

Figure A.6. Absence FSA Template

Figure A.7. Existence FSA Template
A.2.1.4 The Existence Behavior FSA Template

As was noted in Section 3.1.1, the Existence behavior states that some event A is required to occur at least once in the sequence of events. This behavior template has one option, Bounded, which determines how many times A is allowed to occur: exactly once or multiple times. In the Existence FSA template, which is shown in Figure A.7, this option is determined by which multi-label item is chosen on state 2’s self-loop. If the item chosen is “¬A”, then A is allowed to occur exactly once. If the item chosen is “.”, then A is allowed to occur one or more times.

A.2.1.5 Proposal for a Generalized 2-Event Behavior FSA Template

One common issue that PROPEL specifiers have struggled with is how to create a property specification that draws on both the Response and Precedence behavior templates. Although it is possible to do this to a certain extent using the current Response behavior template, not all combinations of these two concepts are available for specifiers to experiment with. There are also other two-event behaviors that have been observed in practice and should thus be considered, such as two-event Alternation. These other two-event behaviors are only partially covered by the current Response and Precedence behavior templates. This begs the question of whether a more generalized two-event behavior template should be considered, one that covers all the two-event behaviors that specifiers can currently express in PROPEL and that introduces some new possibilities. Figure A.8 shows a generalized two-event behavior template that we briefly explored using the FSA template property view. In addition to the 96 behaviors that correspond to those supported by the current Response behavior template and the 48 behaviors that correspond to those supported by the current Precedence behavior template, there are 46 other behaviors made possible by this generalized two-event behavior template. 16 of these behaviors might qualify as a type of two-event Alternation, and they all require that the sequence of events must be started by only one of the parameters (i.e., A). The 30 remaining behaviors each focus on a different combination of option settings for a subset of the options that are available in the current Response and Precedence behavior templates, but do not correspond to either of those unifying concepts.

The 16 behaviors that might qualify as an “A-prefix” type of Alternation can be separated into the following categories, which are described in terms of the Generalized 2-Event FSA template:

- 4 FSAs where, if A occurs, A is not allowed to occur again until after an intervening B occurs, and there are no restrictions on the occurrence of B. To create these, make the multi-label item on state 1’s self-loop be “¬A”, make state 2 accepting, don’t allow A to occur on state 2’s self-loop, make the multi-label item on state 3’s self-loop be “¬A”, and make the transition on A from state 3 to state 2 exist. Nullity and Immediacy are unrestricted, so state 1 can either be accepting or not accepting, and state 2’s self-loop can either not exist or can have the “¬(A,B)” multi-label item on it.
Figure A.8. Generalized 2-Event FSA Template

- 4 FSAs where, if \( A \) and then a subsequent \( B \) occurs, \( B \) is not allowed to occur again until after an intervening \( A \) occurs, and there are no restrictions on the occurrence of \( A \). To create these, make the multi-label item on state 1’s self-loop be “\( \neg A \)”, make state 2 accepting, allow \( A \) to occur on state 2’s self-loop, make the multi-label item on state 3’s self-loop be “\( \neg(A,B) \)”, and make the transition on \( A \) from state 3 to state 2 exist. Nullity and Immediacy are unrestricted, so state 1 can either be accepting or not accepting, and state 2’s self-loop can either have the “\( A \)” or “\( \neg B \)” multi-label items on it.

- 8 FSAs where, after \( A \) occurs, \( A \) and \( B \) must strictly alternate. That is, if \( A \) occurs, \( A \) is not allowed to occur again until after an intervening \( B \) occurs, and vice versa. To create these, make the multi-label item on state 1’s self-loop be “\( \neg A \)”, make state 2 accepting, don’t allow \( A \) to occur on state 2’s self-loop, and make the multi-label item on state 3’s self-loop be “\( \neg(A,B) \)”. Nullity, Immediacy, and Repeatability are unrestricted, so state 1 can either be accepting or not accepting, state 2’s self-loop can either not exist or can have the “\( \neg(A,B) \)” multi-label item on it, and the transition on \( A \) from state 3 to state 2 might or might not exist.

The 30 remaining behaviors that each focus on a different combination of option settings can be separated into the following categories, which are described in terms of the Generalized 2-Event FSA template:

- 2 FSAs make only Nullity restrictions: they require \( A \) to occur, but they restrict nothing else. To create these, make state 1 not accepting, make state 2 accepting, and do not make any outgoing transitions from state 2 or state 3 go to the non-accepting trap state.

- 2 FSAs make only Immediacy restrictions: they prevent secondary events from occurring between an \( A \) and the first subsequent occurrence of \( B \), but they restrict nothing else. To create these, make state 2 accepting, make the multi-label item on state 2’s self-loop be “\( A \)”, and do not make any outgoing transitions from state 1 or state 3 go to the non-accepting trap state.
• 3 FSAs assume that the behavior is not repeatable and they make only Post-arity or Finalization restrictions: after A and a subsequent B have occurred, they either prohibit A, B, or both from ever occurring again. To create these, make state 2 accepting, make the multi-label item on state 3’s self-loop be either “¬A”, “¬B”, or “¬(A,B)”, respectively, and do not make any outgoing transitions from state 1 or state 2 go to the non-accepting trap state.

• 11 FSAs enforce two or three of the above sets of restrictions, but restrict nothing else. To create these, follow the instructions given for each of those sets of restrictions.

• 12 FSAs make the remaining legal combinations (given the constraints defined in Appendix A.2.1.1) of all the options in the Response behavior template. These FSAs assume that B is allowed to occur on state 1’s self-loop, that A is not allowed to occur at state 2, and that the behavior is not repeatable. To create these, iterate through all the remaining combinations of varying the accepting status of state 1, varying whether secondary events are allowed to occur on state 2’s self-loop, and varying whether “¬A”, “¬B”, or “.” is the multi-label item on state 3’s self-loop. Do not make any outgoing transitions from state 1 go to the non-accepting trap state.

A.2.2 Scope FSA Templates

The following sections present the algorithm for applying each scope FSA template to any of the behavior FSA templates discussed above. Assuming that the behavior alphabet and the scope alphabet are disjoint and that the parameters in the behavior templates and in the scope templates are associated only with single events or with disjunctions of events, the scope application algorithm given below will apply to any FSA templates developed for the new behaviors mentioned in Chapter 7.

A scope FSA template can be applied to an behavior FSA template by adding additional states called “scope states” and by adding transitions between the scope states and the states of the behavior FSA template. These transitions are labeled by the scope delimiters, START and END. We add these scope delimiters to the alphabet of the property, Σ. A scope state that has a self-loop for every event in the alphabet is called a trap state, and there is at most one accepting trap state and one non-accepting trap state in an FSA. Figures A.9, A.10, and A.11 show the After, Before, and Between scope templates applied to the Existence FSA template, respectively. The Global scope template is not shown since it makes no changes to the behavior FSA template. For the sake of brevity, instead of separately showing the self-loops that are added for occurrences of the scope delimiters, we have changed the multi-label items from the original behavior FSA template’s self-loops to reflect the scope-delimiter self-loops. We have also not shown the transitions that go to a non-accepting trap state; when a transition is not provided that explicitly allows a scope delimiter to occur, it should be assumed
that an occurrence of that scope delimiter puts the FSA into a non-accepting trap state. In the remainder of
this section, we explain what is needed to apply each scope FSA template to the Existence FSA template.
Scopes are added to the other behavior FSA templates in the same way.

A.2.2.1 The Global Scope FSA Template

As was noted in Section 3.2.1, the Global scope template states that the behavior is required to hold from
the start of the event sequence until the end of the event sequence. The Global FSA template is the only
one that has no optional FSA template components for specifiers to select and it does not use either scope
delimiter. Applying the Global FSA template makes no changes to the behavior FSA templates discussed in
section A.2.1. The Global FSA template is the default for a new property specification.

A.2.2.2 The After Scope FSA Template

As was noted in Section 3.2.1, the After scope template states that the behavior must hold after an occur-
rence of START until the end of the event sequence. When we apply the After FSA template to the behaviors’
FSA templates (see Figure A.9 for an example of applying it to the Existence FSA template), we must deter-
mine at each state what the effect will be of encountering START at that point in the event sequence. State 1
in Figure A.9 is a scope state that is added to the Existence FSA template. An occurrence of START at this
state would potentially begin the restricted interval within which the Existence behavior would be required
to hold. As is shown with the self-loop that is labeled “¬START” on state 1, all events that occur before the
first occurrence of START are ignored: they occur on state 1’s self-loop.

The After scope template has one option, first/last START, which determines what happens if START
occurs more than once before the end of the event sequence. There are two alternative settings associated
with this option. One setting is that the first occurrence of START begins a restricted interval and later
occurrences of START have no effect. For this setting, all occurrences of START after the first one must
self-loop on each of the states that were also in the original behavior FSA template. In Figure A.9, this is

Figure A.9. The After Scope Template Applied to the Existence FSA Template
states 2, 3, and 4. The other possible setting is that only the last occurrence of \textit{START} begins the restricted interval, and thus each occurrence of \textit{START} resets the beginning of the restricted interval. For this setting, all occurrences of \textit{START} must be on transitions that go from the original behavior FSA template states back to the original behavior FSA template start state (i.e., state 2 in Figure A.9). It should be noted that since state 2 was the original behavior FSA template start state, both option settings are represented the same way in Figure A.9. This is true of the original start states in all the behavior FSA templates.

To represent these two option settings, we add the ability for specifiers to decide whether \textit{START} occurs on self-loops for each original behavior FSA template state, or whether \textit{START} occurs on transitions that go back to the original behavior FSA template start state. Instead of separately showing the self-loops that are added for occurrences of \textit{START}, we have changed the multi-label items from the original behavior FSA template’s self-loops to reflect the \textit{START} self-loops. It is not necessary to change the “\textit{¬A}” or “.” multi-label items, because \textit{START} is in the sets of events that those multi-label items represent. It is necessary to add two multi-label items to state 3’s self-loop, however, because state 3 has an optional outgoing transition to state 2 on \textit{START}, and since the fully-resolved FSA must be deterministic, \textit{START} cannot occur on both state 3’s self-loop and also on the transition from state 3 to state 2. If specifiers decide that the state 3 to state 2 transition exists, then either the “\textit{¬(A,START)}” multi-label item or the “\textit{¬START}” multi-label item must be the one on state 3’s self-loop. Which one is selected depends on the setting of the \textit{Bounded} option in the Existence behavior template. Similarly, the transition from state 4 to state 2 on \textit{START} would also exist because it is based on the same option setting as the transition from state 3 to state 2. For similar reasons of determinism, the “\textit{¬START}” multi-label item must thus be the one on state 4’s self-loop. If specifiers instead decide that the state 3 to state 2 transition does not exist, then either the “\textit{¬A}” multi-label item or the “.” multi-label item must be the one on state 3’s self-loop. In addition, the transition from state 4 to state 2 on \textit{START} would also not exist, and the “.” multi-label item must thus be the one on state 4’s self-loop.

\textbf{A.2.2.3 The Before Scope FSA Template}

As was noted in Section 3.2.1, the Before scope template states that the behavior must hold from the start of any event sequence until the first occurrence of \textit{END} in that sequence. When we apply the Before FSA template to the behaviors’ FSA templates (see Figure A.10 for an example of applying it to the Existence FSA template), we must determine at each state what the effect will be of encountering \textit{END} at that point in the event sequence. If \textit{END} occurs in state 1, this results in a restricted interval that has ended before any of the behavior events (in the case of the Existence behavior template, \textit{A}) have occurred. We refer to

\footnote{State 4 was the original behavior FSA template’s non-accepting trap state.}
Figure A.10. The Before Scope Template Applied to the Existence FSA Template

this as an “empty restricted interval,” and it is handled differently depending on the behavior template in use. For instance, the Absence behavior holds if the restricted interval is empty, since no occurrences of \( A \) in the restricted interval satisfies that behavior. For all the other behaviors we have described, however, an empty restricted interval constitutes a violation of the property. For example, take the Existence FSA template shown in Figure A.10. Recall that the Existence behavior template is concerned with expressing the concept that \( A \) must occur at least once in the restricted interval. In this case, the occurrence of \( \text{END} \) in state 1 puts the FSA template into a non-accepting trap state, because the restricted interval is ended before the Existence behavior is satisfied. As noted previously, we do not show the transitions that go to a non-accepting trap state. When the FSA is in state 2, \( A \) has occurred and the Existence behavior is satisfied. Thus, if \( \text{END} \) occurs when the FSA template is in state 2, the FSA template will take the transition to state 3, an accepting trap state, because the property holds and cannot be violated, since the Before scope is not repeatable. State 3 is a scope state that is added to the original set of states in the Existence FSA template. State 4 is the original behavior FSA template non-accepting trap state.

The Before scope template has one option, \textit{missing END}, which determines what happens if \( \text{END} \) never occurs in the event sequence. This option is represented in the FSA template property view by the accepting status of each of the behavior FSA template states whose original status in the behavior FSA template was not-accepting or optionally-accepting. There are two alternative settings associated with this option. One setting is that if \( \text{END} \) does not occur, then the behavior is required to hold throughout the entire event sequence. For this setting, the accepting status of the original behavior FSA template states is unchanged, because the behavior must hold throughout the entire event sequence and if it is violated, then the property is violated. Since states 1 and 4 in Figure A.10 were not accepting in the original Existence FSA template, this option setting is represented by these states being set to not-accepting. The other possible option setting is that if \( \text{END} \) does not occur, then the behavior is not required to hold anywhere in the event sequence. For this
setting, all of the original behavior FSA template states must be accepting, because the property cannot be violated unless END occurs before the behavior holds. In the case of the FSA template shown in Figure A.10, the property can only be violated if END occurs when the FSA is in state 1 or state 4. Since these two states in Figure A.10 were not accepting in the original Existence FSA template, this option setting is represented by these states being set to accepting. It should be noted that because state 2 was an accepting state in the original Existence FSA template, its accepting status is unchanged when the Before FSA template is applied.

Applying the Before scope template to the Existence FSA template also affects the multi-label items. Because the set-complement operator ("¬") means that everything that is not in the set specified by the multi-label item is accepted, END needs to be added to each of those multi-label items in which the set-complement operator is used. END is not permitted to occur on state 1’s self-loop, and thus it is added to the original “¬A” multi-label item that was on that self-loop in the Existence behavior FSA template, and the new multi-label item therefore becomes “¬(A,END)”. The same transformation happens on state 2’s self-loop. Note that the second multi-label item in state 2’s self-loop is also changed from the whole set of events (“.”) to the set of events excluding END (“¬END”). This is done to preserve determinism in the fully-resolved FSA, since there is always an outgoing transition from state 2 to state 3 on END. Similarly, the “.” multi-label item on state 4’s self-loop becomes “¬END”, because there is always an outgoing transition from state 4 to the non-accepting trap state on END.

A.2.2.4 The Between Scope FSA Template

As was noted in Section 3.2.1, the Between template states that the behavior must hold after an occurrence of START until the first subsequent occurrence of END. When we apply the Between FSA template to the behaviors’ FSA templates (see Figure A.11 for an example of applying it to the Existence FSA template), we must determine at each state what the effect will be of encountering either scope delimiter at that point in the event sequence. States 1 and 4 in Figure A.11 are scope states that are added to the Existence FSA template. An occurrence of START at state 1 would potentially begin a restricted interval within which the Existence behavior would be required to hold. All other events that occur before the first occurrence of START are ignored, including END: they occur on state 1’s self-loop. State 4 is discussed in more detail later in this section.

If we examine what happens when the scope delimiters occur in state 2, we can first consider the option that is represented by state 2’s accepting status. This option is missing END, which determines what should happen if, after a restricted interval is begun, END never occurs to end it. As was described in Appendix A.2.2.3, this option is represented in the FSA template property view by the accepting status of each of the behavior FSA template states whose original status in the behavior FSA template was not-accepting.
Figure A.11. The Between Scope Template Applied to the Existence FSA Template

or optionally-accepting: states 2 and 5. There are two alternative settings associated with this option. One setting is that if END does not occur to end a restricted interval, then the behavior is required to hold through to the end of the event sequence. For this setting, the accepting status of the original behavior FSA template states is unchanged, because the behavior must hold within that restricted interval and if the behavior is violated, then the property is violated. Since states 2 and 5 in Figure A.11 were not accepting in the original Existence FSA template, this option setting is represented by these states being set to not-accepting. The other possible option setting is that if END does not occur, then the behavior is not required to hold for the remainder of the event sequence. For this setting, all of the original behavior FSA template states must be accepting, because the property cannot be violated unless END occurs before the behavior holds. In the case of the FSA template shown in Figure A.11, the property can only be violated if END occurs when the FSA is in states 2 or 5. Since these in Figure A.11 were not accepting in the original Existence FSA template, this option setting is represented by these states being set to accepting. It should be noted that state 3 was an accepting state in the original Existence FSA template, so its accepting status is unchanged when the Between FSA template is applied.

In addition to the optionally-accepting status of state 2, another aspect of how the scope delimiters are handled in that state should be considered. Since START always self-loops on the state that was the original behavior FSA template’s start state (see Appendix A.2.2.2 for an explanation of why), START self-loops on state 2. If END occurs in state 2, however, this results in an empty restricted interval, as described in Appendix A.2.2.3. Recall that the Existence behavior template is concerned with expressing the concept that A must occur at least once in each restricted interval. In this case, the occurrence of END in state 2 puts the...
FSA template into a non-accepting trap state, because this restricted interval is ended before the Existence behavior is satisfied. As noted previously, we do not show the transitions that go to a non-accepting trap state.

If we examine what happens when the scope delimiters occur in state 3, we can first consider the *first/last START* option, which determines what happens if *START* occurs more than once before the end of the given restricted interval. The way that the FSA template property view represents this option is similarly described in Appendix A.2.2.2. There are two alternative settings associated with this option. One setting is that the first occurrence of *START* begins a restricted interval and later occurrences of *START* have no effect. For this setting, all occurrences of *START* after the first one must self-loop on each of the states that were also in the original behavior FSA template. In Figure A.11, this is states 2, 3, and 5. The other possible setting is that only the last occurrence of *START* before the end of a restricted interval begins this restricted interval. That is, each occurrence of *START* resets the beginning of this restricted interval. For this setting, all occurrences of *START* must be on transitions that go from the original behavior FSA template states back to the original behavior FSA template start state (i.e., state 2 in Figure A.11).

To represent these two option settings, we add the ability for specifiers to decide whether *START* occurs on self-loops for each original behavior FSA template state, or whether *START* occurs on transitions that go back to the original behavior FSA template start state. Instead of separately showing the self-loops that are added for occurrences of *START*, we have changed the multi-label items from the original behavior FSA template’s self-loops to reflect the *START* self-loops. It is necessary to add two multi-label items to state 3’s self-loop, however, because state 3 has an optional outgoing transition to state 2 on *START*, and since the fully-resolved FSA must be deterministic, *START* cannot occur on both state 3’s self-loop and also on the transition from state 3 to state 2. If specifiers decide that the state 3 to state 2 transition exists, then either the “¬(A,START,END)” multi-label item or the “¬(START,END)” multi-label item must be the one on state 3’s self-loop. Which one is selected depends on the setting of the Bounded option in the Existence behavior template. Similarly, the transition from state 5 to state 2 on *START* would also exist because it is based on the same option setting as the transition from state 3 to state 2. For similar reasons of determinism, the “¬(START,END)” multi-label item must thus be the one on state 5’s self-loop. If specifiers instead decide that the state 3 to state 2 transition does not exist, then either the “¬(A,END)” multi-label item or the “¬END” multi-label item must be the one on state 3’s self-loop. In addition, the transition from state 5 to state 2 on *START* would also not exist, and the “¬END” multi-label item must thus be the one on state 5’s self-loop.

Applying the Between scope template to the Existence FSA template also affects the multi-label items. Because the set-complement operator (“¬”) means that everything that is not in the set specified by the multi-label item is accepted, *END* needs to be added to each of those multi-label items in which the set-
complement operator is used in the original behavior FSA template. END is not permitted to occur on state 2’s self-loop, and thus it is added to the original “¬A” multi-label item that was on that self-loop in the Existence behavior FSA template, and the new multi-label item therefore becomes “¬(A,END)”. The same transformation happens on state 3’s self-loop. Note that the third multi-label item in state 3’s self-loop is also changed from the whole set of events (”.”) to the set of events excluding END (”¬END”). In fact, END is excluded from all the multi-label items on state 3’s self-loop, to preserve determinism in the fully-resolved FSA, since there is always an outgoing transition from state 3 to state 4 on END. Similarly, the “.” multi-label item on state 5’s self-loop becomes “¬END” and END is added to the “¬(START,END)” multi-label item, because there is always an outgoing transition from state 5 to the non-accepting trap state on END.

At this point, we are left with the question of what happens when the FSA template is in state 3 and END occurs. In this situation, the FSA template takes the transition to state 4, an accepting state, because the behavior holds within that restricted interval. There is one more option in the Between scope template, called single/multiple restricted intervals, which determines whether more than one restricted interval can exist in a single event sequence. This option is only available in the Between scope template. There are two alternative settings associated with this option, which are based on two possibilities of what can happen after the end of a restricted interval. One possibility is that a subsequent occurrence of START potentially begins a new restricted interval and thus multiple restricted intervals can exist consecutively in the event sequence. For this setting, an occurrence of START in state 4 takes the FSA from state 4 to state 2, returning to a situation where the behavior again must hold before the first subsequent END occurs. In this situation, the “¬START” multi-label item must be the one on state 4’s self-loop, to preserve the determinism of the fully-resolved FSA. The other possibility is that a subsequent occurrence of START does not begin a new restricted interval and thus there can be only one restricted interval in the event sequence. For this setting, an occurrence of START in state 4 just self-loops, the multi-label item must thus be “.”, and state 4 is treated as an accepting trap state.

A.3 Disciplined Natural Language Templates

The DNL template property view notation that is used in this appendix is defined in Section 3.1.4.1. Like the other property views, the DNL templates can help specifiers to resolve the options that are associated with the selected property pattern template. Thus, the same options that must be resolved in the QT and FSA template property views, and in the ST and SST scope views, are options in the DNL template property view as well.
This property is composed of two parts: a behavior and a scope. The behavior specifies the restrictions on the occurrences of events, and the scope specifies the interval(s) in the event sequences within which the behavior restrictions are required to hold. This property does not impose any restrictions on the occurrences of events that are not explicitly mentioned in this property specification.

### Preamble
1. The Event(s) of primary interest in this behavior is A and B

### Scope Header
**SCOPE:**

### Scope Body
- **Scope DNL:**
  - Global scope template
  - After scope template
  - Before scope template
  - Between scope template
- **Global DNL Template:**
- **After DNL Template:**
- **Before DNL Template:**
- **Between DNL Template:**

### Behavior Header
**BEHAVIOR:**

### Primary Events Phrase
1. The Event(s) of primary interest in this behavior is A and B

### Secondary Events Phrase
2. How many event(s) of secondary interest in this behavior is C_1, C_2, ..., C_{n-1}, and C_n

### Behavior Body
- **Behavior DNL:**
  - Response behavior template
  - Precedence behavior template
  - Absence behavior template
  - Existence behavior template
- **Response DNL Template:**
- **Precedence DNL Template:**
- **Absence DNL Template:**
- **Existence DNL Template:**

---

**Figure A.12.** Outline of How a Property is Represented in DNL

### A.3.1 Outline of a Property in DNL

As was mentioned in Section 3.2.4, to aid the readability of a combined scope-and-behavior DNL template property view, we divide the combined DNL template into three main sections, in the following manner:

1. The preamble
2. The description of the scope
3. The description of the behavior

Figure A.12 shows the details of this DNL template property view structure. The way that the DNL template property view is portrayed in this figure differs from its presentation in PROPEL, however. For the purpose of this discussion, this figure shows all of the alternatives available for each phrase, and shows the scope template, behavior template, or number of secondary events that is represented by each alternative. This
figure and all other figures in this DNL template appendix use a graphical notation that is analogous to Backus-Naur Form. Beyond defining the derivation rules, this graphical notation also includes information about which terminals and non-terminals are to be selected, given the specifiers’ decision (i.e., the choice of a scope template, behavior template, number of secondary events, or option setting(s)).

Each phrase in the DNL template property view structure shown in Figure A.12 has a name, which is shown in bold text in the top left corner of the phrase’s area, and everything that makes up that phrase is contained with that phrase’s area. The background color of the phrase areas alternate between white and gray, to delimit the boundaries of each phrase’s area. The main structure of each phrase is shown in the box that begins in the top left corner of that phrase’s area, directly underneath the phrase’s name. The main structure of each phrase is made up of terminals and/or non-terminals. The name of each non-terminal is given in italicized, title-case text and surrounded by a small box. The set of possible expansions for that non-terminal is given in the larger box that is linked to that non-terminal’s small box by a dotted line. There are two columns in these larger boxes: the column on the left contains a shorthand description of the specifiers’ decision and the column on the right contains the set of terminals and/or non-terminals that matches that decision.

The DNL template property view is composed of seven phrases, in the order given in Figure A.12. The Preamble phrase is optional; PROPEL specifiers have the choice of whether to display this explanatory text. When using multiple properties’ DNL specifications in requirements documents, for example, specifiers might want to give the Preamble phrase only once. The Preamble phrase does not contain any non-terminals. The Scope Header phrase is just the single word shown, “SCOPE:,” and it does not contain any non-terminals. The Scope Body phrase contains one non-terminal, “Scope DNL,” which can be expanded to one of four non-terminals, each of which corresponds to one of the scope DNL templates shown in Figures A.13, A.14, A.15, and A.16, respectively. The Behavior Header phrase is just the single word shown, “BEHAVIOR:,” and it does not contain any non-terminals. The Primary Events phrase is common to all behavior DNL templates and is thus only shown here. It contains two non-terminals, which change based on whether the behavior template that has been selected uses one or two primary events. The Secondary Events phrase, which is also common to all behavior templates, contains two non-terminals, which change based on the number of secondary events that specifiers have added to the property’s alphabet. The Behavior Body phrase contains one non-terminal, “Behavior DNL,” which can be expanded to one of four non-terminals, each of which corresponds to one of the behavior DNL templates shown in Figures A.17, A.18, A.19, and A.20, respectively. We now

\[4\] The main structures of the Scope Header and Behavior Header phrases do not denote non-terminals, since they are given in capitalized text.
Core Phrase

1. From the start of any event sequence through to the end of that event sequence, the behavior must hold.

Figure A.13. Global DNL Template

look in more detail at how the non-terminals in the Scope Body phrase and in the Behavior Body phrase are represented in the DNL template property view, for each of the scope and behavior templates.

A.3.2 Scope DNL Templates

This appendix gives the details of how all four scope templates are represented in the DNL template property view and how all of the scope templates’ options can be resolved using the scope DNL templates. Figures A.13, A.14, A.15, and A.16 each show one of the scope DNL templates. The way that these templates are portrayed in these figures differs from their presentation in PROPEL, however. For the purpose of this discussion, these figures show all of the alternatives available for each phrase, and show the option setting(s) that are represented by each alternative, using the notation that is given in Figure A.12. For a more realistic example of how specifiers interact with a scope DNL template to specify a property, see Section 3.2.4.1. The following sections discuss each of the scope DNL templates in more detail.

A.3.2.1 The Global Scope DNL Template

Since the Global DNL template does not have any options, it consists only of a single Core phrase that describes the core concept of the Global scope template and there are no non-terminals associated with this phrase. Figure A.13 provides a full description of this Core phrase, using the notation that is given in Figure A.12.

A.3.2.2 The After Scope DNL Template

The After DNL template consists of four phrases: the Summary phrase, the Core phrase, the “If no START” phrase, and the “Before first START” phrase. None of the phrases in the After DNL template has any synonyms. Figure A.14 provides a full description of the After DNL template, using the notation that is given in Figure A.12. The Summary phrase is shown first in the After DNL template, and there are no non-terminals associated with this phrase. The Core phrase is shown next, and it represents the first/last START option. Since there are two option settings for that option, PROPEL specifiers have two choices in the DNL template for this Core phrase. There are three non-terminals in this Core phrase that change based on that option, but only the “First/Last 1” non-terminal corresponds to a combo box in PROPEL’s implementation of this scope DNL template. When specifiers select one of the option settings in that combo box, all three
Summary Phrase
1. There can be at most one restricted interval in the event sequence and it has a starting delimiter, START.

Core Phrase

2. The behavior is required to hold from the first occurrence of START through to the end of the event sequence. If START occurs more than once before the end of the event sequence, only the first occurrence of START begins the restricted interval; later occurrences of START do not have an effect.

If no START Phrase
3. START is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence.

Before first START Phrase
4. There are no restrictions imposed on the occurrences of any events before the first occurrence of START, if it ever occurs.

Summary Phrase
1. There can be at most one restricted interval in the event sequence and it has an ending delimiter, END.

Core Phrase

2. The behavior is required to hold from the start of the event sequence through to the first occurrence of END, if it ever occurs.

If no END Phrase
3. END is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence.

After first END Phrase
4. There are no restrictions imposed on the occurrences of any events after the first occurrence of END, if it ever occurs.

Figure A.14. After DNL Template

non-terminals change to match the selection. The “If no START” phrase is shown next, and there are no non-terminals associated with this phrase. The final phrase in the After DNL template is the “Before first START” phrase, and it has one non-terminal. This non-terminal changes based on the first/last START option, and when specifiers select an option setting for that option in the Core phrase, this non-terminal changes to match the selection.
A.3.2.3 The Before Scope DNL Template

The Before DNL template consists of four phrases: the Summary phrase, the Core phrase, the “If no END” phrase, and the “After first END” phrase. None of the phrases in the Before DNL template has any synonyms. Figure A.15 provides a full description of the Before DNL template, using the notation that is given in Figure A.12. The Summary phrase is shown first in the Before DNL template, followed by the Core phrase. The “If no END” phrase represents the missing END option. Since there are two option settings for that option, PROPEL specifiers have two choices in the DNL template for this phrase. There is one non-terminal in this phrase that changes based on that option. This is the only phrase that has a non-terminal in the Before DNL template. The final phrase is the “After first END” phrase.

A.3.2.4 The Between Scope DNL Template

The Between DNL template consists of six phrases: the Summary phrase, the Core phrase, the “First/Last START” phrase, the “If no END” phrase, the Repeatability phrase, and the “No Restrictions” phrase. None of the phrases in the Between DNL template has any synonyms. Figure A.16 provides a full description of the Between DNL template, using the notation that is given in Figure A.12. The Summary phrase is shown first in the Between DNL template, and there are no non-terminals associated with this phrase. The Core phrase is shown next, and it has one non-terminal that changes based on the settings of two options, first/last START and single/multiple restricted intervals. Strictly speaking, however, this non-terminal does not truly represent either of those options, since it does not distinguish between all the possible combinations of their settings. The “First/Last START” phrase has four non-terminals. “First/Last 1” and “First/Last 2” represent the first/last START option, and since there are two option settings for that option, PROPEL specifiers have two choices in the DNL template for those non-terminals. Only the “First/Last 1” non-terminal corresponds to a combo box in PROPEL’s implementation of this scope DNL template, however. When specifiers select one of the option settings in that combo box, both non-terminals change to match the selection. The “Potentially” non-terminal in the “First/Last START” phrase is affected by the missing END option, and the “Article” non-terminal is affected by the single/multiple restricted intervals option. Likewise, the non-terminals inside the “First/Last 2” non-terminal are also affected by that option.

The next phrase, “If no END”, represents the missing END option. Since there are two option settings for that option, PROPEL specifiers have two choices in the DNL template for this phrase. The Repeatability phrase, which appears next in the Between DNL template, represents the single/multiple restricted intervals option. Since there are two option settings for that option, PROPEL specifiers have two choices in the DNL template for this phrase. Although Figure A.16 is technically correct, this phrase is actually not implemented in PROPEL as it is shown in that figure, because doing so would require that this entire phrase appear as a
large combo box with all the text of both alternatives shown in it. For the purposes of helping specifiers to focus on the key difference between the option settings represented in this phrase, the first sentence from both of the alternatives is always shown to the specifier, and the combo box displays only the choice between “can be at most one restricted interval” and “might be many restricted intervals”. If specifiers select the second choice, then the remainder of the text for that alternative, including its non-terminal, is shown to the specifiers, outside of the combo box. Otherwise, only the first sentence is shown to the specifiers. The final phrase in the Between DNL template is the “No Restrictions” phrase, and there are no non-terminals associated with this phrase.

A.3.3 Behavior DNL Templates

This appendix gives the details of how all four behavior templates are represented in the DNL template property view and how all of the behavior templates’ options can be resolved using the behavior DNL templates. Figures A.17, A.18, A.19, and A.20 each show one of the behavior DNL templates. The way that these templates are portrayed in these figures differs from their presentation in PROPEL, however. For the purpose of this discussion, these figures show all of the alternatives available for each phrase, and show the option setting(s) that are represented by each alternative, using the notation that is given in Figure A.12. For a more realistic example of how specifiers interact with a behavior DNL template to specify a property, see Section 3.1.4.2. The following sections discuss each of the behavior DNL templates in more detail.

A.3.3.1 The Response Behavior DNL Template

In addition to the phrases described in Figure A.12 that are common to all the behavior templates, the Response DNL template consists of a Core phrase and four subsidiary phrases: the Precedency phrase, the Nullity phrase, the Immediacy & Pre-arity (IP) phrase, and the Post-arity, Repeatability, & Finalization (PRF) phrase. None of the phrases in the Response DNL template has any synonyms. Figure A.17 provides a full description of the five phrases that are specific to the Response DNL template, using the notation that is given in Figure A.12.

The Core phrase is shown first in the Response DNL template, and then the next phrase represents the Precedency option. Since there are two option settings for that option, PROPEL specifiers have two choices in the DNL template for this Precedency phrase. If there are secondary events in the behavior, this phrase is expanded to contain the statement about those secondary events. This expanded form is the one that is shown

---

5 Although the number of secondary events in the behavior is not actually considered an option, the DNL changes based on it, so those parts of the phrases that refer to secondary events are shown with the same notation as the options. This convention is followed for all the non-terminals in the DNL template figures.
Summary Phrase

1. A restricted interval in the event sequence can have both a starting delimiter, START, and an ending delimiter, END.

Core Phrase

<table>
<thead>
<tr>
<th>First/Last START Phrase</th>
<th>first START</th>
<th>first</th>
<th>last START</th>
<th>last</th>
</tr>
</thead>
<tbody>
<tr>
<td>If no END Phrase</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Repeatability Phrase</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>No Restrictions Phrase</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

First/Last START Phrase

3. If there are multiple occurrences of START without an occurrence of END in between them, only the First/Last of those occurrences of START Potentially starts Article restricted interval; First/Last 2

If no END Phrase

4. START is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence.

Even if START does occur, END is not required to occur subsequently Interval Existence

Repeatability Phrase

5. Scope Repeatability

<table>
<thead>
<tr>
<th>single interval</th>
<th>multiple intervals</th>
</tr>
</thead>
</table>
| There can be at most one restricted interval in the event sequence. | There might be many restricted intervals in the event sequence. If START occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of START, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a Potential Interval new restricted interval.

No Restrictions Phrase

6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

**Figure A.16. Between DNL Template**

in Figure A.17. The next phrase in the Response DNL template represents the Nullity option, and since there are two option settings for that option, PROPEL specifiers have two choices for this Nullity phrase. Next, the Immediacy and Pre-arity options are concerned with what is allowed to occur after an occurrence of A and before the first subsequent occurrence of B, so they are grouped together into the one IP phrase. The parts
### Core Phrase

3. If \( A \) occurs, \( B \) is required to occur subsequently.

### Precedency Phrase

<table>
<thead>
<tr>
<th>Precedency</th>
<th>B is not allowed to occur;</th>
<th>B is allowed to occur zero or more times;</th>
</tr>
</thead>
<tbody>
<tr>
<td>secondary:one ( A ) is</td>
<td>secondary:one ( C_1 ) is</td>
<td>secondary:many All the events of secondary interest are</td>
</tr>
</tbody>
</table>

### Nullity Phrase

5. \[ Nullity \]

<table>
<thead>
<tr>
<th>Nullity</th>
<th>( A ) is not required to occur.</th>
<th>( A ) is required to occur.</th>
</tr>
</thead>
<tbody>
<tr>
<td>nullity:no</td>
<td>nullity:yes</td>
<td></td>
</tr>
</tbody>
</table>

### Immediacy & Pre-arity Phrase

6. After \( A \) occurs, but before the first subsequent \( B \) occurs:

<table>
<thead>
<tr>
<th>Pre-arity</th>
<th>( A ) is not allowed to occur again;</th>
<th>( A ) is allowed to occur again, zero or more times;</th>
<th>( B ) is allowed to occur zero or more times;</th>
</tr>
</thead>
<tbody>
<tr>
<td>secondary:one ( C_1 ) is not</td>
<td>secondary:many None of the events of secondary interest are</td>
<td></td>
<td></td>
</tr>
<tr>
<td>secondary:one ( C_1 ) is</td>
<td>secondary:many All the events of secondary interest are</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Post-arity, Repeatability, & Finalization Phrase

7. After \( A \) and the first subsequent \( B \) occur:

<table>
<thead>
<tr>
<th>Post-arity</th>
<th>( B ) is allowed to occur again, zero or more times, before another ( A ) occurs;</th>
<th>( A ) is allowed to occur again and, if it does, then the situation is the same as when the first ( A ) occurred, meaning that the restrictions described in parts 3, 6, and 7 would again apply.</th>
</tr>
</thead>
<tbody>
<tr>
<td>secondary:many ( C_1 ) is</td>
<td>secondary:many All the events of secondary interest are</td>
<td></td>
</tr>
</tbody>
</table>

### Figure A.17. Response DNL Template
of that phrase that represent the *Immediacy* option refer to secondary events, but if there are no secondary events in the behavior then those parts of the phrase are not visible, and only the two choices for the two *Pre-arity* option settings are visible. Since there are two option settings apiece for both options, when there are secondary events in the behavior, PROPEL specifiers have four choices in the Response DNL template for this phrase. This expanded form is the one that is shown in Figure A.17. The next phrase represents the *Post-arity, Repeatability, and Finalization* options. Since these three options are all concerned with what is allowed to occur after A and the first subsequent B occur, they are grouped into the one PRF phrase in the Response DNL template. Only six combinations of those three options’ settings are logically consistent, so PROPEL specifiers have six choices in the DNL template for this phrase. If there are secondary events in the behavior, this phrase is expanded to contain a statement about those secondary events. This expanded form is the one that is shown in Figure A.17.

**A.3.3.2 The Precedence Behavior DNL Template**

The Precedence DNL template is very similar in structure to the Response DNL template. In addition to the phrases shown in Figure A.12 that are common to all the behavior templates, the Precedence DNL template consists of a Core phrase and five subsidiary phrases: the Before First ‘A’ (BFA) phrase, the Nullity phrase, the Not Response (NR) phrase, the Immediacy & Pre-arity (IP) phrase, and the Post-arity, Repeatability, & Finalization (PRF) phrase. None of the phrases in the Precedence DNL template has any synonyms. Figure A.18 provides a full description of the six phrases that are specific to the Precedence DNL template, using the notation that is given in Figure A.12.

The Core phrase is shown first in the Precedence DNL template, and then the BFA phrase is shown next, but it is only visible if there are secondary events in the behavior. This expanded form is the one that is shown in Figure A.18. The next phrase in the Precedence DNL template represents the *Nullity* option, and since there are two option settings for that option, PROPEL specifiers have two choices for this Nullity phrase. This phrase is similar to the Nullity phrase in the Response DNL template, except that the representation of the “nullity:yes” option setting adds text to clarify that “A is required to occur, whether or not B eventually occurs.” The next phrase in the Precedence DNL template is the NR phrase, which explicitly states that there is no Response-type restriction on the interaction of A and B. This is included in the Precedence DNL template as a parallel to the Precedency phrase in the Response DNL template, since that phrase explicitly states whether there is a Precedence-type restriction on the interaction of A and B. Part of this phrase is only visible if specifiers set the *Nullity* option to not require A to occur. The IP phrase in the Precedence DNL template has the same two options and associated terminals and non-terminals as was described for the IP phrase in the Response DNL template. Finally, the PRF phrase in the Precedence DNL template has the
### Core Phrase

3. B is not allowed to occur until after A occurs.

### Before First 'A' Phrase

4. Before the first A occurs, secondary events allowed to occur zero or more times.

<table>
<thead>
<tr>
<th>pre-arity:one</th>
<th>immediacy:no</th>
<th>secondary:one</th>
<th>C₁ is not allowed to occur again; secondary events allowed to occur.</th>
</tr>
</thead>
<tbody>
<tr>
<td>pre-arity:many</td>
<td>immediacy:no</td>
<td>secondary:many</td>
<td>None of the events of secondary interest are allowed to occur.</td>
</tr>
<tr>
<td>pre-arity:many</td>
<td>immediacy:yes</td>
<td>secondary:one</td>
<td>C₁ is allowed to occur again, zero or more times; secondary events allowed to occur.</td>
</tr>
<tr>
<td>pre-arity:many</td>
<td>immediacy:yes</td>
<td>secondary:many</td>
<td>All the events of secondary interest are allowed to occur.</td>
</tr>
</tbody>
</table>

### Nullity Phrase

5. Nullity

| nullity:no | A is not required to occur. |
| nullity:yes | A is required to occur, whether or not B eventually occurs. |

### Not Response Phrase

6. Nullity Conditional

| nullity:no | Even if A does occur, |
| nullity:yes | |

### Immediacy & Pre-arity Phrase

7. After A occurs, but before the first subsequent B occurs:

<table>
<thead>
<tr>
<th>pre-arity:one</th>
<th>immediacy:yes</th>
<th>secondary:one</th>
<th>C₁ is not allowed to occur again; secondary events allowed to occur.</th>
</tr>
</thead>
<tbody>
<tr>
<td>pre-arity:many</td>
<td>immediacy:yes</td>
<td>secondary:many</td>
<td>None of the events of secondary interest are allowed to occur.</td>
</tr>
<tr>
<td>pre-arity:many</td>
<td>immediacy:yes</td>
<td>secondary:many</td>
<td>All the events of secondary interest are allowed to occur.</td>
</tr>
</tbody>
</table>

### Post-arity, Repeatability, & Finalization Phrase

8. After A and the first subsequent B occur:

<table>
<thead>
<tr>
<th>post-arity:many</th>
<th>repeatability:yes</th>
<th>finalization:yes</th>
<th>secondary:one</th>
<th>C₁ is allowed to occur again, zero or more times, before another A occurs; secondary events allowed to occur zero or more times;</th>
</tr>
</thead>
<tbody>
<tr>
<td>post-arity:one</td>
<td>repeatability:yes</td>
<td>finalization:yes</td>
<td>secondary:one</td>
<td>C₁ is allowed to occur again until after another A occurs; secondary events allowed to occur zero or more times;</td>
</tr>
<tr>
<td>post-arity:many</td>
<td>repeatability:yes</td>
<td>finalization:yes</td>
<td>secondary:one</td>
<td>C₁ is allowed to occur again and, if it does, then the situation is the same as when the first A occurred, meaning that the restrictions described in parts 6, 7, and 8 would again apply.</td>
</tr>
<tr>
<td>post-arity:one</td>
<td>repeatability:yes</td>
<td>finalization:yes</td>
<td>secondary:many</td>
<td>All the events of secondary interest are allowed to occur zero or more times.</td>
</tr>
<tr>
<td>post-arity:many</td>
<td>repeatability:no</td>
<td>finalization:yes</td>
<td>secondary:many</td>
<td>All the events of secondary interest are allowed to occur zero or more times.</td>
</tr>
</tbody>
</table>

### Figure A.18. Precedence DNL Template
same three options and associated terminals and non-terminals as was described for the PRF phrase in the Response DNL template, with one minor difference. Since there is one more phrase in the Precedence DNL template than there is in the Response DNL template, the numbering of all the phrases in the Precedence DNL template is changed accordingly, and the phrase-number references in the first two choices of the Precedence DNL template’s PRF phrase are thus changed to match.

A.3.3.3 The Absence Behavior DNL Template

In addition to the phrases shown in Figure A.12 that are common to all the behavior templates, the Absence DNL template consists of a Core phrase. The Core phrase has one non-terminal, called Prevented, which has two synonyms. We thus provide two different ways of expressing the same information in the Core phrase. Figure A.19 gives a full description of this phrase and its available choices, using the notation that is given in Figure A.12. The two synonyms that express the Core phrase are shown together, but they are separated in the larger box by a thin line, rather than by a bold line, to indicate that they have the same meaning.

A.3.3.4 The Existence Behavior DNL Template

In addition to the phrases shown in Figure A.12 that are common to all the behavior templates, the Existence DNL template consists of a Core phrase, which has two non-terminals. One non-terminal is called
Required and it provides specifiers with a choice of two synonyms to express the core concept of the Existence behavior template. The other non-terminal is an option, called Bounded, which is concerned with whether A is allowed to occur exactly once or more than once. Both of the settings for the Bounded option have two synonyms apiece. Figure A.20 provides a full description of the Existence DNL template and its available choices, using the notation that is given in Figures A.12 and A.19.

A.4 Scope Timeline

The ST scope view notation that is used in this appendix is defined in Figure 3.32, in Section 3.2.5. The ST scope view can help specifiers to resolve the options that are associated with the selected scope template. Thus, the same scope-template options that must be resolved in the QT, FSA template, and DNL template property views, and in the SST scope view, are options in the ST scope view as well. This appendix gives the details of how all four scope templates are represented in the ST scope view, and how all of the scope templates’ options can be resolved using this scope view. Central to this discussion are Figures A.21, A.22, A.23, and A.24, which each show one of the scope templates’ timeline representation. There is one aspect of the how the scope templates’ timelines are portrayed in these figures that differs from their presentation in PROPEL. For the purposes of the discussion in this appendix, all of the scope templates’ options are fully resolved and the possible combinations of their option settings are shown side-by-side. In these figures, the name of the scope template and any option setting(s) that the scope timeline represents are shown in a box in the upper left-hand corner of each timeline. For a more realistic example of how specifiers interact with an ST to specify a property, see Section 3.2.5.1. The following sections discuss each of the scope templates’ timelines in more detail.

A.4.1 The Global Scope Timeline

There is only one scope timeline for the Global scope, as shown in Figure A.21, because there are no options in the Global scope template.
A.4.2 The After Scope Timelines

There are two scope timelines for the After scope, as shown in Figure A.22, because there is one option, _first/last START_, in the After scope template. There is one timeline for each of the option settings. Note that the difference between the two timelines is where the bottom arrow, which is the restricted-interval arrow, starts. The leftmost instance of the START delimiter denotes the first occurrence of START, and the rightmost instance denotes the last occurrence of START.

A.4.3 The Before Scope Timelines

There are two scope timelines for the Before scope, as shown in Figure A.23, because there is one option, _missing END_, in the Before scope template. There is one timeline for each of the option settings. Note that the difference between the two timelines is where the bottom arrow ends. When the bottom arrowhead stops just before the occurrence of the END delimiter, this indicates that the behavior is not required to hold if END does not occur. Alternatively, when the bottom arrowhead stops after the instance of the END delimiter, this indicates that the behavior is required to hold throughout the entire event sequence if END does not occur.
A.4.4 The Between Scope Timelines

There are four scope timelines for the Between scope, as shown in Figure A.24, because the ST represents two of the options, *first/last START* and *missing END*, that are in the Between scope template. There is one timeline for each combination of the two options’ settings, and those combinations are shown using the same notation that was described in Sections A.4.2 and A.4.3. There is a third option in the Between scope template that is not shown in the current version of the ST scope view, however: the *single/multiple restricted intervals* option. Because this third option cannot be represented in the current ST scope view, a revised version of the ST scope view is presented in the following section.

A.4.5 Revised Scope Timeline

Figure A.25 gives the notation used in the revised ST scope view and explains the notation using the equivalent DNL descriptions. As in the original ST notation, shown in Figure 3.32 in Section 3.2.5, time flows from left to right, in the direction of the timeline arrow, which is the top arrow. The notation for this arrow is given in the top left corner of Figure A.25, and it is unchanged from the original ST notation. The restricted interval, which defines the chosen scope, is denoted by the bottom arrow. The notation for this arrow is expanded beyond what is defined in the original ST notation for the restricted-interval arrow, and the differences will be discussed in the following sections. Unlike the timeline arrow, the restricted-interval arrow can change its appearance and its graphical starting and ending locations, based on the presence or absence of the starting and ending delimiters, and based on the option settings for the selected scope template.

Similar to what is shown in Figure 3.33 in Section 3.2.5.1, specifiers can interact with the revised ST scope view via contextual menus whose contents are dependent on which of the scope delimiters are present and what the option settings currently are. Specifiers can click on different areas in the revised ST scope view to select a scope template (i.e., by adding or removing the scope delimiters) and to resolve the options in the selected template. The contextual menus that enable specifiers to make these decisions are not shown in this appendix. In addition to the expanded notation for the restricted-interval arrow, the revised ST notation differs from the original ST notation in two major ways. The revised ST notation explicitly shows when each option is unresolved, and it can represent the *single/multiple restricted intervals* option. The details of how each option and its settings are represented in the revised ST notation are discussed in the following sections.

A.4.5.1 The Revised Global Scope Timeline

The Global scope timeline in this revised notation remains the same as what is described in Section A.4.1.
A.4.5.2 The Revised After Scope Timelines

The After scope timelines in this revised notation are changed in three ways from their representation in the original ST notation, and the changes are shown in Figure A.26. Rather than putting two instances of START on the timeline, there is only one instance. Thus, the restricted-interval arrow always starts at that one instance. Another change is that the label representing START is shortened to “S” when START is not associated with an event. The third change is the one that represents the how the first/last START option is set. If the option is unresolved, then there is a “?” above the “S”. Otherwise, the word above the “S” corresponds to one of the other two possible option settings.
A.4.5.3 The Revised Before Scope Timelines

The Before scope timelines in this revised notation differ significantly from their representation in the original ST notation, and the changes are shown in Figure A.27. One minor change is that the label representing END is shortened to “E” when END is not associated with an event. The major change is that each option setting for the Before scope template option, missing END, is represented by two separate timelines,
rather than just one. The reason for this is to clearly show the difference between what happens when END occurs in the event sequence and when it does not. Thus, the first timeline shown in each of the four large boxes in Figure A.27 is preceded by the phrase “If END does occur;” and is always the same. When END occurs, the restricted interval always ends before the first occurrence of END, and thus the restricted-interval arrowhead stops before the END delimiter on those first timelines.

The interesting differences all occur in the second timeline shown in each of the four large boxes. The second timeline is preceded by the phrase “If END does not occur;”. In the large box in the upper left-hand corner of Figure A.27, the second timeline shows that the missing END is unresolved, using notation similar to the optional-transition notation in the FSA template property view. In that property view, when an option transition is unresolved, the line for that optional transition is dashed. The analogy holds for this revised ST notation, because the restricted-interval arrow can either be made to exist (i.e., the option setting shown in the large box in the bottom right-hand corner of Figure A.27), or it can be made to not exist (i.e., the option setting shown in the large boxes in the bottom left-hand and upper right-hand corners of Figure A.27). The reason that this option setting is shown two different ways is to extend the analogy from the FSA template property view and to give specifiers the ability to remove visual clutter if they want to. In the FSA template property view, specifiers have the choice of showing rejected transitions as grayed out, or of having rejected transitions completely removed, visually speaking. Thus, in this revised ST notation, specifiers have a similar choice.

A.4.5.4 The Revised Between Scope Timelines

The Between scope timelines in this revised notation differ significantly from their representation in the original ST notation, and the changes are shown in Figures A.28-A.32. The Between scope timelines incorporate all the changes from the After scope timelines and the Before scope timelines and include some additional notation to represent the combination of the options that those other scope templates’ timelines represent. The Between scope timelines represent the first/last START option with a single instance of START and the restricted-interval arrow always starts at that one instance. If the option is unresolved, then there is a “?” above the “S”. Otherwise, the word above the “S” corresponds to one of the other two possible option settings. The Between scope timelines represent the missing END option with two separate timelines, rather than just one. The first timeline shown in each large box within Figures A.28-A.32 is preceded by the phrase “If END does occur;” and is always the same. The instance of END on that first timeline represents the first occurrence of END after an occurrence of START. Since the first timeline represents the case where the given restricted interval ends at an occurrence of END, that timeline always shows the restricted-interval arrow. The second timeline shown in each large box within Figures A.28-A.32 is preceded by the phrase “If
**END** does not occur:” and represents the possible missing **END** option settings in a way similar to how that option is represented in the Before scope timelines. The only difference in the Between scope timelines is that the restricted interval arrow, when it is visible, always starts at the instance of **START** on the timeline, rather than at the start of the event sequence. The label on the instance of **START** on the second timeline always matches the label on the instance of **START** on the first timeline, showing the same setting for the first/last **START** option.

The one new notation that the Between scope timelines introduce is the way that the **single/multiple restricted intervals** option is represented. When this option is unresolved, the first timeline that is shown in the large boxes within Figures A.28-A.32 shows the vertical bar that represents the end of the event sequence as a dashed line. This notation is similar to the optional-transition notation in the FSA template property view and to the notation used to show that the missing **END** option is unresolved in the second timeline. The **single/multiple restricted intervals** option has two option settings. For the setting where there can be at most one restricted interval in the event sequence, the vertical bar looks the same as it did in the original ST notation. For the setting where there can be multiple restricted intervals in the event sequence, the vertical bar is changed to show double vertical lines with a looping arrow on top. The double vertical lines draw on the analogy of musical notation for showing repetition, and the looping arrow is included to bolster that idea of repetition.

Even with all these changes, this revised ST scope view is still a work in progress. Despite the fact that there is now notation in this revised ST to represent the **single/multiple restricted intervals** option, this notation represents multiple restricted intervals with a relatively subtle change in the ST scope view. An alternative scope view that more obviously shows repetition is a Scope Statechart Template, which is briefly discussed in the following section.
Figure A.28. Revised Between Scope Timelines (Part 1)
Figure A.29. Revised Between Scope Timelines (Part 2)
If END does occur:
Between scope, first START, restricted interval needs END (rejected interval is grayed out), multiple restricted intervals:

If END does not occur:
Between scope, first START, restricted interval needs END (rejected interval is grayed out), single restricted interval:

If END does occur:
Between scope, first START, restricted interval needs END (rejected interval is not visible), single/multiple restricted intervals unresolved:

If END does not occur:
Between scope, first START, restricted interval exists without END, single/multiple restricted intervals unresolved:

If END does occur:
Between scope, first START, restricted interval exists without END, single restricted interval:

If END does not occur:

Figure A.30. Revised Between Scope Timelines (Part 3)
Figure A.31. Revised Between Scope Timelines (Part 4)
<table>
<thead>
<tr>
<th>Scenarios</th>
</tr>
</thead>
<tbody>
<tr>
<td>**Between scope, first/last START unresolved, missing END unresolved,</td>
</tr>
<tr>
<td>single/multiple restricted intervals unresolved:**</td>
</tr>
<tr>
<td>If END does occur:</td>
</tr>
<tr>
<td><img src="image1" alt="Timeline Diagram" /></td>
</tr>
<tr>
<td>If END does not occur:</td>
</tr>
<tr>
<td><img src="image2" alt="Timeline Diagram" /></td>
</tr>
<tr>
<td>**Between scope, first/last START unresolved, missing END unresolved,</td>
</tr>
<tr>
<td>single/multiple restricted intervals unresolved:**</td>
</tr>
<tr>
<td>If END does occur:</td>
</tr>
<tr>
<td><img src="image3" alt="Timeline Diagram" /></td>
</tr>
<tr>
<td>If END does not occur:</td>
</tr>
<tr>
<td><img src="image4" alt="Timeline Diagram" /></td>
</tr>
<tr>
<td>**Between scope, first/last START unresolved, missing END unresolved,</td>
</tr>
<tr>
<td>single/multiple restricted intervals unresolved:**</td>
</tr>
<tr>
<td>If END does occur:</td>
</tr>
<tr>
<td><img src="image5" alt="Timeline Diagram" /></td>
</tr>
<tr>
<td>If END does not occur:</td>
</tr>
<tr>
<td><img src="image6" alt="Timeline Diagram" /></td>
</tr>
</tbody>
</table>

**Figure A.32. Revised Between Scope Timelines (Part 5)**
A multi-label, where "x" and "y" are transition labels, using the same notation as the FSA template property view, with comparable semantics.

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>The initial state</td>
<td></td>
</tr>
<tr>
<td>The final state</td>
<td></td>
</tr>
<tr>
<td>An intermediate state</td>
<td></td>
</tr>
<tr>
<td>A transition</td>
<td></td>
</tr>
<tr>
<td>An optional transition, using the same notation as the FSA template property view, with comparable semantics.</td>
<td></td>
</tr>
<tr>
<td>A transition label: &quot;a&quot; denotes a trigger that causes the transition to occur, and &quot;b&quot; and &quot;c&quot; denote actions that are performed during the transition, in the order given (i.e., &quot;b&quot; first and &quot;c&quot; second). There can be 1..n triggers and 0..m actions in a transition label.</td>
<td></td>
</tr>
<tr>
<td>A multi-label, where &quot;x&quot; and &quot;y&quot; are transition labels, using the same notation as the FSA template property view, with comparable semantics.</td>
<td></td>
</tr>
</tbody>
</table>

Figure A.33. Scope Statechart Template Notation

**A.5 Scope Statechart Templates**

The SST scope view can help specifiers to resolve the options that are associated with the selected scope template. Thus, the same scope-template options that must be resolved in the QT, FSA template, and DNL template property views, and in the ST scope view, are options in the SST scope view as well. Unlike the ST scope view, the SST scope view does not enable specifiers to select a scope template; it can only be used after a scope template has been selected in either the QT property view or the ST scope view. A discussion of the design goals for the property views and the scope views, including the motivation for this SST scope view, is given in Section 7.2.1.3. In addition, Table 7.1 gives a breakdown of each of the design goals that the SST scope view is intended to fulfill.

**A.5.1 Scope Statechart Template Notation**

The SST scope view notation, which is defined in Figure A.33, extends the basic UML State Diagram notation by including optional components that are similar to the FSA template extensions, specifically optional transitions and multi-labels on transitions. Similar to the property views and the ST scope view, SSTs follow the event-based paradigm. A transition label in an SST is made up of two parts: one or more triggers,⁶ which describe what happens in the event sequence, and zero or more actions, which describe what

---

⁶In UML parlance, this “trigger” concept is referred to as an “event,” but we are avoiding the use of the word “event” in this context because events in the rest of PROPEL are a subset of this trigger concept. Triggers are both events (i.e., the specifier-defined events that are associated with START and END) and meta-events (i.e., initiate-sequence, complete-sequence).
happens with any (potential) restricted intervals that are based on that event sequence. There are 4 triggers of interest:

- initiate-sequence: This occurs immediately before the first event in the sequence occurs.
- complete-sequence: This occurs immediately after the last event in the sequence occurs.
- **START**: This is an occurrence of the event associated with the **START** parameter.
- **END**: This is an occurrence of the event associated with the **END** parameter.

There are 4 actions of interest:

- start-interval: Start a potential restricted interval
- end-interval: End that potential restricted interval
- check-behavior: Check whether the behavior holds in that restricted interval (i.e., the restricted interval exists)
- discard-interval: Discard that potential restricted interval (i.e., the restricted interval does not exist)

Each scope template is represented as an SST, using a subset of these triggers and actions. This appendix gives the details of how all four scope templates are represented in the SST scope view, and how all of the scope templates’ options can be resolved using this scope view. Central to this discussion are Figures A.34, A.35, A.36, and A.37, which each show one of the scope templates. For the purposes of the discussion in this appendix, there are two aspects of the SSTs shown in these figures that are not intended to be shown to specifiers. One aspect is that each of the states (excluding the initial state and the final state) are given a unique ID. The other aspect is that each scope template option’s settings are shown in a dark gray box next to the parts of the SST that represent those settings. This SST scope view is in its early stages of development and is not currently in **PROPEL**. The following sections discuss each of the scope templates’ SST view in more detail.

### A.5.2 The Global Scope Statechart Template

The Global scope template is represented in the SST notation as a simple statechart with no optional components, since this scope template has no options. This simple statechart is shown in Figure A.34.
Figure A.34. Global Statechart Template

Figure A.35. After Statechart Template
A.5.3 The After Scope Statechart Template

The After scope template is represented in the SST notation as a statechart template with one optional component, a multi-label on state 2’s self-loop, because there is one option, first/last START, in this scope template. This SST is shown in Figure A.35. When the first/last START option is set such that the first occurrence of START is meant to start the restricted interval, then the multi-label item on state 2’s self-loop will be just the START trigger, with no actions following it. This is because any occurrences of START while the SST is in state 2 are after the first occurrence of START, and thus no changes are made to the restricted interval. When the first/last START option is set such that the last occurrence of START is meant to start the restricted interval, then the multi-label item on state 2’s self-loop will be “START / discard-interval, start-interval.” Thus, when the START trigger occurs, the discard-interval and start-interval actions are performed, in that order. The discard-interval action causes all behavior events that occurred before this new occurrence of START to be ignored with respect to enforcing the behavior. The start-interval action resets the beginning of the restricted interval.8

A.5.4 The Before Scope Statechart Template

The Before scope template is represented in the SST notation as a statechart template with one optional component, a multi-label on the transition from state 1 to the final state, because there is one option, missing END, in this scope template. This SST is shown in Figure A.36. When the missing END option is set such that, if END does not occur, the behavior is required to hold throughout the entire event sequence, then the multi-label item on the transition from state 1 to the final state will be “complete-sequence / end-interval, check-behavior.” Thus, when the complete-sequence trigger occurs, the end-interval and check-behavior actions are performed, in that order. The end-interval action ends the restricted interval and the check-behavior action requires that the behavior had to have held throughout the restricted interval. When the missing END option is set such that, if END does not occur, the behavior is not required to hold anywhere in the event sequence, then the multi-label item on the transition from state 1 to the final state will “complete-sequence / discard-interval.” Thus, when the complete-sequence trigger occurs, the discard-interval action is performed, and it causes all behavior events that occurred during the event sequence to be ignored with respect to enforcing the behavior.
A.5.5 The Between Scope Statechart Template

The Between scope template is represented in the SST notation as a statechart template with four optional components. This SST is shown in Figure A.37. Although there are only three options in this scope template, first/last START, missing END, and single/multiple restricted intervals, there are four optional components because two of them represent the same option. The single/multiple restricted intervals option is represented both by the multi-label on state 3’s self-loop, and by the optional transition from state 3 to state 2. When this option is set such that there can be at most one restricted interval in the event sequence, the multi-label item on state 3’s self-loop will be the START or END trigger, with no actions following the occurrence of either delimiter. In addition, the optional transition from state 3 to state 2 does not exist. This is because any occurrences of either delimiter are after the end-interval action has taken place, and thus no new restricted interval can be started and all occurrences of START or END are ignored with respect to enforcing the behavior. When the single/multiple restricted intervals option is set such that there can be multiple restricted intervals in the event sequence, the multi-label item on state 3’s self-loop will be just the END trigger, with no

---

7Note that although UML State Diagrams allow “guards” on transitions, we do not make use of this feature.

8It should be noted that although this might not strictly be how model checking for a property would be implemented, this SST notation conveys a mental model to specifiers that explains how to understand a selected scope (i.e., how to map from any sequence of events to the restricted intervals that that sequence implies).
Figure A.37. Between Statechart Template
actions following the occurrence of that delimiter. This is because occurrences of END at that state are after the end-interval action has taken place, but before the first subsequent occurrence START. Such occurrences of END are ignored with respect to enforcing the behavior. If START does occur at state 3, it is a trigger on the transition from state 3 to state 2, and it is followed by the start-interval action, which starts a new restricted interval.

The other two options in the Between scope template, first/last START and missing END, are represented in the Between SST in a way that is similar to how the After SST and Before SST represent them, respectively. The multi-label on state 2’s self-loop can be described in the same manner as the After SST’s state 2 is in Section A.5.3. In the Between SST, the transition from state 2 to the final state has the same multi-label as the one on the transition from state 1 to the final state in the Before SST, and it can be described in the same manner as that transition is in Section A.5.4.
APPENDIX B

SUPPORT FOR MULTIPLE PROPERTIES
B.1 Project Tree View Redesign

Figure B.1. Example Project Tree View With Property- and Alphabet-Oriented Layouts

See Section 7.2.1.5 for a discussion of this Project Tree View redesign.
B.2 Summary Views

There are five pre-defined Summary Views available in PROPEL: Full, Project, Alphabet, Property, and Event. Each pre-defined Summary View displays similar information, but organizes it around that particular perspective.

B.2.1 Full View

The Full Summary View displays a tabular representations of an entire project’s structure and contents from the perspective of the root project, including all the comments for every part of the structure. This view differs from the Project Summary View in that all available information is shown in the Full Summary View, but in the Project Summary View, only a subset of the available information is displayed. The Full Summary View includes the following columns:

- Project and Comments: the name of the root project and its associated comments.
- Subproject and Comments: the names of the subprojects and their associated comments. The number of subproject-and-comments column pairs correlates to the depth of the project’s hierarchical structure. A parent subproject spans the height of its child alphabets as well as its child subprojects, with the alphabets appearing above the subprojects.
- Alphabet and Comments: the names of the alphabets and their associated comments, grouped by project/subproject.
- Event and Comments: the names of the events and their associated comments, grouped by alphabet.
- Property and Comments: the names of the properties and their associated comments, grouped by alphabet.
- Behavior: the name of each of the behavior templates used in the properties that are in the project; each behavior template name shown is associated with the property that is in the same row(s), in the Property column to the behavior template’s left. If there is no behavior template chosen yet for a given property, the word “BLANK” is displayed in place of the name of the behavior template.
- Behavior Parameters: the parameters’ names (i.e., A or B), followed by a colon (“:”) and the name of the specifier-defined event associated with each parameter, if there is such an event. If there is no associated event, the word “NOT SET” is displayed in place of the event name.
- Scope: the name of each of the scope templates used in the properties in the project; each scope template shown is associated with the property that is in the same row(s), in the Property column to the scope template’s left.
• Scope Parameters: the parameters’ names (i.e., START or END), followed by a colon (":"), and the name of the specifier-defined event associated with each parameter, if there is such an event. If there is no associated event, the word “NOT SET” is displayed in place of the event name.

An example of the Full Summary View is given in Figure B.2. The other Summary Views each display a subset of the information that is given in the Full Summary View, and they each rearrange that subset in terms of their own particular Summary View focus. The same set of example data that is shown in Figure B.2 is used to populate the examples of the other Summary Views.

B.2.2 Project View

The Project Summary View displays a tabular representation of an entire project’s structure and contents from the perspective of the root project, including all the comments for every part of the structure. This view differs from the Full Summary View in that all available information is shown in the Full Summary View, but in the Project Summary View, only a subset of the available information is displayed. The Project Summary View includes the following columns:

• Project: the name of the root project.

• Subproject: the names of the subprojects. The number of subproject-and-comments column pairs correlates to the depth of the project’s hierarchical structure. A parent subproject spans the height of its child alphabets as well as its child subprojects, with the alphabets appearing above the subprojects.

• Alphabet: the names of the alphabets, grouped by project/subproject.

• Event: the names of the events, grouped by alphabet.

• Property: the names of the properties, grouped by alphabet.

• Project Comments: the comments associated with the root project.

An example of the Project Summary View is given in Figure B.3.

B.2.3 Alphabet View

The Alphabet Summary View displays a tabular representation of a subset of the root project’s structure and content from the perspective of the alphabets in the project. The Alphabet Summary View includes the following columns:

• Project.Subproject(s).Alphabet: the name of the root project, the name(s) of all the subprojects that are parents of the alphabet being shown in a given row of the table, and the name of the alphabet whose information is shown in that row.
• Event: the names of the events, grouped by alphabet.

• Property: the names of the properties, grouped by alphabet.

• Alphabet Comments: the comments that are associated with the alphabet that is shown in the same row in the table.

An example of the Alphabet Summary View is given in Figure B.4.

B.2.4 Property View

The Property Summary View displays a tabular representation of a subset of the root project’s structure and content from the perspective of the properties in the project. The Property Summary View includes the following columns:

• Property: the names of the properties, grouped by alphabet.

• Project.Subproject(s).Alphabet: the name of the root project, the name(s) of all the subprojects that are parents of the alphabet associated with the property in a given row of the table, and the name of the alphabet that is associated with that property.

• Event: the names of the events in the alphabet that is associated with the property in the same row of the table.

• Behavior: the names of each of the behavior templates used in the properties that are in the project; each behavior template name shown is associated with the property that is in the same row(s), in the Property column to the behavior template’s left. If there is no behavior template chosen yet for a given property, the word “BLANK” is displayed in place of the name of the behavior template.

• Behavior Parameters: the parameters’ names (i.e., A or B), followed by a colon (“:”) and the name of the specifier-defined event associated with each parameter, if there is such an event. If there is no associated event, the word “NOT SET” is displayed in place of the event name.

• Scope: the name of each of the scope templates used in the properties in the project; each scope template shown is associated with the property that is in the same row(s), in the Property column to the scope template’s left.

• Scope Parameters: the parameters’ names (i.e., START or END), followed by a colon (“:”) and the name of the specifier-defined event associated with each parameter, if there is such an event. If there is no associated event, the word “NOT SET” is displayed in place of the event name.
• Property Comments: the comments associated with the property that is shown in the same row of the table.

An example of the Property Summary View is given in Figure B.5.

B.2.5 Event View

The Event Summary View displays a tabular representation of a subset of the root project’s structure and content from the perspective of the events in the project. The Event Summary View includes the following columns:

• Event: the names of the events, grouped by alphabet.

• Project.Subproject(s).Alphabet: the name of the root project, the name(s) of all the subproject(s) that are parents of the alphabet that is associated with the events in the same row of the table, and the name of the alphabet associated with those events.

• Property: the names of the properties that are associated with the alphabets that the events are in.

• Event Comments: the comments associated with the event that is shown in the same row of the table.

An example of the Event Summary View is given in Figure B.6.
<table>
<thead>
<tr>
<th>Project</th>
<th>Comments</th>
<th>Subproject</th>
<th>Comments</th>
<th>Subproject</th>
<th>Comments</th>
<th>Alphabet</th>
<th>Comments</th>
<th>Event</th>
<th>Comments</th>
<th>Property</th>
<th>Comments</th>
<th>Behavior</th>
<th>Behavior Parameters</th>
<th>Scope</th>
<th>Scope Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>Root Project</td>
<td>Root comments</td>
<td>S1</td>
<td>S1 comments</td>
<td>A1</td>
<td>A1 comments</td>
<td>E1</td>
<td>E1 comments</td>
<td>P1</td>
<td>P1 comments</td>
<td>PRECEDENCE</td>
<td>A, E1</td>
<td>GLOBAL</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>A2</td>
<td>A2 comments</td>
<td>E2</td>
<td>E2 comments</td>
<td>P2</td>
<td>P2 comments</td>
<td>ABSENCE</td>
<td>A, E3</td>
<td>BETWEEN</td>
<td>START: E4</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>A3</td>
<td>A3 comments</td>
<td>E3</td>
<td>E3 comments</td>
<td>P3</td>
<td>P3 comments</td>
<td>RESPONSE</td>
<td>A, E9</td>
<td>AFTER</td>
<td>E: NOT SET</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>A4</td>
<td>A4 comments</td>
<td>E4</td>
<td>E4 comments</td>
<td>P4</td>
<td>P4 comments</td>
<td>EXISTENCE</td>
<td>A, E11</td>
<td>BEFORE</td>
<td>IND: E9</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>A5</td>
<td>A5 comments</td>
<td>E5</td>
<td>E5 comments</td>
<td>P5</td>
<td>P5 comments</td>
<td>PRECEDENCE</td>
<td>A, E2</td>
<td>AFTER</td>
<td>START: E11</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>A6</td>
<td>A6 comments</td>
<td>E6</td>
<td>E6 comments</td>
<td>P6</td>
<td>P6 comments</td>
<td>ABSENCE</td>
<td>A, E12</td>
<td>GLOBAL</td>
<td>START: E13</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Figure B.2.** Full Summary View Example
<table>
<thead>
<tr>
<th>Project</th>
<th>Subproject</th>
<th>Subproject</th>
<th>Alphabet</th>
<th>Event</th>
<th>Property</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>Root Project</td>
<td>S1</td>
<td></td>
<td>A1</td>
<td>E1</td>
<td>P1</td>
<td>Root comments</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>E2</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>A2</td>
<td>E3</td>
<td>P2</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>E4</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>E5</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>E6</td>
<td></td>
<td></td>
</tr>
<tr>
<td>S2</td>
<td>S3</td>
<td></td>
<td>A3</td>
<td>E7</td>
<td>P3</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>E8</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>A4</td>
<td>E9</td>
<td>P4</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>E10</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>E11</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>A5</td>
<td>E12</td>
<td>P6</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>E13</td>
<td>P7</td>
<td></td>
</tr>
</tbody>
</table>

**Figure B.3.** Project Summary View Example

<table>
<thead>
<tr>
<th>Project,Subproject(s),Alphabet</th>
<th>Event</th>
<th>Property</th>
<th>Alphabet Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>Root Project,S1.A1</td>
<td>E1</td>
<td>P1</td>
<td>A1 comments</td>
</tr>
<tr>
<td></td>
<td>E2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Root Project,S1.A2</td>
<td>E3</td>
<td>P2</td>
<td>A2 comments</td>
</tr>
<tr>
<td></td>
<td>E4</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>E5</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>E6</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Root Project,S2.S3.A3</td>
<td>E7</td>
<td>P3</td>
<td>A3 comments</td>
</tr>
<tr>
<td></td>
<td>E8</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Root Project,S2.S3.A4</td>
<td>E9</td>
<td>P4</td>
<td>A4 comments</td>
</tr>
<tr>
<td></td>
<td>E10</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>E11</td>
<td>P5</td>
<td></td>
</tr>
<tr>
<td>Root Project,S2.S4.A5</td>
<td>E12</td>
<td>P6</td>
<td>A5 comments</td>
</tr>
<tr>
<td>Root Project,S2.S4.A6</td>
<td>E13</td>
<td>P7</td>
<td>A6 comments</td>
</tr>
</tbody>
</table>

**Figure B.4.** Alphabet Summary View Example
<table>
<thead>
<tr>
<th>Property</th>
<th>Project.Subproject(s).Alphabet</th>
<th>Event</th>
<th>Behavior</th>
<th>Behavior Parameters</th>
<th>Scope</th>
<th>Scope Parameters</th>
<th>Property Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>P1</td>
<td>Root Project.S1.A1</td>
<td>E1</td>
<td>PRECEDENCE</td>
<td>A: E1</td>
<td>GLOBAL</td>
<td></td>
<td>P1 comments</td>
</tr>
<tr>
<td></td>
<td></td>
<td>E2</td>
<td></td>
<td>B: E2</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>P2</td>
<td>Root Project.S1.A2</td>
<td>E3</td>
<td>ABSENCE</td>
<td>A: E3</td>
<td>BETWEEN</td>
<td>START: E4</td>
<td>P2 comments</td>
</tr>
<tr>
<td></td>
<td></td>
<td>E4</td>
<td></td>
<td></td>
<td></td>
<td>END: E5</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>E5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>E6</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>P3</td>
<td>Root Project.S2.S3.A3</td>
<td>E7</td>
<td>RESPONSE</td>
<td>A: E8</td>
<td>AFTER</td>
<td>START: E7</td>
<td>P3 comments</td>
</tr>
<tr>
<td></td>
<td></td>
<td>E8</td>
<td></td>
<td>B: NOT SET</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>E10</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>B: E10</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>END: NOT SET</td>
<td></td>
</tr>
</tbody>
</table>

**Figure B.5.** Property Summary View Example
<table>
<thead>
<tr>
<th>Event</th>
<th>Project.Subproject(s).Alphabet</th>
<th>Property</th>
<th>Event Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>E1</td>
<td>Root Project.S1.A1</td>
<td>P1</td>
<td>E1 comments</td>
</tr>
<tr>
<td>E2</td>
<td></td>
<td></td>
<td>E2 comments</td>
</tr>
<tr>
<td>E3</td>
<td>Root Project.S1.A2</td>
<td>P2</td>
<td>E3 comments</td>
</tr>
<tr>
<td>E4</td>
<td></td>
<td></td>
<td>E4 comments</td>
</tr>
<tr>
<td>E5</td>
<td></td>
<td></td>
<td>E5 comments</td>
</tr>
<tr>
<td>E6</td>
<td></td>
<td></td>
<td>E6 comments</td>
</tr>
<tr>
<td>E7</td>
<td>Root Project.S2.S3.A3</td>
<td>P3</td>
<td>E7 comments</td>
</tr>
<tr>
<td>E8</td>
<td></td>
<td></td>
<td>E8 comments</td>
</tr>
<tr>
<td>E9</td>
<td>Root Project.S2.S3.A4</td>
<td>P4</td>
<td>E9 comments</td>
</tr>
<tr>
<td>E10</td>
<td></td>
<td></td>
<td>E10 comments</td>
</tr>
<tr>
<td>E11</td>
<td></td>
<td>P5</td>
<td>E11 comments</td>
</tr>
<tr>
<td>E12</td>
<td>Root Project.S2.S4.A5</td>
<td>P6</td>
<td>E12 comments</td>
</tr>
<tr>
<td>E13</td>
<td>Root Project.S2.S4.A6</td>
<td>P7</td>
<td>E13 comments</td>
</tr>
</tbody>
</table>

**Figure B.6.** Event Summary View Example
APPENDIX C

CASE STUDY DATA
Medical errors are recognized as one of the major causes of death and other undesirable medical outcomes in the United States, and many of the medical errors that occur in practice are caused or exacerbated by faulty processes. One reason why such failures occur is that medical processes tend to be complex, involving a wide variety of resources and multiple medical professionals. These processes also tend to be prone to many exceptional behaviors, which existing safety policies either do not cover, or cover only partially. The medical domain is a rich source of complex, real-world requirements, and thus the five case studies that we used for our evaluation of PROPEL were all drawn from the medical domain.

A high-level description of these five case studies is given in Section 5.2, and Table 5.2 provides a summary that compares the five case studies in terms of the following dimensions:

- how we elicited the informal property specifications
- how many domain experts were involved
- who created the formal property specifications
- to what extent at least one domain expert was available to vet those informal and formal property specifications
- how many informal and formal property specifications were created for each case study

In this appendix, we briefly discuss the medical process area that each case study covers and we give a slightly more detailed account of how the above dimensions map to each case study.
C.1 Blood Transfusion

C.1.1 Description of the Domain

The Blood Transfusion (BT) case study is part of a larger medical safety project that our research group is participating in. In this case study, the domain expert was a Nursing Ph.D. We elicited most of the properties through interviews with her, and we referred to Nursing textbooks and articles to obtain the remaining information, which she then helped us fit into the case study as appropriate. Although we, the computer scientists, were primarily the ones who worked with PROPEL to formally specify the properties, the domain expert used the DNL template property view to specify 5-6 of them herself, with us observing in those cases. In addition, she worked with us to vet all of the informal and formal property specifications against her domain expertise. This domain expert also worked with us on the VPID case study (see Appendix C.4) and she did not have any prior knowledge of property specification formalisms or RE.

Due to resource limitations, we focused on a subset of the BT domain, specifically covering only in-patient clinical situations. Thus, this study does not cover laboratory situations (see Appendix C.5 for partial coverage of that area) or Emergency Department situations (see Appendix C.3 for partial coverage of that area). In addition, all properties in this case study are assumed to be in the context of one patient and one medical professional (except where explicitly noted that two medical professionals must be present), and carrying out only one physician order for blood transfusion (although there could be multiple units of blood product transfused in a single order).

There are 44 informal property specifications in this case study, and they are given in Appendix C.1.2. See Section 5.4.3.2 for an explanation of the notation used in the informal specifications. Appendix C.1.3 gives the glossary for all of the bolded terms used in the informal specifications; see Section 5.4.3.1 for an explanation of the glossary structure. For the 42 informal BT property specifications that can be captured in PROPEL, there are 75 formal BT property specifications, and they are given in Appendix C.1.4. The FSA and DNL property specifications are given for each property, with some modifications made to these two views to enable the property specifications to fit on a single page. The FSA specification is shown in two parts: at the top is a mapping from parameter names to specifier-specified events, labeled “Event alphabet:”, and directly below that mapping is an FSA specification of the property. The FSA is assumed to be total and deterministic\(^1\) and its transitions are labeled with the parameters, rather than with the specifier-specified events. The DNL specification is also shown in two parts, the scope and the behavior, but a third part, the

\(^1\)For brevity, we do not show the transitions that go to a non-accepting trap state. When no transition is provided that explicitly allows an event to occur, it should be assumed that an occurrence of that event puts the FSA into a non-accepting trap state.
preamble, is elided. The preamble is the same for every DNL property specification, and can be found in Section 3.2.4.
C.1.2 Organization of the Informal Property Specifications

Blood Transfusion Properties

A. CHECKS THAT ARE DONE BEFORE OBTAINING UNIT(S) OF BLOOD PRODUCT

MP A.1 If a physician has ordered a blood transfusion for a Patient but that Patient's type and screen is not available in the blood bank, a blood specimen must be obtained from that Patient.

MP A.2 A Patient must be assessed for appropriate I.V. access before the unit(s) of blood product are allowed to be picked up from the blood bank.

B. ADMINISTRATION OF BLOOD PRODUCT

Before performing a blood transfusion for a Patient, the following activities must be performed:

MP B.1 the presence of that Patient's signed consent document must be confirmed.

MP B.2 that Patient's history pre-assessment must be reviewed.

MP B.3 the activities described above are allowed to occur in either order.

Note: “perform blood transfusion” refers to the part of the blood transfusion process that begins directly after the nurse picks up the units of blood product from the blood bank.

Before infusing each unit of blood product into a Patient, the following activities must be performed by two different nurses independently:

-- B.4 that Patient's ID band must be verified:
   All of this Patient identifying information must be verified by two separate nurses for B.4 to be satisfied. The cases enumerate what information for the ID band must be verified.

MP B.4.1 it must be confirmed that that Patient has exactly one ID band.

MP B.4.2 that Patient's stated first and last name and birth date must be obtained.

MP B.4.3 it must be made sure that the first and last name and birth date on that Patient's ID band match that Patient's stated first and last name and birth date.

MP B.4.4 it must be made sure that the first and last name and medical record number on that Patient's ID band match the first and last name and medical record number on the physician order for a blood transfusion.

MP B.4.5 these verifications described above must occur in that order.

-- B.5 the information for the unit of blood product must be verified:
   All of this blood product information must be verified by two separate nurses for B.5 to be satisfied. The cases enumerate what information for the unit of blood product must be verified.

MP B.5.1 it must be made sure that the first and last name and medical record number on that Patient's ID band match the first and last name and medical record number on the tag affixed to the unit of blood product.

MP B.5.2 it must be made sure that the blood type and blood product unit number on the tag affixed to the unit of blood product match the blood type and blood product unit number on the unit of blood product.

MP B.5.3 it must be made sure that the unit of blood product has not expired.

MP B.5.4 these verifications described above must occur in that order.

MP B.6 a baseline single-unit assessment must be done for that Patient to make sure that that Patient is well enough to receive an infusion.

MP B.7 these activities described above must occur in that order.

MP B.8 Before infusing a unit of blood product, if a Patient's baseline single-unit assessment indicates that that Patient is possibly not well enough to receive an infusion, the physician must be asked for further instructions.

MP B.9 The infusion of a unit of blood product must begin within 30 minutes of the unit of blood product being picked up from the blood bank.

MP B.10 An expired unit of blood product should not be infused into a Patient.
## Blood Transfusion Properties

<table>
<thead>
<tr>
<th>MP</th>
<th>B.10</th>
<th>An expired unit of blood product should not be infused into a Patient.</th>
</tr>
</thead>
<tbody>
<tr>
<td>MP</td>
<td>B.11</td>
<td>A 15-minute single-unit assessment must be done for a Patient after the first 15 minutes of that Patient’s infusion has passed.</td>
</tr>
<tr>
<td>MP</td>
<td>B.12</td>
<td>A post-single-unit assessment must be done for a Patient immediately after infusing each unit of blood product into that Patient. Every event in the transfusion process is prohibited from occurring between completing an infusion of a unit of blood product and performing a post-single-unit assessment.</td>
</tr>
<tr>
<td>X</td>
<td>B.13</td>
<td>If a Patient’s infusion is ongoing and other medications must be administered to that Patient and appropriate I.V. access for that Patient is not available, that Patient’s infusion must be temporarily stopped to allow the other medications to be administered through the infusion I.V.</td>
</tr>
<tr>
<td>MP</td>
<td>B.14</td>
<td>If it is suspected that a Patient’s infusion will exceed the infusion time limit or if it is discovered that that Patient’s infusion did exceed the infusion time limit, the blood bank must be asked for further instructions.</td>
</tr>
<tr>
<td>MP</td>
<td>B.15</td>
<td>If the blood bank or physician instructs that a Patient’s infusion be resumed, that Patient’s infusion must be resumed.</td>
</tr>
<tr>
<td>MP</td>
<td>B.16</td>
<td>If a Patient’s infusion has been stopped (e.g., for a suspected transfusion reaction), it must eventually be discontinued, unless the blood bank or the physician instructs that that Patient’s infusion be resumed.</td>
</tr>
</tbody>
</table>

**Figure C.2. Blood Transfusion Case Study Informal Specifications - Page 2**
Blood Transfusion Properties

C. THE HANDLING OF A SUSPECTED TRANSFUSION REACTION

If it is suspected that a Patient is having a transfusion reaction to an infusion:

MP C.1 • that Patient’s infusion must be stopped immediately.

Every event in the transfusion process is prohibited from occurring between suspecting a transfusion reaction and stopping the infusion.

MP C.2 • that Patient’s ID band must be verified:

   All of this Patient identifying information must be verified by at least one nurse for C.2 to be satisfied. The cases enumerate what information for the ID band must be verified.

MP C.2.1 • it must be confirmed that that Patient has at least one ID band.

MP C.2.2 • that Patient’s stated first and last name and birth date must be obtained.

MP C.2.3 • it must be made sure that the first and last name and birth date on the Patient’s ID band match that Patient’s stated first and last name and birth date.

MP C.2.4 • these verifications described above must occur in that order.

MP C.3 • the following information for the unit of blood product must be verified:

   All of this blood product information must be verified by at least one nurse for C.3 to be satisfied. The cases enumerate what information for the unit of blood product must be verified.

MP C.3.1 • it must be made sure that the first and last name and medical record number on that Patient’s ID band match the first and last name and medical record number on the tag affixed to the unit of blood product.

MP C.3.2 • it must be made sure that the blood type and blood product unit number on the tag affixed to the unit of blood product match the blood type and blood product unit number on the unit of blood product.

MP C.3.3 • it must be made sure that the unit of blood product has not expired.

MP C.3.4 • these verifications described above must occur in that order.

This ordering is exactly the same as in B.5.4.

MP C.4 • the physician and the blood bank must be asked for further instructions.

MP C.5 • these activities described above must occur in that order.

MP C.6 • If a Patient’s infusion has been stopped due to a suspected transfusion reaction and the blood bank orders that specimens be obtained from that Patient, those specimens must be obtained from that Patient.

Figure C.3. Blood Transfusion Case Study Informal Specifications - Page 3
Blood Transfusion Properties

C. THE HANDLING OF A SUSPECTED TRANSFUSION REACTION

MP C.6 If a Patient’s infusion has been stopped due to a suspected transfusion reaction and the blood bank orders that specimens be obtained from that Patient, those specimens must be obtained from that Patient.

D. CHECKS THAT ARE DONE BEFORE OBTAINING SPECIMEN(S) FROM A PATIENT

These checks are generalized for any specimens that might be obtained from a Patient, and certain specializations are possible. For example, the generalized event “receive a blood bank or physician order for a type and screen” or “receive a blood bank order to obtain a post-suspected-transfusion-reaction urine specimen”. The orders specify exactly which type(s) of specimen must be obtained from the Patient. In all these cases, however, the same checks must occur in the same order.

After receiving a blood bank or physician order to obtain a specimen from a Patient and immediately before obtaining any specimen from that Patient or applying any specimen container labels, the following activities must be performed:

To summarize: that Patient’s ID band must be verified by performing the following activities.

Every event in the transfusion process (except for applying a label to a specimen container) is prohibited from occurring between performing the following activities and obtaining the specimen, and vice versa.

MP D.1 • it must be confirmed that that Patient has exactly one ID band.
MP D.2 • that Patient’s stated first and last name and birth date must be obtained.
MP D.3 • it must be made sure that the first and last name and birth date on that Patient’s ID band match that Patient’s stated first and last name and birth date.
MP D.4 • it must be made sure that the first and last name and medical record number on that Patient’s ID band match the first and last name and medical record number on the order to obtain a specimen.
MP D.5 • it must be made sure that the first and last name and medical record number on that Patient’s ID band match the first and last name and medical record number on the specimen container label.
MP D.6 • the activities described above must occur in that order.

X D.7 If any event occurs between obtaining a specimen from a Patient and applying the specimen container label, that specimen container cannot be sent to the blood bank; that specimen container must be discarded.

Obtaining a specimen from a Patient and applying the specimen container label are allowed to occur in either order.

Figure C.4. Blood Transfusion Case Study Informal Specifications - Page 4

263
### C.1.3 Glossary

**Blood Transfusion Property Glossary**

<table>
<thead>
<tr>
<th>Term</th>
<th>Definition</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>appropriate I.V. access</td>
<td>I.V. access other than the infusion I.V.</td>
<td></td>
</tr>
<tr>
<td>be available</td>
<td></td>
<td></td>
</tr>
<tr>
<td>blood bank</td>
<td>syn. ‘blood repository’</td>
<td></td>
</tr>
<tr>
<td>ask for instructions</td>
<td></td>
<td></td>
</tr>
<tr>
<td>instruct that an infusion be</td>
<td>discontinued</td>
<td></td>
</tr>
<tr>
<td>not have type and screen</td>
<td></td>
<td></td>
</tr>
<tr>
<td>order that specimens be</td>
<td>obtained</td>
<td>Includes ordering a type and screen.</td>
</tr>
<tr>
<td>pick unit(s) of blood product</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**blood product information**

<table>
<thead>
<tr>
<th>Term</th>
<th>Definition</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>verify</td>
<td>Check to make sure that a Patient’s first and last name and medical record number on the tag affixed to the unit of blood product matches that Patient's first and last name and medical record number on that Patient ID band. Also check to make sure that the blood type and blood product unit number on the tag affixed to the unit of blood product match the blood type and blood product unit number on the unit of blood product. Finally, check to make sure that the expiration date on the unit of blood product has not been exceeded.</td>
<td>syn. ‘information for the unit of blood product’</td>
</tr>
</tbody>
</table>

**blood product unit number**

<table>
<thead>
<tr>
<th>Term</th>
<th>Definition</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>make sure match between tag</td>
<td></td>
<td></td>
</tr>
<tr>
<td>affixed to unit of blood</td>
<td></td>
<td></td>
</tr>
<tr>
<td>product</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**blood product, unit of blood**

<table>
<thead>
<tr>
<th>Term</th>
<th>Definition</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>administer</td>
<td></td>
<td></td>
</tr>
<tr>
<td>be expired</td>
<td></td>
<td>This is not the 30-minute time limit on beginning to infuse the unit of blood product after picking it up from the blood bank. This is the expiration date.</td>
</tr>
<tr>
<td>make sure blood product unit</td>
<td></td>
<td></td>
</tr>
<tr>
<td>number match with tag</td>
<td></td>
<td></td>
</tr>
<tr>
<td>affixed to unit of blood</td>
<td></td>
<td></td>
</tr>
<tr>
<td>product</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Figure C.5. Blood Transfusion Case Study Glossary - Page 1
### Blood Transfusion Property Glossary

<table>
<thead>
<tr>
<th>Term</th>
<th>Definition</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID band</td>
<td>have Patient name, birth date, and medical record number match with order</td>
<td>Applies for both the blood transfusion orders and the orders to obtain a specimen.</td>
</tr>
<tr>
<td>make sure name and birth date match with Patient stated name and birth date</td>
<td></td>
<td></td>
</tr>
<tr>
<td>infusion</td>
<td>15 minutes pass</td>
<td></td>
</tr>
<tr>
<td>begin</td>
<td>The infusion must begin within 30 minutes of the unit of blood product being picked up from the blood bank.</td>
<td></td>
</tr>
<tr>
<td>discontinue</td>
<td>After stopping, never restart the infusion.</td>
<td>This is a big source of trouble in practice: often stop-and-discontinue is done when stop-and-resume might be what is needed.</td>
</tr>
<tr>
<td>resume</td>
<td>After stopping, restart the infusion.</td>
<td></td>
</tr>
<tr>
<td>stop</td>
<td>Halt (possibly only temporarily) the flow of blood from the unit of blood product into the Patient's body. syn. 'pause' and 'interrupt'. Must be followed by either resuming the infusion or discontinuing the infusion. Note that this is not the same as &quot;completing&quot; the infusion, which implies a successful infusion of an entire unit of blood product.</td>
<td></td>
</tr>
<tr>
<td>suspect it will exceed infusion time limit</td>
<td></td>
<td></td>
</tr>
<tr>
<td>infusion I.V.</td>
<td>The I.V. through which the blood transfusion is infused.</td>
<td></td>
</tr>
<tr>
<td>infusion time limit</td>
<td>How long a single unit of blood product can be infused into a Patient. For red blood cells: 4 hrs.; for platelets: 30 mins.</td>
<td></td>
</tr>
<tr>
<td>medical record number</td>
<td></td>
<td></td>
</tr>
<tr>
<td>make sure first and last name and medical record number match with order</td>
<td></td>
<td></td>
</tr>
<tr>
<td>make sure first and last name and medical record number match with specimen container label</td>
<td></td>
<td></td>
</tr>
<tr>
<td>make sure first and last name and medical record number match with tag affixed to unit of blood product</td>
<td></td>
<td></td>
</tr>
<tr>
<td>medical record number</td>
<td></td>
<td></td>
</tr>
<tr>
<td>make sure first and last name and medical record number match with ID band</td>
<td></td>
<td></td>
</tr>
<tr>
<td>nurse</td>
<td>The implied agent doing all the activities in the blood transfusion process.</td>
<td></td>
</tr>
<tr>
<td>order</td>
<td>Could be an order for a blood transfusion (from a physician) or an order to obtain a specimen from a Patient, such as an order for a type and screen (from the blood bank or a physician).</td>
<td></td>
</tr>
</tbody>
</table>

**Figure C.6.** Blood Transfusion Case Study Glossary - Page 2
## Blood Transfusion Property Glossary

<table>
<thead>
<tr>
<th>Term</th>
<th>Definition</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>order</td>
<td>Could be an order for a blood transfusion (from a physician) or an order to obtain a specimen from a patient, such as an order for a type and screen (from the blood bank or a physician).</td>
<td>Applies only for the orders to obtain a specimen.</td>
</tr>
<tr>
<td>other medications</td>
<td>administer</td>
<td></td>
</tr>
<tr>
<td>Patient</td>
<td>assess for appropriate I.V. access</td>
<td></td>
</tr>
<tr>
<td></td>
<td>be well enough to receive infusion</td>
<td></td>
</tr>
<tr>
<td></td>
<td>have transfusion reaction</td>
<td></td>
</tr>
<tr>
<td></td>
<td>receive infusion</td>
<td></td>
</tr>
<tr>
<td>Patient first and last name and birth date</td>
<td>make sure match between first and last name and birth date on ID band and Patient stated name and birth date</td>
<td></td>
</tr>
<tr>
<td>Patient history pre-assessment</td>
<td>find unknown prior history of adverse reactions in &quot;Unknown&quot; here refers to there being no indication in the Patient’s record that the physician knows about the prior history of adverse reactions.</td>
<td></td>
</tr>
<tr>
<td>review</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Patient ID band</td>
<td>Make sure that the relevant letters and numbers on the Patient’s ID band are exactly the same as the relevant letters and numbers on the artifact that the ID band is being compared to.</td>
<td>syn. &quot;Patient armband&quot;, &quot;Patient wristband&quot;, &quot;Patient ID bracelet&quot;</td>
</tr>
<tr>
<td>verify</td>
<td>Check to make sure that the first and last name and birth date on the Patient’s ID band match that Patient’s stated first and last name and birth date. Also check to make sure that the first and last name and medical record number on that Patient’s ID band match the first and last name and medical record number on the (tag affixed to the unit of blood product / order / specimen container label).</td>
<td></td>
</tr>
<tr>
<td>Patient signed consent document</td>
<td>The Patient’s signed informed consent document. One consent document is sufficient from admission to discharge.</td>
<td></td>
</tr>
<tr>
<td>Patient status</td>
<td>Determine the Patient’s vital signs: blood pressure, pulse, temperature, and breathing rate.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>15-minute single-unit</td>
<td>Patient assessment done immediately after 15 minutes of the infusion for a single unit of blood product has been completed. This could lead to the suspicion of a transfusion reaction.</td>
</tr>
<tr>
<td></td>
<td>baseline single-unit</td>
<td>Patient assessment done before starting to infuse a single unit of blood product, to establish the Patient’s baseline status and to declare intent to proceed with the infusion (barring physician instructions not to proceed).</td>
</tr>
<tr>
<td></td>
<td>post-single-unit</td>
<td>Patient assessment done after a single unit of blood product has been completely infused. If the infusion is stopped earlier because of a suspected transfusion reaction, this post-single-unit assessment will not be done. The infusion may complete and this post-single-unit assessment may be done and it may lead to the suspicion of a transfusion reaction after that point.</td>
</tr>
<tr>
<td>Term</td>
<td>Definition</td>
<td>Notes</td>
</tr>
<tr>
<td>--------------------------</td>
<td>---------------------------------------------------------------------------</td>
<td>----------------------------------------------------------------------</td>
</tr>
<tr>
<td>physician</td>
<td>be asked for further instructions</td>
<td>In the event of a suspected transfusion reaction.</td>
</tr>
<tr>
<td></td>
<td>instruct that an infusion be discontinued</td>
<td></td>
</tr>
<tr>
<td></td>
<td>order blood transfusion</td>
<td></td>
</tr>
<tr>
<td></td>
<td>order that specimens be obtained</td>
<td>Includes ordering a type and screen.</td>
</tr>
<tr>
<td>specimen</td>
<td></td>
<td>Includes blood, urine, stool, etc. specimens.</td>
</tr>
<tr>
<td></td>
<td>obtain</td>
<td></td>
</tr>
<tr>
<td>specimen container</td>
<td></td>
<td></td>
</tr>
<tr>
<td>specimen container label</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>apply</td>
<td></td>
</tr>
<tr>
<td></td>
<td>make sure first and last name and medical record number match with ID band</td>
<td></td>
</tr>
<tr>
<td></td>
<td>make sure first and last name and medical record number match with order to obtain a specimen</td>
<td></td>
</tr>
<tr>
<td>tag affixed to a unit of blood product</td>
<td>make sure blood product unit number match with unit of blood product</td>
<td></td>
</tr>
<tr>
<td></td>
<td>make sure blood type match with unit of blood product</td>
<td></td>
</tr>
<tr>
<td></td>
<td>make sure medical record number match with ID band</td>
<td></td>
</tr>
<tr>
<td>transfusion reaction</td>
<td>suspect Patient is having</td>
<td></td>
</tr>
<tr>
<td>type and screen</td>
<td></td>
<td>be available in the blood bank</td>
</tr>
</tbody>
</table>

Figure C.8. Blood Transfusion Case Study Glossary - Page 4
C.1.4 Formal Property Specifications

Event alphabet:
- A: discover blood bank doesn’t have type and screen
- B: obtain blood specimen
- START: receive physician order for blood transfusion

SCOPE:
1. There can be at most one restricted interval in the event sequence and it has a starting delimiter, receive physician order for blood transfusion.
2. The behavior is required to hold from the first occurrence of receive physician order for blood transfusion through to the end of the event sequence. Even if receive physician order for blood transfusion occurs more than once before the end of the event sequence, only the first occurrence of receive physician order for blood transfusion begins the restricted interval; later occurrences of receive physician order for blood transfusion do not have an effect.
3. receive physician order for blood transfusion is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence.
4. There are no restrictions imposed on the occurrences of any events before the first occurrence of receive physician order for blood transfusion, if it ever occurs.

BEHAVIOR:
1. The events of primary interest in this behavior are discover blood bank doesn’t have type and screen and obtain blood specimen.
2. There are no events of secondary interest in this behavior.
3. If discover blood bank doesn’t have type and screen occurs, obtain blood specimen is required to occur subsequently.
4. Before the first discover blood bank doesn’t have type and screen occurs, obtain blood specimen is allowed to occur zero or more times.
5. discover blood bank doesn’t have type and screen is not required to occur.
6. After discover blood bank doesn’t have type and screen occurs, but before the first subsequent obtain blood specimen occurs, discover blood bank doesn’t have type and screen is allowed to occur again, zero or more times.
7. After discover blood bank doesn’t have type and screen and the first subsequent obtain blood specimen occur:
   - Both discover blood bank doesn’t have type and screen and obtain blood specimen are allowed to occur again, zero or more times, and there are no restrictions on the occurrences of any future events.

FSA and DNL for Property BT.A.1
Event alphabet:

- A: assess Patient for appropriate I.V. access
- B: pick up blood product

SCOPE:
1. From the start of any event sequence through to the end of that event sequence, the behavior must hold.

BEHAVIOR:
1. The events of primary interest in this behavior are assess Patient for appropriate I.V. access and pick up blood product.
2. There are no events of secondary interest in this behavior.
3. pick up blood product is not allowed to occur until after assess Patient for appropriate I.V. access occurs.
4. assess Patient for appropriate I.V. access is required to occur, whether or not pick up blood product eventually occurs.
5. pick up blood product is not required to occur after assess Patient for appropriate I.V. access occurs.
6. After assess Patient for appropriate I.V. access occurs, but before the first subsequent pick up blood product occurs, assess Patient for appropriate I.V. access is allowed to occur again, zero or more times.
7. After assess Patient for appropriate I.V. access and the first subsequent pick up blood product occur:
   - Both assess Patient for appropriate I.V. access and pick up blood product are allowed to occur again, zero or more times, and there are no restrictions on the occurrences of any future events.

FSA and DNL for Property BT.A.2
Event alphabet:
- A: nurse 1 confirms presence of ID band
- B: infuse a unit of blood product
- C: nurse 2 confirms presence of ID band

**SCOPE:**
1. From the start of any event sequence through to the end of that event sequence, the behavior must hold.

**BEHAVIOR:**
1. The events of primary interest in this behavior are nurse 1 confirms presence of ID band and infuse a unit of blood product.
2. The event of secondary interest in this behavior is nurse 2 confirms presence of ID band.
3. infuse a unit of blood product is not allowed to occur until after nurse 1 confirms presence of ID band occurs.
4. Before the first nurse 1 confirms presence of ID band occurs, nurse 2 confirms presence of ID band is allowed to occur zero or more times.
5. nurse 1 confirms presence of ID band is not required to occur.
6. Even if nurse 1 confirms presence of ID band does occur, infuse a unit of blood product is not required to occur after nurse 1 confirms presence of ID band occurs.
7. After nurse 1 confirms presence of ID band occurs, but before the first subsequent infuse a unit of blood product occurs:
   - nurse 1 confirms presence of ID band is allowed to occur again, zero or more times;
   - nurse 2 confirms presence of ID band is allowed to occur zero or more times.
8. After nurse 1 confirms presence of ID band and the first subsequent infuse a unit of blood product occur:
   - nurse 2 confirms presence of ID band is allowed to occur zero or more times;
   - infuse a unit of blood product is not allowed to occur again until after another nurse 1 confirms presence of ID band occurs; nurse 1 confirms presence of ID band is allowed to occur again and, if it does, then the situation is the same as when the first nurse 1 confirms presence of ID band occurred, meaning that the restrictions described in parts 6, 7, and 8 would again apply.

FSA and DNL for Property BT.B.4.1a
Event alphabet:

- A: nurse 2 confirms presence of ID band
- B: infuse a unit of blood product
- C: nurse 1 confirms presence of ID band

SCOPE:
1. From the start of any event sequence through to the end of that event sequence, the behavior must hold.

BEHAVIOR:
1. The events of primary interest in this behavior are nurse 2 confirms presence of ID band and infuse a unit of blood product.
2. The event of secondary interest in this behavior is nurse 1 confirms presence of ID band.
3. infuse a unit of blood product is not allowed to occur until after nurse 2 confirms presence of ID band occurs.
4. Before the first nurse 2 confirms presence of ID band occurs, nurse 1 confirms presence of ID band is allowed to occur zero or more times.
5. nurse 2 confirms presence of ID band is not required to occur.
6. Even if nurse 2 confirms presence of ID band does occur, infuse a unit of blood product is not required to occur after nurse 2 confirms presence of ID band occurs.
7. After nurse 2 confirms presence of ID band occurs, but before the first subsequent infuse a unit of blood product occurs:
   - nurse 2 confirms presence of ID band is allowed to occur again, zero or more times;
   - nurse 1 confirms presence of ID band is allowed to occur zero or more times.
8. After nurse 2 confirms presence of ID band and the first subsequent infuse a unit of blood product occur:
   - nurse 1 confirms presence of ID band is allowed to occur zero or more times;
   - infuse a unit of blood product is not allowed to occur again until after another nurse 2 confirms presence of ID band occurs; nurse 2 confirms presence of ID band is allowed to occur again and, if it does, then the situation is the same as when the first nurse 2 confirms presence of ID band occurred, meaning that the restrictions described in parts 6, 7, and 8 would again apply.

FSA and DNL for Property BT.B.4.1b
**Event alphabet:**
- **A:** nurse 1 obtains Patient’s stated name and birth date
- **B:** infuse a unit of blood product
- **C:** nurse 2 obtains Patient’s stated name and birth date

**SCOPE:**
1. From the start of any event sequence through to the end of that event sequence, the behavior must hold.

**BEHAVIOR:**
1. The events of primary interest in this behavior are nurse 1 obtains Patient’s stated name and birth date and infuse a unit of blood product.
2. The event of secondary interest in this behavior is nurse 2 obtains Patient’s stated name and birth date.
3. infuse a unit of blood product is not allowed to occur until after nurse 1 obtains Patient’s stated name and birth date occurs.
4. Before the first nurse 1 obtains Patient’s stated name and birth date occurs, nurse 2 obtains Patient’s stated name and birth date is allowed to occur zero or more times.
5. nurse 1 obtains Patient’s stated name and birth date is not required to occur.
6. Even if nurse 1 obtains Patient’s stated name and birth date does occur, infuse a unit of blood product is not required to occur after nurse 1 obtains Patient’s stated name and birth date occurs.
7. After nurse 1 obtains Patient’s stated name and birth date occurs, but before the first subsequent infuse a unit of blood product occurs:
   - nurse 1 obtains Patient’s stated name and birth date is allowed to occur again, zero or more times;
   - nurse 2 obtains Patient’s stated name and birth date is allowed to occur zero or more times.
8. After nurse 1 obtains Patient’s stated name and birth date and the first subsequent infuse a unit of blood product occur:
   - nurse 2 obtains Patient’s stated name and birth date is allowed to occur zero or more times;
   - infuse a unit of blood product is not allowed to occur again until after another nurse 1 obtains Patient’s stated name and birth date occurs; nurse 1 obtains Patient’s stated name and birth date is allowed to occur again and, if it does, then the situation is the same as when the first nurse 1 obtains Patient’s stated name and birth date occurred, meaning that the restrictions described in parts 6, 7, and 8 would again apply.

FSA and DNL for Property BT.B.4.2a
Event alphabet:
- A: nurse 2 obtains Patient’s stated name and birth date
- B: infuse a unit of blood product
- C: nurse 1 obtains Patient’s stated name and birth date

SCOPE:
1. From the start of any event sequence through to the end of that event sequence, the behavior must hold.

BEHAVIOR:
1. The events of primary interest in this behavior are nurse 2 obtains Patient’s stated name and birth date and infuse a unit of blood product.
2. The event of secondary interest in this behavior is nurse 1 obtains Patient’s stated name and birth date.
3. infuse a unit of blood product is not allowed to occur until after nurse 2 obtains Patient’s stated name and birth date occurs.
4. Before the first nurse 2 obtains Patient’s stated name and birth date occurs, nurse 1 obtains Patient’s stated name and birth date is allowed to occur zero or more times.
5. nurse 2 obtains Patient’s stated name and birth date is not required to occur.
6. Even if nurse 2 obtains Patient’s stated name and birth date does occur, infuse a unit of blood product is not required to occur after nurse 2 obtains Patient’s stated name and birth date occurs.
7. After nurse 2 obtains Patient’s stated name and birth date occurs, but before the first subsequent infuse a unit of blood product occurs:
   - nurse 2 obtains Patient’s stated name and birth date is allowed to occur again, zero or more times;
   - nurse 1 obtains Patient’s stated name and birth date is allowed to occur zero or more times.
8. After nurse 2 obtains Patient’s stated name and birth date and the first subsequent infuse a unit of blood product occur:
   - nurse 1 obtains Patient’s stated name and birth date is allowed to occur zero or more times;
   - infuse a unit of blood product is not allowed to occur again until after another nurse 2 obtains Patient’s stated name and birth date occurs; nurse 2 obtains Patient’s stated name and birth date is allowed to occur again and, if it does, then the situation is the same as when the first nurse 2 obtains Patient’s stated name and birth date occurred, meaning that the restrictions described in parts 6, 7, and 8 would again apply.

FSA and DNL for Property BT.B.4.2b
Event alphabet:

- A: nurse 1 makes sure that ID band and Patient’s stated name and birth date match
- B: infuse a unit of blood product
- C: nurse 2 makes sure that ID band and Patient’s stated name and birth date match

SCOPE:
1. From the start of any event sequence through to the end of that event sequence, the behavior must hold.

BEHAVIOR:
1. The events of primary interest in this behavior are nurse 1 makes sure that ID band and Patient’s stated name and birth date match and infuse a unit of blood product.
2. The event of secondary interest in this behavior is nurse 2 makes sure that ID band and Patient’s stated name and birth date match.
3. infuse a unit of blood product is not allowed to occur until after nurse 1 makes sure that ID band and Patient’s stated name and birth date match occurs.
4. Before the first nurse 1 makes sure that ID band and Patient’s stated name and birth date match occurs, nurse 2 makes sure that ID band and Patient’s stated name and birth date match is allowed to occur zero or more times.
5. nurse 1 makes sure that ID band and Patient’s stated name and birth date match is not required to occur.
6. Even if nurse 1 makes sure that ID band and Patient’s stated name and birth date match does occur, infuse a unit of blood product is not required to occur after nurse 1 makes sure that ID band and Patient’s stated name and birth date match occurs.
7. After nurse 1 makes sure that ID band and Patient’s stated name and birth date match occurs, but before the first subsequent infuse a unit of blood product occurs:
   - nurse 1 makes sure that ID band and Patient’s stated name and birth date match is allowed to occur again, zero or more times;
   - nurse 2 makes sure that ID band and Patient’s stated name and birth date match is allowed to occur zero or more times.
8. After nurse 1 makes sure that ID band and Patient’s stated name and birth date match and the first subsequent infuse a unit of blood product occur:
   - nurse 2 makes sure that ID band and Patient’s stated name and birth date match is allowed to occur zero or more times;
   - infuse a unit of blood product is not allowed to occur again until after another nurse 1 makes sure that ID band and Patient’s stated name and birth date match occurs; nurse 1 makes sure that ID band and Patient’s stated name and birth date match is allowed to occur again and, if it does, then the situation is the same as when the first nurse 1 makes sure that ID band and Patient’s stated name and birth date match occurred, meaning that the restrictions described in parts 6, 7, and 8 would again apply.

FSA and DNL for Property BT.B.4.3a
Event alphabet:
• A: nurse 2 makes sure that ID band and Patient’s stated name and birth date match
• B: infuse a unit of blood product
• C: nurse 1 makes sure that ID band and Patient’s stated name and birth date match

SCOPE:
1. From the start of any event sequence through to the end of that event sequence, the behavior must hold.

BEHAVIOR:
1. The events of primary interest in this behavior are nurse 2 makes sure that ID band and Patient’s stated name and birth date match and infuse a unit of blood product.
2. The event of secondary interest in this behavior is nurse 1 makes sure that ID band and Patient’s stated name and birth date match.
3. infuse a unit of blood product is not allowed to occur until after nurse 2 makes sure that ID band and Patient’s stated name and birth date match occurs.
4. Before the first nurse 2 makes sure that ID band and Patient’s stated name and birth date match occurs, nurse 1 makes sure that ID band and Patient’s stated name and birth date match is allowed to occur zero or more times.
5. nurse 2 makes sure that ID band and Patient’s stated name and birth date match is not required to occur.
6. Even if nurse 2 makes sure that ID band and Patient’s stated name and birth date match does occur, infuse a unit of blood product is not required to occur after nurse 2 makes sure that ID band and Patient’s stated name and birth date match occurs.
7. After nurse 2 makes sure that ID band and Patient’s stated name and birth date match occurs, but before the first subsequent infuse a unit of blood product occurs:
   • nurse 2 makes sure that ID band and Patient’s stated name and birth date match is allowed to occur again, zero or more times;
   • nurse 1 makes sure that ID band and Patient’s stated name and birth date match is allowed to occur zero or more times.
8. After nurse 2 makes sure that ID band and Patient’s stated name and birth date match occurs, but before the first subsequent infuse a unit of blood product occur:
   • nurse 1 makes sure that ID band and Patient’s stated name and birth date match is allowed to occur zero or more times;
   • infuse a unit of blood product is not allowed to occur again until after another nurse 2 makes sure that ID band and Patient’s stated name and birth date match occurs; nurse 2 makes sure that ID band and Patient’s stated name and birth date match is allowed to occur again and, if it does, then the situation is the same as when the first nurse 2 makes sure that ID band and Patient’s stated name and birth date match occurred, meaning that the restrictions described in parts 6, 7, and 8 would again apply.

FSA and DNL for Property BT.B.4.3b
Event alphabet:
• A: nurse 1 makes sure that ID band and physician order for blood transfusion match
• B: infuse a unit of blood product
• C: nurse 2 makes sure that ID band and physician order for blood transfusion match

SCOPE:
1. From the start of any event sequence through to the end of that event sequence, the behavior must hold.

BEHAVIOR:
1. The events of primary interest in this behavior are nurse 1 makes sure that ID band and physician order for blood transfusion match and infuse a unit of blood product.
2. The event of secondary interest in this behavior is nurse 2 makes sure that ID band and physician order for blood transfusion match.
3. Infuse a unit of blood product is not allowed to occur until after nurse 1 makes sure that ID band and physician order for blood transfusion match occurs.
4. Before the first nurse 1 makes sure that ID band and physician order for blood transfusion match occurs, nurse 2 makes sure that ID band and physician order for blood transfusion match is allowed to occur zero or more times.
5. Nurse 1 makes sure that ID band and physician order for blood transfusion match is not required to occur.
6. Even if nurse 1 makes sure that ID band and physician order for blood transfusion match does occur, infuse a unit of blood product is not required to occur after nurse 1 makes sure that ID band and physician order for blood transfusion match occurs.
7. After nurse 1 makes sure that ID band and physician order for blood transfusion match occurs, but before the first subsequent infuse a unit of blood product occurs:
   • Nurse 1 makes sure that ID band and physician order for blood transfusion match is allowed to occur again, zero or more times;
   • Nurse 2 makes sure that ID band and physician order for blood transfusion match is allowed to occur zero or more times.
8. After nurse 1 makes sure that ID band and physician order for blood transfusion match and the first subsequent infuse a unit of blood product occur:
   • Nurse 2 makes sure that ID band and physician order for blood transfusion match is allowed to occur zero or more times;
   • Infuse a unit of blood product is not allowed to occur again until after another nurse 1 makes sure that ID band and physician order for blood transfusion match occurs; nurse 1 makes sure that ID band and physician order for blood transfusion match is allowed to occur again and, if it does, then the situation is the same as when the first nurse 1 makes sure that ID band and physician order for blood transfusion match occurred, meaning that the restrictions described in parts 6, 7, and 8 would again apply.

FSA and DNL for Property BT.B.4.4a
Event alphabet:
- A: nurse 2 makes sure that ID band and physician order for blood transfusion match
- B: infuse a unit of blood product
- C: nurse 1 makes sure that ID band and physician order for blood transfusion match

SCOPE:
1. From the start of any event sequence through to the end of that event sequence, the behavior must hold.

BEHAVIOR:
1. The events of primary interest in this behavior are nurse 2 makes sure that ID band and physician order for blood transfusion match and infuse a unit of blood product.
2. The event of secondary interest in this behavior is nurse 1 makes sure that ID band and physician order for blood transfusion match.
3. infuse a unit of blood product is not allowed to occur until after nurse 2 makes sure that ID band and physician order for blood transfusion match occurs.
4. Before the first nurse 2 makes sure that ID band and physician order for blood transfusion match occurs, nurse 1 makes sure that ID band and physician order for blood transfusion match is allowed to occur zero or more times.
5. Even if nurse 2 makes sure that ID band and physician order for blood transfusion match occurs, infuse a unit of blood product is not required to occur after nurse 2 makes sure that ID band and physician order for blood transfusion match occurs.
6. After nurse 2 makes sure that ID band and physician order for blood transfusion match occurs, infuse a unit of blood product is not allowed to occur again until after another nurse 2 makes sure that ID band and physician order for blood transfusion match occurs, nurse 2 makes sure that ID band and physician order for blood transfusion match is allowed to occur again and, if it does, then the situation is the same as when the first nurse 2 makes sure that ID band and physician order for blood transfusion match occurred, meaning that the restrictions described in parts 6, 7, and 8 would again apply.

FSA and DNL for Property BT.B.4.4b
Event alphabet:
- A: nurse 1 confirms presence of ID band
- B: nurse 1 obtains Patient’s stated name and birth date
- START: unit of blood product arrives
- END: infuse a unit of blood product
- C: nurse 1 makes sure that ID band and Patient’s stated name and birth date match, nurse 1 makes sure that ID band and physician order for blood transfusion match, nurse 2 confirms presence of ID band, nurse 2 makes sure that ID band and Patient’s stated name and birth date match, nurse 2 makes sure that ID band and physician order for blood transfusion match, nurse 2 obtains Patient’s stated name and birth date

SCOPE:
1. A restricted interval in the event sequence can have both a starting delimiter, **unit of blood product arrives**, and an ending delimiter, **infuse a unit of blood product**.
2. The behavior is required to hold from an occurrence of **unit of blood product arrives**, if it ever occurs, through to the first subsequent occurrence of **infuse a unit of blood product**, if it ever occurs.
3. If there are multiple occurrences of **unit of blood product arrives** without an occurrence of **infuse a unit of blood product** in between them, only the first of those occurrences of **unit of blood product arrives** potentially starts a restricted interval; later occurrences of **unit of blood product arrives** within this restricted interval do not have an effect.
4. **unit of blood product arrives** is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if **unit of blood product arrives** does occur, **infuse a unit of blood product** is not required to occur subsequently and if it does not occur subsequently, then the behavior is not required to hold for the remainder of the event sequence.
5. There might be many restricted intervals in the event sequence. If **unit of blood product arrives** occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of **unit of blood product arrives**, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a potential new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

BEHAVIOR:
1. The events of primary interest in this behavior are **nurse 1 confirms presence of ID band** and **nurse 1 obtains Patient’s stated name and birth date**.
2. The events of secondary interest in this behavior are **nurse 1 confirms presence of ID band** and **nurse 1 obtains Patient’s stated name and birth date**, **nurse 1 makes sure that ID band and Patient’s stated name and birth date match**, **nurse 2 confirms presence of ID band**, **nurse 2 makes sure that ID band and Patient’s stated name and birth date match**, **nurse 2 makes sure that ID band and physician order for blood transfusion match**, and **nurse 2 obtains Patient’s stated name and birth date**.
3. **nurse 1 obtains Patient’s stated name and birth date** is not allowed to occur until after **nurse 1 confirms presence of ID band** occurs.
4. Before the first **nurse 1 confirms presence of ID band** occurs, all the events of secondary interest are allowed to occur zero or more times.
5. **nurse 1 confirms presence of ID band** is not required to occur.
6. Even if **nurse 1 confirms presence of ID band** does occur, **nurse 1 obtains Patient’s stated name and birth date** is not required to occur after **nurse 1 confirms presence of ID band** occurs.
7. After **nurse 1 confirms presence of ID band** occurs, but before the first subsequent **nurse 1 obtains Patient’s stated name and birth date** occurs:
   - **nurse 1 confirms presence of ID band** is allowed to occur again, zero or more times;
   - None of the events of secondary interest are allowed to occur.
8. After **nurse 1 confirms presence of ID band** and the first subsequent **nurse 1 obtains Patient’s stated name and birth date** occurs:
   - All the events of secondary interest are allowed to occur zero or more times;
   - Both **nurse 1 confirms presence of ID band** and **nurse 1 obtains Patient’s stated name and birth date** are allowed to occur again, zero or more times, and there are no restrictions on the occurrences of any future events.

FSA and DNL for Property BT.B.4.5a (B.4.1a → B.4.2a)
Event alphabet:
- A: nurse 1 obtains Patient’s stated name and birth date
- B: nurse 1 makes sure that ID band and Patient’s stated name and birth date match
- START: unit of blood product arrives
- END: infuse a unit of blood product
- C: nurse 1 confirms presence of ID band, nurse 1 makes sure that ID band and physician order for blood transfusion match, nurse 2 confirms presence of ID band, nurse 2 makes sure that ID band and Patient’s stated name and birth date match, nurse 2 makes sure that ID band and physician order for blood transfusion match, nurse 2 obtains Patient’s stated name and birth date

SCOPE:
1. A restricted interval in the event sequence can have both a starting delimiter, unit of blood product arrives, and an ending delimiter, infuse a unit of blood product.
2. The behavior is required to hold from an occurrence of unit of blood product arrives, if it ever occurs, through to the first subsequent occurrence of infuse a unit of blood product, if it ever occurs.
3. If there are multiple occurrences of unit of blood product arrives without an occurrence of infuse a unit of blood product in between them, only the first of those occurrences of unit of blood product arrives potentially starts a restricted interval; later occurrences of unit of blood product arrives within this restricted interval do not have an effect.
4. Infuse a unit of blood product is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if unit of blood product arrives does occur, infuse a unit of blood product is not required to occur subsequently and if it does not occur subsequently, then the behavior is not required to hold for the remainder of the event sequence.
5. There might be many restricted intervals in the event sequence. If unit of blood product arrives occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of unit of blood product arrives, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a potential new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

BEHAVIOR:
1. The events of primary interest in this behavior are nurse 1 obtains Patient’s stated name and birth date and nurse 1 makes sure that ID band and Patient’s stated name and birth date match.
2. The events of secondary interest in this behavior are nurse 1 confirms presence of ID band, nurse 1 makes sure that ID band and physician order for blood transfusion match, nurse 2 confirms presence of ID band, nurse 2 makes sure that ID band and Patient’s stated name and birth date match, nurse 2 makes sure that ID band and physician order for blood transfusion match, and nurse 2 obtains Patient’s stated name and birth date.
3. Nurse 1 makes sure that ID band and Patient’s stated name and birth date match is not allowed to occur until after nurse 1 obtains Patient’s stated name and birth date occurs.
4. Before the first nurse 1 obtains Patient’s stated name and birth date occurs, all the events of secondary interest are allowed to occur zero or more times.
5. Nurse 1 obtains Patient’s stated name and birth date is not required to occur.
6. Even if nurse 1 obtains Patient’s stated name and birth date occurs, nurse 1 makes sure that ID band and Patient’s stated name and birth date match is not required to occur after nurse 1 obtains Patient’s stated name and birth date occurs.
7. After nurse 1 obtains Patient’s stated name and birth date occurs, but before the first subsequent nurse 1 makes sure that ID band and Patient’s stated name and birth date match occurs:
   - Nurse 1 obtains Patient’s stated name and birth date is allowed to occur again, zero or more times;
   - None of the events of secondary interest are allowed to occur.
8. After nurse 1 obtains Patient’s stated name and birth date and the first subsequent nurse 1 makes sure that ID band and Patient’s stated name and birth date match occur:
   - All the events of secondary interest are allowed to occur zero or more times;
   - Both nurse 1 obtains Patient’s stated name and birth date and nurse 1 makes sure that ID band and Patient’s stated name and birth date match are allowed to occur again, zero or more times, and there are no restrictions on the occurrences of any future events.
Event alphabet:
- A: nurse 1 makes sure that ID band and Patient’s stated name and birth date match
- B: nurse 1 makes sure that ID band and physician order for blood transfusion match
- START: unit of blood product arrives
- END: infuse a unit of blood product
- C: nurse 1 confirms presence of ID band, nurse 1 obtains Patient’s stated name and birth date, nurse 2 confirms presence of ID band, nurse 2 makes sure that ID band and Patient’s stated name and birth date match, nurse 2 makes sure that ID band and physician order for blood transfusion match, nurse 2 obtains Patient’s stated name and birth date

SCOPE:
1. A restricted interval in the event sequence can have both a starting delimiter, unit of blood product arrives, and an ending delimiter, infuse a unit of blood product.
2. The behavior is required to hold from an occurrence of unit of blood product arrives, if it ever occurs, through to the first subsequent occurrence of infuse a unit of blood product, if it ever occurs.
3. If there are multiple occurrences of unit of blood product arrives without an occurrence of infuse a unit of blood product in between them, only the first occurrence of unit of blood product arrives potentially starts a restricted interval; later occurrences of unit of blood product arrives within this restricted interval do not have an effect.
4. Infuse a unit of blood product is not required to occur and if it never occurs, the behavior is not required to hold anywhere in the event sequence. Even if unit of blood product arrives does occur, infuse a unit of blood product is not required to occur subsequently and if it does not occur subsequently, then the behavior is not required to hold for the remainder of the event sequence.
5. There might be many restricted intervals in the event sequence. If unit of blood product arrives occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of unit of blood product arrives, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a potential new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

BEHAVIOR:
1. The events of primary interest in this behavior are nurse 1 makes sure that ID band and Patient’s stated name and birth date match and nurse 1 makes sure that ID band and physician order for blood transfusion match.
2. The events of secondary interest in this behavior are nurse 1 confirms presence of ID band, nurse 1 obtains Patient’s stated name and birth date, nurse 2 confirms presence of ID band, nurse 2 makes sure that ID band and Patient’s stated name and birth date match, nurse 2 makes sure that ID band and physician order for blood transfusion match, and nurse 2 obtains Patient’s stated name and birth date.
3. Nurse 1 makes sure that ID band and physician order for blood transfusion match is not allowed to occur until after nurse 1 makes sure that ID band and Patient’s stated name and birth date match occurs.
4. Before the first nurse 1 makes sure that ID band and Patient’s stated name and birth date match occurs, all the events of secondary interest are allowed to occur zero or more times.
5. Nurse 1 makes sure that ID band and Patient’s stated name and birth date match is not required to occur.
6. Even if nurse 1 makes sure that ID band and Physician order for blood transfusion match is not required to occur after nurse 1 makes sure that ID band and Patient’s stated name and birth date match occurs, nurse 1 makes sure that ID band and physician order for blood transfusion match is not required to occur again, zero or more times.
7. Nurse 1 makes sure that ID band and Patient’s stated name and birth date match occurs, but before the first subsequent nurse 1 makes sure that ID band and physician order for blood transfusion match occurs:
   - nurse 1 makes sure that ID band and Patient’s stated name and birth date match is allowed to occur again, zero or more times;
   - None of the events of secondary interest are allowed to occur.
8. After nurse 1 makes sure that ID band and Patient’s stated name and birth date match and the first subsequent nurse 1 makes sure that ID band and physician order for blood transfusion match occur:
   - All the events of secondary interest are allowed to occur zero or more times;
   - Both nurse 1 makes sure that ID band and Patient’s stated name and birth date match and nurse 1 makes sure that ID band and physician order for blood transfusion match are allowed to occur again, zero or more times, and there are no restrictions on the occurrences of any future events.

FSA and DNL for Property BT.B.4.5a (B.4.3a → B.4.4a)
Event alphabet:
- A: nurse 2 confirms presence of ID band
- B: nurse 2 obtains Patient’s stated name and birth date
- START: unit of blood product arrives
- END: infuse a unit of blood product
- C: nurse 1 confirms presence of ID band, nurse 1 makes sure that ID band and Patient’s stated name and birth date match, nurse 1 makes sure that ID band and physician order for blood transfusion match, nurse 1 obtains Patient’s stated name and birth date, nurse 2 makes sure that ID band and Patient’s stated name and birth date match, nurse 2 makes sure that ID band and physician order for blood transfusion match

**SCOPE:**
1. A restricted interval in the event sequence can have both a starting delimiter, unit of blood product arrives, and an ending delimiter, infuse a unit of blood product.
2. The behavior is required to hold from an occurrence of unit of blood product arrives, if it ever occurs, through to the first subsequent occurrence of infuse a unit of blood product, if it ever occurs.
3. If there are multiple occurrences of unit of blood product arrives without an occurrence of infuse a unit of blood product in between them, only the first of those occurrences of unit of blood product arrives potentially starts a restricted interval; later occurrences of unit of blood product arrives within this restricted interval do not have an effect.
4. unit of blood product arrives is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if unit of blood product arrives does occur, infuse a unit of blood product is not required to occur subsequently and if it does not occur subsequently, then the behavior is not required to hold for the remainder of the event sequence.
5. There might be many restricted intervals in the event sequence. If unit of blood product arrives occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of unit of blood product arrives, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a potential new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

**BEHAVIOR:**
1. The events of primary interest in this behavior are nurse 2 confirms presence of ID band and nurse 2 obtains Patient’s stated name and birth date.
2. The events of secondary interest in this behavior are nurse 1 confirms presence of ID band, nurse 1 makes sure that ID band and Patient’s stated name and birth date match, nurse 1 makes sure that ID band and physician order for blood transfusion match, nurse 1 obtains Patient’s stated name and birth date, nurse 2 makes sure that ID band and Patient’s stated name and birth date match, and nurse 2 makes sure that ID band and physician order for blood transfusion match.
3. nurse 2 obtains Patient’s stated name and birth date is not allowed to occur until after nurse 2 confirms presence of ID band occurs.
4. Before the first nurse 2 confirms presence of ID band occurs, all the events of secondary interest are allowed to occur zero or more times.
5. nurse 2 confirms presence of ID band is not required to occur.
6. Even if nurse 2 confirms presence of ID band does occur, nurse 2 obtains Patient’s stated name and birth date is not required to occur after nurse 2 confirms presence of ID band occurs.
7. After nurse 2 confirms presence of ID band occurs, but before the first subsequent nurse 2 obtains Patient’s stated name and birth date occurs:
   - nurse 2 confirms presence of ID band is allowed to occur again, zero or more times;
   - None of the events of secondary interest are allowed to occur.
8. After nurse 2 confirms presence of ID band and the first subsequent nurse 2 obtains Patient’s stated name and birth date occur:
   - All the events of secondary interest are allowed to occur zero or more times;
   - Both nurse 2 confirms presence of ID band and nurse 2 obtains Patient’s stated name and birth date are allowed to occur again, zero or more times, and there are no restrictions on the occurrences of any future events.

FSA and DNL for Property BT.B.4.5b (B.4.1b → B.4.2b)
Event alphabet:
- A: nurse 2 obtains Patient’s stated name and birth date
- B: nurse 2 makes sure that ID band and Patient’s stated name and birth date match
- START: unit of blood product arrives
- END: infuse a unit of blood product
- C: nurse 1 confirms presence of ID band, nurse 1 makes sure that ID band and Patient’s stated name and birth date match, nurse 1 makes sure that ID band and physician order for blood transfusion match, nurse 1 obtains Patient’s stated name and birth date, nurse 2 confirms presence of ID band, nurse 2 makes sure that ID band and physician order for blood transfusion match

SCAPE:
1. A restricted interval in the event sequence can have both a starting delimiter, unit of blood product arrives, and an ending delimiter, infuse a unit of blood product.
2. The behavior is required to hold from an occurrence of unit of blood product arrives, if it ever occurs, through to the first subsequent occurrence of infuse a unit of blood product, if it ever occurs.
3. If there are multiple occurrences of unit of blood product arrives without an occurrence of infuse a unit of blood product in between them, only the first of those occurrences of unit of blood product arrives potentially starts a restricted interval; later occurrences of unit of blood product arrives within this restricted interval do not have an effect.
4. Unit of blood product arrives is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if unit of blood product arrives does occur, infuse a unit of blood product is not required to occur subsequently and if it does not occur subsequently, then the behavior is not required to hold for the remainder of the event sequence.
5. There might be many restricted intervals in the event sequence. If unit of blood product arrives occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of unit of blood product arrives, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a potential new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

BEHAVIOR:
1. The events of primary interest in this behavior are nurse 2 obtains Patient’s stated name and birth date and nurse 2 makes sure that ID band and Patient’s stated name and birth date match.
2. The events of secondary interest in this behavior are nurse 1 confirms presence of ID band, nurse 1 makes sure that ID band and Patient’s stated name and birth date match, nurse 2 makes sure that ID band and physician order for blood transfusion match, nurse 1 obtains Patient’s stated name and birth date, nurse 2 confirms presence of ID band, and nurse 2 makes sure that ID band and physician order for blood transfusion match.
3. Nurse 2 makes sure that ID band and Patient’s stated name and birth date match is not allowed to occur until after nurse 2 obtains Patient’s stated name and birth date occurs.
4. Before the first nurse 2 obtains Patient’s stated name and birth date occurs, all the events of secondary interest are allowed to occur zero or more times.
5. Nurse 2 obtains Patient’s stated name and birth date is not required to occur.
6. Even if nurse 2 obtains Patient’s stated name and birth date does occur, nurse 2 makes sure that ID band and Patient’s stated name and birth date match is not required to occur after nurse 2 obtains Patient’s stated name and birth date occurs.
7. After nurse 2 obtains Patient’s stated name and birth date occurs, but before the first subsequent nurse 2 makes sure that ID band and Patient’s stated name and birth date match occurs:
   - nurse 2 obtains Patient’s stated name and birth date is allowed to occur again, zero or more times;
   - None of the events of secondary interest are allowed to occur.
8. After nurse 2 obtains Patient’s stated name and birth date and the first subsequent nurse 2 makes sure that ID band and Patient’s stated name and birth date match occur:
   - All the events of secondary interest are allowed to occur zero or more times;
   - Both nurse 2 obtains Patient’s stated name and birth date and nurse 2 makes sure that ID band and Patient’s stated name and birth date match are allowed to occur again, zero or more times, and there are no restrictions on the occurrences of any future events.

FSA and DNL for Property BT.B.4.5b (B.4.2b → B.4.3b)
Event alphabet:
- A: nurse 2 makes sure that ID band and Patient’s stated name and birth date match
- B: nurse 2 makes sure that ID band and physician order for blood transfusion match
- START: unit of blood product arrives
- END: infuse a unit of blood product
- C: nurse 1 confirms presence of ID band, nurse 1 makes sure that ID band and Patient’s stated name and birth date match, nurse 1 obtains Patient’s stated name and birth date

Scope:
1. A restricted interval in the event sequence can have both a starting delimiter, unit of blood product arrives, and an ending delimiter, infuse a unit of blood product.
2. The behavior is required to hold from an occurrence of unit of blood product arrives, if it ever occurs, through to the first subsequent occurrence of infuse a unit of blood product, if it ever occurs.
3. If there are multiple occurrences of unit of blood product arrives without an occurrence of infuse a unit of blood product in between them, only the first of those occurrences of unit of blood product arrives potentially starts a restricted interval; later occurrences of unit of blood product arrives within this restricted interval do not have an effect.
4. unit of blood product arrives is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if unit of blood product arrives does occur, infuse a unit of blood product is not required to occur subsequently and if it does not occur subsequently, then the behavior is not required to hold for the remainder of the event sequence.
5. There might be many restricted intervals in the event sequence. If unit of blood product arrives occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of unit of blood product arrives, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a potential new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

Behavior:
1. The events of primary interest in this behavior are nurse 2 makes sure that ID band and Patient’s stated name and birth date match and nurse 2 makes sure that ID band and physician order for blood transfusion match.
2. The events of secondary interest in this behavior are nurse 1 confirms presence of ID band, nurse 1 makes sure that ID band and Patient’s stated name and birth date match, nurse 1 obtains Patient’s stated name and birth date.
3. nurse 2 makes sure that ID band and physician order for blood transfusion match is not allowed to occur until after nurse 2 makes sure that ID band and Patient’s stated name and birth date match occurs.
4. Before the first nurse 2 makes sure that ID band and Patient’s stated name and birth date match occurs, all the events of secondary interest are allowed to occur zero or more times.
5. nurse 2 makes sure that ID band and Patient’s stated name and birth date match is not required to occur.
6. Even if nurse 2 makes sure that ID band and Patient’s stated name and birth date match does occur, nurse 2 makes sure that ID band and physician order for blood transfusion match is not required to occur after nurse 2 makes sure that ID band and Patient’s stated name and birth date match occurs.
7. After nurse 2 makes sure that ID band and Patient’s stated name and birth date match occurs, but before the first subsequent nurse 2 makes sure that ID band and physician order for blood transfusion match occurs:
   - nurse 2 makes sure that ID band and Patient’s stated name and birth date match is allowed to occur again, zero or more times;
   - None of the events of secondary interest are allowed to occur.
8. After nurse 2 makes sure that ID band and Patient’s stated name and birth date match and the first subsequent nurse 2 makes sure that ID band and physician order for blood transfusion match occurs:
   - All the events of secondary interest are allowed to occur zero or more times;
   - Both nurse 2 makes sure that ID band and Patient’s stated name and birth date match and nurse 2 makes sure that ID band and physician order for blood transfusion match are allowed to occur again, zero or more times, and there are no restrictions on the occurrences of any future events.

FSA and DNL for Property BT.B.4.5b (B.4.3b → B.4.4b)
Event alphabet:
- A: nurse 1 makes sure that ID band and tag affixed to unit of blood product i match
- B: infuse unit of blood product i
- C: nurse 2 makes sure that ID band and tag affixed to unit of blood product i match

SCOPE:
1. From the start of any event sequence through to the end of that event sequence, the behavior must hold.

BEHAVIOR:
1. The events of primary interest in this behavior are nurse 1 makes sure that ID band and tag affixed to unit of blood product i match and infuse unit of blood product i.
2. The event of secondary interest in this behavior is nurse 2 makes sure that ID band and tag affixed to unit of blood product i match.
3. infuse unit of blood product i is not allowed to occur until after nurse 1 makes sure that ID band and tag affixed to unit of blood product i match occurs.
4. Before the first nurse 1 makes sure that ID band and tag affixed to unit of blood product i match occurs, nurse 2 makes sure that ID band and tag affixed to unit of blood product i match is allowed to occur zero or more times.
5. Even if nurse 1 makes sure that ID band and tag affixed to unit of blood product i match does occur, infuse unit of blood product i is not required to occur after nurse 1 makes sure that ID band and tag affixed to unit of blood product i match occurs.
6. After nurse 1 makes sure that ID band and tag affixed to unit of blood product i match occurs, but before the first subsequent infuse unit of blood product i occurs:
   - nurse 1 makes sure that ID band and tag affixed to unit of blood product i match is allowed to occur again, zero or more times;
   - nurse 2 makes sure that ID band and tag affixed to unit of blood product i match is allowed to occur zero or more times.
7. After nurse 1 makes sure that ID band and tag affixed to unit of blood product i match occurs, but before the first subsequent infuse unit of blood product i and the first subsequent infuse unit of blood product i occur:
   - nurse 2 makes sure that ID band and tag affixed to unit of blood product i match is allowed to occur zero or more times;
   - infuse unit of blood product i is not allowed to occur again until after another nurse 1 makes sure that ID band and tag affixed to unit of blood product i match occurs; nurse 1 makes sure that ID band and tag affixed to unit of blood product i match is allowed to occur again and, if it does, then the situation is the same as when the first nurse 1 makes sure that ID band and tag affixed to unit of blood product i match occurred, meaning that the restrictions described in parts 6, 7, and 8 would again apply.

FSA and DNL for Property BT.B.5.1a

284
Event alphabet:
- A: nurse 2 makes sure that ID band and tag affixed to unit of blood product i match
- B: infuse unit of blood product i
- C: nurse 1 makes sure that ID band and tag affixed to unit of blood product i match

**SCOPE:**
1. From the start of any event sequence through to the end of that event sequence, the behavior must hold.

**BEHAVIOR:**
1. The events of primary interest in this behavior are nurse 2 makes sure that ID band and tag affixed to unit of blood product i match and infuse unit of blood product i.
2. The event of secondary interest in this behavior is nurse 1 makes sure that ID band and tag affixed to unit of blood product i match.
3. Infuse unit of blood product i is not allowed to occur until after nurse 2 makes sure that ID band and tag affixed to unit of blood product i match occurs.
4. Before the first nurse 2 makes sure that ID band and tag affixed to unit of blood product i match occurs, nurse 1 makes sure that ID band and tag affixed to unit of blood product i match is allowed to occur zero or more times.
5. Nurse 2 makes sure that ID band and tag affixed to unit of blood product i match is not required to occur.
6. Even if nurse 2 makes sure that ID band and tag affixed to unit of blood product i match does occur, infuse unit of blood product i is not required to occur after nurse 2 makes sure that ID band and tag affixed to unit of blood product i match occurs.
7. After nurse 2 makes sure that ID band and tag affixed to unit of blood product i match occurs, but before the first subsequent infuse unit of blood product i occurs:
   - Nurse 2 makes sure that ID band and tag affixed to unit of blood product i match is allowed to occur again, zero or more times;
   - Nurse 1 makes sure that ID band and tag affixed to unit of blood product i match is allowed to occur zero or more times.
8. After nurse 2 makes sure that ID band and tag affixed to unit of blood product i match and the first subsequent infuse unit of blood product i occurs:
   - Nurse 1 makes sure that ID band and tag affixed to unit of blood product i match is allowed to occur zero or more times;
   - Infuse unit of blood product i is not allowed to occur again until after another nurse 2 makes sure that ID band and tag affixed to unit of blood product i match occurs; nurse 2 makes sure that ID band and tag affixed to unit of blood product i match is allowed to occur again and, if it does, then the situation is the same as when the first nurse 2 makes sure that ID band and tag affixed to unit of blood product i match occurred, meaning that the restrictions described in parts 6, 7, and 8 would again apply.

**FSA and DNL for Property BT.B.5.1b**
Event alphabet:
- A: nurse 1 makes sure that tag affixed to unit of blood product i and unit of blood product i match
- B: infuse unit of blood product i
- C: nurse 2 makes sure that tag affixed to unit of blood product i and unit of blood product i match

**SCOPE:**
1. From the start of any event sequence through to the end of that event sequence, the behavior must hold.

**BEHAVIOR:**
1. The events of primary interest in this behavior are nurse 1 makes sure that tag affixed to unit of blood product i and unit of blood product i match and infuse unit of blood product i.
2. The event of secondary interest in this behavior is nurse 2 makes sure that tag affixed to unit of blood product i and unit of blood product i match.
3. Infuse unit of blood product i is not allowed to occur until after nurse 1 makes sure that tag affixed to unit of blood product i and unit of blood product i match occurs.
4. Before the first nurse 1 makes sure that tag affixed to unit of blood product i and unit of blood product i match occurs, nurse 2 makes sure that tag affixed to unit of blood product i and unit of blood product i match is allowed to occur zero or more times.
5. Nurse 1 makes sure that tag affixed to unit of blood product i and unit of blood product i match is not required to occur.
6. Even if nurse 1 makes sure that tag affixed to unit of blood product i and unit of blood product i match does occur, infuse unit of blood product i is not required to occur after nurse 1 makes sure that tag affixed to unit of blood product i and unit of blood product i match occurs.
7. After nurse 1 makes sure that tag affixed to unit of blood product i and unit of blood product i match occurs, but before the first subsequent infuse unit of blood product i occurs:
   - Nurse 1 makes sure that tag affixed to unit of blood product i and unit of blood product i match is allowed to occur again, zero or more times.
   - Nurse 2 makes sure that tag affixed to unit of blood product i and unit of blood product i match is allowed to occur zero or more times.
8. After nurse 1 makes sure that tag affixed to unit of blood product i and unit of blood product i match and the first subsequent infuse unit of blood product i occur:
   - Nurse 2 makes sure that tag affixed to unit of blood product i and unit of blood product i match is allowed to occur zero or more times.
   - Infuse unit of blood product i is not allowed to occur again until after another nurse 1 makes sure that tag affixed to unit of blood product i and unit of blood product i match occurs; nurse 1 makes sure that tag affixed to unit of blood product i and unit of blood product i match is allowed to occur again and, if it does, then the situation is the same as when the first nurse 1 makes sure that tag affixed to unit of blood product i and unit of blood product i match occurred, meaning that the restrictions described in parts 6, 7, and 8 would again apply.

**FSA and DNL for Property BT.B.5.2a**
Event alphabet:
• A: nurse 2 makes sure that tag affixed to unit of blood product i and unit of blood product i match
• B: infuse unit of blood product i
• C: nurse 1 makes sure that tag affixed to unit of blood product i and unit of blood product i match

SCOPE:
1. From the start of any event sequence through to the end of that event sequence, the behavior must hold.

BEHAVIOR:
1. The events of primary interest in this behavior are nurse 2 makes sure that tag affixed to unit of blood product i and unit of blood product i match and infuse unit of blood product i.
2. The event of secondary interest in this behavior is nurse 1 makes sure that tag affixed to unit of blood product i and unit of blood product i match.
3. Infuse unit of blood product i is not allowed to occur until after nurse 2 makes sure that tag affixed to unit of blood product i and unit of blood product i match occurs.
4. Before the first nurse 2 makes sure that tag affixed to unit of blood product i and unit of blood product i match occurs, nurse 1 makes sure that tag affixed to unit of blood product i and unit of blood product i match is allowed to occur zero or more times.
5. Nurse 2 makes sure that tag affixed to unit of blood product i and unit of blood product i match is not required to occur.
6. Even if nurse 2 makes sure that tag affixed to unit of blood product i and unit of blood product i match does occur, infuse unit of blood product i is not required to occur after nurse 2 makes sure that tag affixed to unit of blood product i and unit of blood product i match occurs.
7. After nurse 2 makes sure that tag affixed to unit of blood product i and unit of blood product i match occurs, but before the first subsequent infuse unit of blood product i occurs:
   • nurse 2 makes sure that tag affixed to unit of blood product i and unit of blood product i match is allowed to occur again, zero or more times;
   • nurse 1 makes sure that tag affixed to unit of blood product i and unit of blood product i match is allowed to occur zero or more times.
8. After nurse 2 makes sure that tag affixed to unit of blood product i and unit of blood product i match and the first subsequent infuse unit of blood product i occurs:
   • nurse 1 makes sure that tag affixed to unit of blood product i and unit of blood product i match is allowed to occur zero or more times;
   • infuse unit of blood product i is not allowed to occur again until after another nurse 2 makes sure that tag affixed to unit of blood product i and unit of blood product i match occurs; nurse 2 makes sure that tag affixed to unit of blood product i and unit of blood product i match is allowed to occur again and, if it does, then the situation is the same as when the first nurse 2 makes sure that tag affixed to unit of blood product i and unit of blood product i match occurred, meaning that the restrictions described in parts 6, 7, and 8 would again apply.

FSA and DNL for Property BT.B.5.2b

287
Event alphabet:
- A: nurse 1 makes sure that unit of blood product i has not expired
- B: infuse unit of blood product i
- C: nurse 2 makes sure that unit of blood product i has not expired

SCOPE:
1. From the start of any event sequence through to the end of that event sequence, the behavior must hold.

BEHAVIOR:
1. The events of primary interest in this behavior are nurse 1 makes sure that unit of blood product i has not expired and infuse unit of blood product i.
2. The event of secondary interest in this behavior is nurse 2 makes sure that unit of blood product i has not expired.
3. Infuse unit of blood product i is not allowed to occur until after nurse 1 makes sure that unit of blood product i has not expired occurs.
4. Before the first nurse 1 makes sure that unit of blood product i has not expired occurs, nurse 2 makes sure that unit of blood product i has not expired is allowed to occur zero or more times.
5. Nurse 1 makes sure that unit of blood product i has not expired is not required to occur.
6. If nurse 1 makes sure that unit of blood product i has not expired does occur, infuse unit of blood product i is not required to occur after nurse 1 makes sure that unit of blood product i has not expired occurs.
7. After nurse 1 makes sure that unit of blood product i has not expired occurs, but before the first subsequent infuse unit of blood product i occurs:
   - nurse 1 makes sure that unit of blood product i has not expired is allowed to occur again, zero or more times;
   - nurse 2 makes sure that unit of blood product i has not expired is allowed to occur zero or more times.
8. After nurse 1 makes sure that unit of blood product i has not expired and the first subsequent infuse unit of blood product i occurs:
   - nurse 2 makes sure that unit of blood product i has not expired is allowed to occur zero or more times;
   - infuse unit of blood product i is not allowed to occur again until after another nurse 1 makes sure that unit of blood product i has not expired occurs; nurse 1 makes sure that unit of blood product i has not expired is allowed to occur again and, if it does, then the situation is the same as when the first nurse 1 makes sure that unit of blood product i has not expired occurred, meaning that the restrictions described in parts 6, 7, and 8 would again apply.

FSA and DNL for Property BT.B.5.3a
Event alphabet:
- A: nurse 2 makes sure that unit of blood product i has not expired
- B: infuse unit of blood product i
- C: nurse 1 makes sure that unit of blood product i has not expired

SCOPE:
1. From the start of any event sequence through to the end of that event sequence, the behavior must hold.

BEHAVIOR:
1. The events of primary interest in this behavior are nurse 2 makes sure that unit of blood product i has not expired and infuse unit of blood product i.
2. The event of secondary interest in this behavior is nurse 1 makes sure that unit of blood product i has not expired.
3. Infuse unit of blood product i is not allowed to occur until after nurse 2 makes sure that unit of blood product i has not expired occurs.
4. Before the first nurse 2 makes sure that unit of blood product i has not expired occurs, nurse 1 makes sure that unit of blood product i has not expired is allowed to occur zero or more times.
5. Nurse 2 makes sure that unit of blood product i has not expired is not required to occur.
6. Even if nurse 2 makes sure that unit of blood product i has not expired does occur, infuse unit of blood product i is not required to occur after nurse 2 makes sure that unit of blood product i has not expired occurs.
7. After nurse 2 makes sure that unit of blood product i has not expired occurs, but before the first subsequent infuse unit of blood product i occurs:
   - Nurse 2 makes sure that unit of blood product i has not expired is allowed to occur again, zero or more times;
   - Nurse 1 makes sure that unit of blood product i has not expired is allowed to occur zero or more times.
8. After nurse 2 makes sure that unit of blood product i has not expired and the first subsequent infuse unit of blood product i occurs:
   - Nurse 1 makes sure that unit of blood product i has not expired is allowed to occur zero or more times;
   - Infuse unit of blood product i is not allowed to occur again until after another nurse 2 makes sure that unit of blood product i has not expired occurs; nurse 2 makes sure that unit of blood product i has not expired is allowed to occur again and, if it does, then the situation is the same as when the first nurse 2 makes sure that unit of blood product i has not expired occurred, meaning that the restrictions described in parts 6, 7, and 8 would again apply.

FSA and DNL for Property BT.B.5.3b
Event alphabet:
- **A**: nurse 1 makes sure that ID band and tag affixed to unit of blood product i match
- **B**: nurse 1 makes sure that tag affixed to unit of blood product i and unit of blood product i match
- **START**: unit of blood product i arrives
- **END**: infuse unit of blood product i
- **C**: assess Patient’s baseline single-unit status, nurse 1 makes sure that ID band and physician order for blood transfusion match, nurse 1 makes sure that unit of blood product i has not expired, nurse 2 makes sure that ID band and tag affixed to unit of blood product i match, nurse 2 makes sure that tag affixed to unit of blood product i and unit of blood product i match, nurse 2 makes sure that unit of blood product i has not expired

**SCOPE:**
1. A restricted interval in the event sequence can have both a starting delimiter, unit of blood product i arrives, and an ending delimiter, infuse unit of blood product i.
2. The behavior is required to hold from an occurrence of unit of blood product i arrives, if it ever occurs, through to the first subsequent occurrence of infuse unit of blood product i, if it ever occurs.
3. If there are multiple occurrences of unit of blood product i arrives without an occurrence of infuse unit of blood product i in between them, only the first of those occurrences of unit of blood product i arrives potentially starts a restricted interval; later occurrences of unit of blood product i arrives within this restricted interval do not have an effect.
4. If unit of blood product i arrives is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if unit of blood product i arrives does occur, infuse unit of blood product i is not required to occur subsequently and if it does not occur subsequently, then the behavior is not required to hold for the remainder of the event sequence.
5. There might be many restricted intervals in the event sequence. If unit of blood product i arrives occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of unit of blood product i arrives, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a potential new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

**BEHAVIOR:**
1. The events of primary interest in this behavior are nurse 1 makes sure that ID band and tag affixed to unit of blood product i match and nurse 1 makes sure that tag affixed to unit of blood product i and unit of blood product i match.
2. The events of secondary interest in this behavior are assess Patient’s baseline single-unit status, nurse 1 makes sure that ID band and physician order for blood transfusion match, nurse 1 makes sure that unit of blood product i has not expired, nurse 2 makes sure that ID band and tag affixed to unit of blood product i match, nurse 2 makes sure that ID band and tag affixed to unit of blood product i and unit of blood product i match, and nurse 2 makes sure that unit of blood product i has not expired.
3. Nurse 1 makes sure that tag affixed to unit of blood product i and unit of blood product i match is not allowed to occur until after nurse 1 makes sure that ID band and tag affixed to unit of blood product i match occurs.
4. Before the first nurse 1 makes sure that ID band and tag affixed to unit of blood product i match occurs, all the events of secondary interest are allowed to occur zero or more times.
5. Nurse 1 makes sure that ID band and tag affixed to unit of blood product i match is not required to occur.
6. Even if nurse 1 makes sure that ID band and tag affixed to unit of blood product i match does occur, nurse 1 makes sure that tag affixed to unit of blood product i and unit of blood product i match is not required to occur after nurse 1 makes sure that ID band and tag affixed to unit of blood product i match occurs.
7. After nurse 1 makes sure that ID band and tag affixed to unit of blood product i match occurs, but before the first subsequent nurse 1 makes sure that ID band and tag affixed to unit of blood product i and unit of blood product i match occurs:
   - Nurse 1 makes sure that ID band and tag affixed to unit of blood product i match is allowed to occur again, zero or more times;
   - None of the events of secondary interest are allowed to occur.
8. After nurse 1 makes sure that ID band and tag affixed to unit of blood product i match and the first subsequent nurse 1 makes sure that tag affixed to unit of blood product i and unit of blood product i match occur:
   - All the events of secondary interest are allowed to occur zero or more times;
   - Both nurse 1 makes sure that ID band and tag affixed to unit of blood product i match and nurse 1 makes sure that tag affixed to unit of blood product i and unit of blood product i match are allowed to occur again, zero or more times, and there are no restrictions on the occurrences of any future events.

FSA and DNL for Property BT.B.5.4a (B.5.1a → B.5.2a)
Event alphabet:
- A: nurse 1 makes sure that tag affixed to unit of blood product i and unit of blood product i match
- B: nurse 1 makes sure that unit of blood product i has not expired
- START: unit of blood product i arrives
- END: infuse unit of blood product i
- C: assess Patient’s baseline single-unit status, nurse 1 makes sure that ID band and physician order for blood transfusion match, nurse 1 makes sure that ID band and tag affixed to unit of blood product i match, nurse 2 makes sure that ID band and tag affixed to unit of blood product i match, nurse 2 makes sure that tag affixed to unit of blood product i and unit of blood product i match, nurse 2 makes sure that unit of blood product i has not expired

SCOPE:
1. A restricted interval in the event sequence can have both a starting delimiter, unit of blood product i arrives, and an ending delimiter, infuse unit of blood product i.
2. The behavior is required to hold from an occurrence of unit of blood product i arrives, if it ever occurs, through to the first subsequent occurrence of infuse unit of blood product i, if it ever occurs.
3. If there are multiple occurrences of unit of blood product i arrives without an occurrence of infuse unit of blood product i in between them, only the first of those occurrences of unit of blood product i arrives potentially starts a restricted interval; later occurrences of unit of blood product i arrives within this restricted interval do not have an effect.
4. unit of blood product i arrives is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if unit of blood product i arrives does occur, infuse unit of blood product i is not required to occur subsequently and if it does not occur subsequently, then the behavior is not required to hold for the remainder of the event sequence.
5. There might be many restricted intervals in the event sequence. If unit of blood product i arrives occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of unit of blood product i arrives, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a potential new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

BEHAVIOR:
1. The events of primary interest in this behavior are nurse 1 makes sure that tag affixed to unit of blood product i and unit of blood product i match and nurse 1 makes sure that unit of blood product i has not expired.
2. The events of secondary interest in this behavior are assess Patient’s baseline single-unit status, nurse 1 makes sure that ID band and physician order for blood transfusion match, nurse 1 makes sure that ID band and tag affixed to unit of blood product i match, nurse 2 makes sure that ID band and physician order for blood transfusion match, nurse 2 makes sure that ID band and tag affixed to unit of blood product i match, nurse 2 makes sure that unit of blood product i and unit of blood product i match, and nurse 2 makes sure that unit of blood product i has not expired.
3. nurse 1 makes sure that unit of blood product i has not expired is not allowed to occur until after nurse 1 makes sure that tag affixed to unit of blood product i and unit of blood product i match occurs.
4. Before the first nurse 1 makes sure that tag affixed to unit of blood product i and unit of blood product i match occurs, all the events of secondary interest are allowed to occur zero or more times.
5. nurse 1 makes sure that tag affixed to unit of blood product i and unit of blood product i match is not required to occur.
6. Even if nurse 1 makes sure that tag affixed to unit of blood product i and unit of blood product i match does occur, nurse 1 makes sure that unit of blood product i has not expired is not required to occur after nurse 1 makes sure that tag affixed to unit of blood product i and unit of blood product i match occurs.
7. After nurse 1 makes sure that tag affixed to unit of blood product i and unit of blood product i match occurs, but before the first subsequent nurse 1 makes sure that unit of blood product i has not expired occurs:
   - nurse 1 makes sure that tag affixed to unit of blood product i and unit of blood product i match is allowed to occur again, zero or more times;
   - None of the events of secondary interest are allowed to occur.
8. After nurse 1 makes sure that tag affixed to unit of blood product i and unit of blood product i match and the first subsequent nurse 1 makes sure that unit of blood product i has not expired occur:
   - All the events of secondary interest are allowed to occur zero or more times;
   - Both nurse 1 makes sure that tag affixed to unit of blood product i and unit of blood product i match and nurse 1 makes sure that unit of blood product i has not expired are allowed to occur again, zero or more times, and there are no restrictions on the occurrences of any future events.

FSA and DNL for Property BT.B.5.4a (B.5.2a → B.5.3a)
Event alphabet:
- A: nurse 2 makes sure that ID band and tag affixed to unit of blood product i match
- B: nurse 2 makes sure that tag affixed to unit of blood product i and unit of blood product i match
- START: unit of blood product i arrives
- END: infuse unit of blood product i
- C: assess Patient’s baseline single-unit status, nurse 1 makes sure that ID band and physician order for blood transfusion match, nurse 1 makes sure that ID band and tag affixed to unit of blood product i match, nurse 1 makes sure that unit of blood product i has not expired, nurse 2 makes sure that ID band and physician order for blood transfusion match, nurse 2 makes sure that tag affixed to unit of blood product i and unit of blood product i match
- A, B, END, C

SCOPE:
1. A restricted interval in the event sequence can have both a starting delimiter, unit of blood product i arrives, and an ending delimiter, infuse unit of blood product i.
2. The behavior is required to hold from an occurrence of unit of blood product i arrives, if it ever occurs, through to the first subsequent occurrence of infuse unit of blood product i, if it ever occurs.
3. If there are multiple occurrences of unit of blood product i arrives without an occurrence of infuse unit of blood product i in between them, only the first of those occurrences of unit of blood product i arrives potentially starts a restricted interval; later occurrences of unit of blood product i arrives within this restricted interval do not have an effect.
4. unit of blood product i arrives is not required to occur if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if unit of blood product i arrives does occur, infuse unit of blood product i is not required to occur subsequently and if it does not occur subsequently, then the behavior is not required to hold for the remainder of the event sequence.
5. There might be many restricted intervals in the event sequence. If unit of blood product i arrives occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of unit of blood product i arrives, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a potential new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

BEHAVIOR:
1. The events of primary interest in this behavior are nurse 2 makes sure that ID band and tag affixed to unit of blood product i match and nurse 2 makes sure that tag affixed to unit of blood product i and unit of blood product i match.
2. The events of secondary interest in this behavior are assess Patient’s baseline single-unit status, nurse 1 makes sure that ID band and physician order for blood transfusion match, nurse 1 makes sure that ID band and tag affixed to unit of blood product i match, nurse 1 makes sure that tag affixed to unit of blood product i and unit of blood product i match, nurse 1 makes sure that unit of blood product i has not expired, nurse 2 makes sure that ID band and physician order for blood transfusion match, and nurse 2 makes sure that unit of blood product i has not expired.
3. nurse 2 makes sure that tag affixed to unit of blood product i and unit of blood product i match is not allowed to occur until after nurse 2 makes sure that ID band and tag affixed to unit of blood product i match occurs.
4. Before the first nurse 2 makes sure that ID band and tag affixed to unit of blood product i match occurs, all the events of secondary interest are allowed to occur zero or more times.
5. nurse 2 makes sure that ID band and tag affixed to unit of blood product i match is not required to occur.
6. Even if nurse 2 makes sure that ID band and tag affixed to unit of blood product i match occurs, nurse 2 makes sure that tag affixed to unit of blood product i and unit of blood product i match is not required to occur after nurse 2 makes sure that ID band and tag affixed to unit of blood product i match occurs.
7. After nurse 2 makes sure that ID band and tag affixed to unit of blood product i match occurs, but before the first subsequent nurse 2 makes sure that ID band and tag affixed to unit of blood product i and unit of blood product i match occurs:
   - nurse 2 makes sure that ID band and tag affixed to unit of blood product i match is allowed to occur again, zero or more times;
   - None of the events of secondary interest are allowed to occur.
8. After nurse 2 makes sure that ID band and tag affixed to unit of blood product i match and the first subsequent nurse 2 makes sure that tag affixed to unit of blood product i and unit of blood product i match occurs:
   - All the events of secondary interest are allowed to occur zero or more times;
   - Both nurse 2 makes sure that ID band and tag affixed to unit of blood product i match and nurse 2 makes sure that tag affixed to unit of blood product i and unit of blood product i match are allowed to occur again, zero or more times, and there are no restrictions on the occurrences of any future events.

FSA and DNL for Property BT.B.5.4b (B.5.1b → B.5.2b)
Event alphabet:

- A: nurse 2 makes sure that tag affixed to unit of blood product i and unit of blood product i match
- B: nurse 2 makes sure that unit of blood product i has not expired
- C: assess Patient’s baseline single-unit status, nurse 1 makes sure that ID band and physician order for blood transfusion match, nurse 1 makes sure that ID band and tag affixed to unit of blood product i match, nurse 1 makes sure that tag affixed to unit of blood product i and unit of blood product i match, nurse 1 makes sure that unit of blood product i has not expired, nurse 2 makes sure that ID band and physician order for blood transfusion match, nurse 2 makes sure that ID band and tag affixed to unit of blood product i match

SCOPE:

1. A restricted interval in the event sequence can have both a starting delimiter, unit of blood product i arrives, and an ending delimiter, infuse unit of blood product i.
2. The behavior is required to hold from an occurrence of unit of blood product i arrives, if it ever occurs, through to the first subsequent occurrence of infuse unit of blood product i, if it ever occurs.
3. If there are multiple occurrences of unit of blood product i arrives without an occurrence of infuse unit of blood product i in between them, only the first of those occurrences of unit of blood product i arrives potentially starts a restricted interval; later occurrences of unit of blood product i arrives within this restricted interval do not have an effect.
4. If unit of blood product i arrives is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if unit of blood product i arrives occurs, infuse unit of blood product i is not required to occur subsequently and if it does not occur subsequently, then the behavior is not required to hold for the remainder of the event sequence.
5. There might be many restricted intervals in the event sequence. If unit of blood product i arrives occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of unit of blood product i arrives, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a potential new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

BEHAVIOR:

1. The events of primary interest in this behavior are nurse 2 makes sure that tag affixed to unit of blood product i and unit of blood product i match and nurse 2 makes sure that unit of blood product i has not expired.
2. The events of secondary interest in this behavior are assess Patient’s baseline single-unit status, nurse 1 makes sure that ID band and physician order for blood transfusion match, nurse 1 makes sure that ID band and tag affixed to unit of blood product i match, nurse 1 makes sure that tag affixed to unit of blood product i and unit of blood product i match, nurse 1 makes sure that unit of blood product i has not expired, nurse 2 makes sure that ID band and physician order for blood transfusion match, and nurse 2 makes sure that ID band and tag affixed to unit of blood product i match.
3. nurse 2 makes sure that unit of blood product i has not expired is not allowed to occur until after nurse 2 makes sure that tag affixed to unit of blood product i and unit of blood product i match occurs.
4. Before the first nurse 2 makes sure that tag affixed to unit of blood product i and unit of blood product i match occurs, all the events of secondary interest are allowed to occur zero or more times.
5. nurse 2 makes sure that tag affixed to unit of blood product i and unit of blood product i match occurs, and all events of secondary interest are allowed to occur zero or more times.
6. If nurse 2 makes sure that tag affixed to unit of blood product i and unit of blood product i match does occur, nurse 2 makes sure that unit of blood product i has not expired is not required to occur after nurse 2 makes sure that tag affixed to unit of blood product i and unit of blood product i match occurs.
7. After nurse 2 makes sure that tag affixed to unit of blood product i and unit of blood product i match occurs, but before the first subsequent nurse 2 makes sure that unit of blood product i has not expired occurs:
   - nurse 2 makes sure that tag affixed to unit of blood product i and unit of blood product i match is allowed to occur again, zero or more times;
   - None of the events of secondary interest are allowed to occur.
8. After nurse 2 makes sure that tag affixed to unit of blood product i and unit of blood product i match occurs, and the first subsequent nurse 2 makes sure that unit of blood product i has not expired occurs:
   - All the events of secondary interest are allowed to occur zero or more times;
   - Both nurse 2 makes sure that tag affixed to unit of blood product i and unit of blood product i match and nurse 2 makes sure that unit of blood product i has not expired are allowed to occur again, zero or more times, and there are no restrictions on the occurrences of any future events.
Event alphabet:
- A: nurse 1 makes sure that ID band and physician order for blood transfusion match
- B: nurse 1 makes sure that ID band and tag affixed to unit of blood product i match
- START: unit of blood product i arrives
- END: infuse unit of blood product i
- C: assess Patient’s baseline single-unit status, nurse 1 makes sure that tag affixed to unit of blood product i and unit of blood product i match, nurse 1 makes sure that ID band and tag affixed to unit of blood product i match, nurse 2 makes sure that ID band and tag affixed to unit of blood product i match, nurse 2 makes sure that unit of blood product i has not expired

SCOPE:
1. A restricted interval in the event sequence can have both a starting delimiter, unit of blood product i arrives, and an ending delimiter, infuse unit of blood product i.
2. The behavior is required to hold from an occurrence of unit of blood product i arrives, if it ever occurs, through to the first subsequent occurrence of infuse unit of blood product i, if it ever occurs.
3. If there are multiple occurrences of unit of blood product i arrives without an occurrence of infuse unit of blood product i in between them, only the first of those occurrences of unit of blood product i arrives potentially starts a restricted interval; later occurrences of unit of blood product i arrives within this restricted interval do not have an effect.
4. unit of blood product i arrives is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if unit of blood product i arrives does occur, infuse unit of blood product i is not required to occur subsequently and if it does not occur subsequently, then the behavior is not required to hold for the remainder of the event sequence.
5. There might be many restricted intervals in the event sequence. If unit of blood product i arrives occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of unit of blood product i arrives, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a potential new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

BEHAVIOR:
1. The events of primary interest in this behavior are nurse 1 makes sure that ID band and physician order for blood transfusion match and nurse 1 makes sure that ID band and tag affixed to unit of blood product i match.
2. The events of secondary interest in this behavior are assess Patient’s baseline single-unit status, nurse 1 makes sure that tag affixed to unit of blood product i and unit of blood product i match, nurse 1 makes sure that unit of blood product i has not expired, nurse 2 makes sure that ID band and physician order for blood transfusion match does occur, nurse 2 makes sure that ID band and tag affixed to unit of blood product i match occurs; nurse 2 makes sure that tag affixed to unit of blood product i match occurs.
3. nurse 1 makes sure that ID band and tag affixed to unit of blood product i match is not allowed to occur until after nurse 1 makes sure that ID band and physician order for blood transfusion match occurs.
4. Before the first occurrence of nurse 1 makes sure that ID band and tag affixed to unit of blood product i match is not required to occur after nurse 1 makes sure that ID band and physician order for blood transfusion match occurs.
5. nurse 1 makes sure that ID band and physician order for blood transfusion match is not required to occur.
6. Even if nurse 1 makes sure that ID band and physician order for blood transfusion match does occur, nurse 1 makes sure that ID band and tag affixed to unit of blood product i match occurs; nurse 1 makes sure that ID band and tag affixed to unit of blood product i match occurs.
7. After nurse 1 makes sure that ID band and physician order for blood transfusion match occurs, but before the first subsequent nurse 1 makes sure that ID band and tag affixed to unit of blood product i match occurs:
   - nurse 1 makes sure that ID band and physician order for blood transfusion match is allowed to occur again, zero or more times;
   - None of the events of secondary interest are allowed to occur.
8. After nurse 1 makes sure that ID band and physician order for blood transfusion match and the first subsequent nurse 1 makes sure that ID band and tag affixed to unit of blood product i match occurs:
   - All the events of secondary interest are allowed to occur zero or more times;
   - Both nurse 1 makes sure that ID band and physician order for blood transfusion match and nurse 1 makes sure that ID band and tag affixed to unit of blood product i match are allowed to occur again, zero or more times, and there are no restrictions on the occurrences of any future events.

FSA and DNL for Property BT.B.7a (B.4.4a → B.5.1a)
Event alphabet:
- A: nurse 1 makes sure that unit of blood product i has not expired
- B: assess Patient’s baseline single-unit status
- END: infuse unit of blood product i
- C: nurse 1 makes sure that ID band and physician order for blood transfusion match, nurse 1 makes sure that ID band and tag affixed to unit of blood product i match, nurse 1 makes sure that tag affixed to unit of blood product i and unit of blood product i match, nurse 2 makes sure that tag affixed to unit of blood product i match, nurse 2 makes sure that unit of blood product i has not expired

**SCOPE:**
1. A restricted interval in the event sequence can have both a starting delimiter, unit of blood product i arrives, and an ending delimiter, infuse unit of blood product i.
2. The behavior is required to hold from an occurrence of unit of blood product i arrives, and an ending delimiter, infuse unit of blood product i.
3. If there are multiple occurrences of unit of blood product i arrives without an occurrence of infuse unit of blood product i in between them, only the first of those occurrences of unit of blood product i arrives potentially starts a restricted interval; later occurrences of unit of blood product i arrives within this restricted interval do not have an effect.
4. If unit of blood product i arrives is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if unit of blood product i arrives does occur, infuse unit of blood product i is not required to occur subsequently and if it does not occur subsequently, then the behavior is not required to hold for the remainder of the event sequence.
5. There might be many restricted intervals in the event sequence. If unit of blood product i arrives occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of unit of blood product i arrives, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a potential new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

**BEHAVIOR:**
1. The events of primary interest in this behavior are nurse 1 makes sure that unit of blood product i has not expired and assess Patient’s baseline single-unit status.
2. The events of secondary interest in this behavior are nurse 1 makes sure that ID band and physician order for blood transfusion match, nurse 1 makes sure that ID band and tag affixed to unit of blood product i match, nurse 1 makes sure that tag affixed to unit of blood product i and unit of blood product i match, nurse 2 makes sure that ID band and physician order for blood transfusion match, nurse 2 makes sure that ID band and tag affixed to unit of blood product i match, nurse 2 makes sure that tag affixed to unit of blood product i and unit of blood product i match, and nurse 2 makes sure that unit of blood product i has not expired.
3. assess Patient’s baseline single-unit status is not allowed to occur until after nurse 1 makes sure that unit of blood product i has not expired occurs.
4. Before the first nurse 1 makes sure that unit of blood product i has not expired occurs, all the events of secondary interest are allowed to occur zero or more times.
5. nurse 1 makes sure that unit of blood product i has not expired is not required to occur.
6. Even if nurse 1 makes sure that unit of blood product i has not expired does occur, assess Patient’s baseline single-unit status is not required to occur after nurse 1 makes sure that unit of blood product i has not expired occurs.
7. After nurse 1 makes sure that unit of blood product i has not expired occurs, but before the first subsequent assess Patient’s baseline single-unit status occurs:
   - nurse 1 makes sure that unit of blood product i has not expired is allowed to occur again, zero or more times;
   - None of the events of secondary interest are allowed to occur.
8. After nurse 1 makes sure that unit of blood product i has not expired and the first subsequent assess Patient’s baseline single-unit status occur:
   - All the events of secondary interest are allowed to occur zero or more times;
   - Both nurse 1 makes sure that unit of blood product i has not expired and assess Patient’s baseline single-unit status are allowed to occur again, zero or more times, and there are no restrictions on the occurrences of any future events.

FSA and DNL for Property BT.B.7a (B.5.3a → B.6)
Event alphabet:
- A: nurse 2 makes sure that ID band and physician order for blood transfusion match
- B: nurse 2 makes sure that ID band and tag affixed to unit of blood product i match
- START: unit of blood product i arrives
- END: infuse unit of blood product i
- C: assess Patient’s baseline single-unit status

SCOPE:
1. A restricted interval in the event sequence can have both a starting delimiter, unit of blood product i arrives, and an ending delimiter, infuse unit of blood product i.
2. The behavior is required to hold from an occurrence of unit of blood product i arrives, if it ever occurs, through to the first subsequent occurrence of infuse unit of blood product i, if it ever occurs.
3. If there are multiple occurrences of unit of blood product i arrives without an occurrence of infuse unit of blood product i in between them, only the first of those occurrences of unit of blood product i arrives potentially starts a restricted interval; later occurrences of unit of blood product i arrives within this restricted interval do not have an effect.
4. unit of blood product i arrives is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if unit of blood product i arrives does occur, infuse unit of blood product i is not required to occur subsequently and if it does not occur subsequently, then the behavior is not required to hold for the remainder of the event sequence.
5. There might be many restricted intervals in the event sequence. If unit of blood product i arrives occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of unit of blood product i arrives, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a potential new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

BEHAVIOR:
1. The events of primary interest in this behavior are nurse 2 makes sure that ID band and physician order for blood transfusion match and nurse 2 makes sure that ID band and tag affixed to unit of blood product i match.
2. The events of secondary interest in this behavior are assess Patient’s baseline single-unit status, nurse 1 makes sure that ID band and physician order for blood transfusion match, nurse 1 makes sure that ID band and tag affixed to unit of blood product i match, nurse 1 makes sure that tag affixed to unit of blood product i and unit of blood product i match, nurse 1 makes sure that unit of blood product i has not expired, nurse 2 makes sure that tag affixed to unit of blood product i and unit of blood product i match, nurse 2 makes sure that unit of blood product i has not expired.

FSA and DNL for Property BT.B.7b (B.4.4b → B.5.1b)
Event alphabet:
- A: nurse 2 makes sure that unit of blood product i has not expired
- B: assess Patient’s baseline single-unit status
- START: unit of blood product i arrives
- END: infuse unit of blood product i
- C: nurse 1 makes sure that ID band and physician order for blood transfusion match, nurse 1 makes sure that ID band and tag affixed to unit of blood product i match, nurse 1 makes sure that tag affixed to unit of blood product i and unit of blood product i match, nurse 2 makes sure that ID band and tag affixed to unit of blood product i match, nurse 2 makes sure that tag affixed to unit of blood product i and unit of blood product i match
- • Both nurse 2 makes sure that unit of blood product i has not expired and assess Patient’s baseline single-unit status are allowed to occur zero or more times.

SCOPE:
1. A restricted interval in the event sequence can have both a starting delimiter, unit of blood product i arrives, and an ending delimiter, infuse unit of blood product i.
2. The behavior is required to hold from an occurrence of unit of blood product i arrives, if it ever occurs, through to the first subsequent occurrence of infuse unit of blood product i, if it ever occurs.
3. If there are multiple occurrences of unit of blood product i arrives without an occurrence of infuse unit of blood product i in between them, only the first of those occurrences of unit of blood product i arrives potentially starts a restricted interval; later occurrences of unit of blood product i arrives within this restricted interval do not have an effect.
4. unit of blood product i arrives is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if unit of blood product i arrives does occur, infuse unit of blood product i is not required to occur subsequently and if it does not occur subsequently, then the behavior is not required to hold for the remainder of the event sequence.
5. There might be many restricted intervals in the event sequence. If unit of blood product i arrives occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of unit of blood product i arrives, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a potential new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

BEHAVIOR:
1. The events of primary interest in this behavior are nurse 2 makes sure that unit of blood product i has not expired and assess Patient’s baseline single-unit status.
2. The events of secondary interest in this behavior are nurse 1 makes sure that ID band and physician order for blood transfusion match, nurse 1 makes sure that ID band and tag affixed to unit of blood product i match, nurse 1 makes sure that tag affixed to unit of blood product i and unit of blood product i match, nurse 1 makes sure that unit of blood product i has not expired, nurse 2 makes sure that ID band and physician order for blood transfusion match, nurse 2 makes sure that ID band and tag affixed to unit of blood product i match, and nurse 2 makes sure that tag affixed to unit of blood product i and unit of blood product i match.
3. assess Patient’s baseline single-unit status is not allowed to occur until after nurse 2 makes sure that unit of blood product i has not expired occurs.
4. Before the first nurse 2 makes sure that unit of blood product i has not expired occurs, all the events of secondary interest are allowed to occur zero or more times.
5. nurse 2 makes sure that unit of blood product i has not expired is not required to occur.
6. Even if nurse 2 makes sure that unit of blood product i has not expired does occur, assess Patient’s baseline single-unit status is not required to occur after nurse 2 makes sure that unit of blood product i has not expired occurs.
7. After nurse 2 makes sure that unit of blood product i has not expired occurs, but before the first subsequent assess Patient’s baseline single-unit status occurs:
   - nurse 2 makes sure that unit of blood product i has not expired is allowed to occur again, zero or more times;
   - None of the events of secondary interest are allowed to occur.
8. After nurse 2 makes sure that unit of blood product i has not expired and the first subsequent assess Patient’s baseline single-unit status occur:
   - All the events of secondary interest are allowed to occur zero or more times;
   - Both nurse 2 makes sure that unit of blood product i has not expired and assess Patient’s baseline single-unit status are allowed to occur again, zero or more times, and there are no restrictions on the occurrences of any future events.

FSA and DNL for Property BT.B.7b (B.5.3b → B.6)
EVENT ALPHABET:
- A: assess Patient’s baseline single-unit status
- B: infuse a unit of blood product
- C: administer other medications, ask physician for instructions, find problematic Patient assessment

SCOPE:
1. From the start of any event sequence through to the end of that event sequence, the behavior must hold.

BEHAVIOR:
1. The events of primary interest in this behavior are assess Patient’s baseline single-unit status and infuse a unit of blood product.
2. The events of secondary interest in this behavior are administer other medications, ask physician for instructions, and find problematic Patient assessment.
3. infuse a unit of blood product is not allowed to occur until after assess Patient’s baseline single-unit status occurs.
4. Before the first assess Patient’s baseline single-unit status occurs, all the events of secondary interest are allowed to occur zero or more times.
5. assess Patient’s baseline single-unit status is not required to occur.
6. Even if assess Patient’s baseline single-unit status does occur, infuse a unit of blood product is not required to occur after assess Patient’s baseline single-unit status occurs.
7. After assess Patient’s baseline single-unit status occurs, but before the first subsequent infuse a unit of blood product occurs:
   - assess Patient’s baseline single-unit status is allowed to occur again, zero or more times;
   - All the events of secondary interest are allowed to occur zero or more times.
8. After assess Patient’s baseline single-unit status and the first subsequent infuse a unit of blood product occur:
   - All the events of secondary interest are allowed to occur zero or more times;
   - infuse a unit of blood product is not allowed to occur again until after another assess Patient’s baseline single-unit status occurs;
   - assess Patient’s baseline single-unit status is allowed to occur again and, if it does, then the situation is the same as when the first assess Patient’s baseline single-unit status occurred, meaning that the restrictions described in parts 6, 7, and 8 would again apply.

FSA and DNL for Property BT.B.6
Event alphabet:
- A: ask physician for instructions
- START: find problematic Patient assessment
- END: infuse a unit of blood product
- C: administer other medications, assess Patient’s baseline single-unit status

SCOPE:
1. A restricted interval in the event sequence can have both a starting delimiter, find problematic Patient assessment, and an ending delimiter, infuse a unit of blood product.
2. The behavior is required to hold from an occurrence of find problematic Patient assessment, if it ever occurs, through to the first subsequent occurrence of infuse a unit of blood product, if it ever occurs.
3. If there are multiple occurrences of find problematic Patient assessment without an occurrence of infuse a unit of blood product in between them, only the last of those occurrences of find problematic Patient assessment starts a restricted interval; each of those occurrences of find problematic Patient assessment resets the beginning of this restricted interval.
4. find problematic Patient assessment is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if find problematic Patient assessment does occur, infuse a unit of blood product is not required to occur subsequently. Even if infuse a unit of blood product does not occur subsequently, the behavior is still required to hold, until the end of the event sequence.
5. There might be many restricted intervals in the event sequence. If find problematic Patient assessment occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of find problematic Patient assessment, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

BEHAVIOR:
1. The event of primary interest in this behavior is ask physician for instructions.
2. The events of secondary interest in this behavior are administer other medications and assess Patient’s baseline single-unit status.
3. ask physician for instructions is required to occur at least once.

FSA and DNL for Property BT.B.8
Event alphabet:
- A: confirm presence of informed consent
- B: perform transfusion

SCOPE:
1. From the start of any event sequence through to the end of that event sequence, the behavior must hold.

BEHAVIOR:
1. The events of primary interest in this behavior are confirm presence of informed consent and perform transfusion.
2. There are no events of secondary interest in this behavior.
3. perform transfusion is not allowed to occur until after confirm presence of informed consent occurs.
4. confirm presence of informed consent is required to occur, whether or not perform transfusion eventually occurs.
5. perform transfusion is not required to occur after confirm presence of informed consent occurs.
6. After confirm presence of informed consent occurs, but before the first subsequent perform transfusion occurs, confirm presence of informed consent is allowed to occur again, zero or more times.
7. After confirm presence of informed consent and the first subsequent perform transfusion occur:
   - Both confirm presence of informed consent and perform transfusion are allowed to occur again, zero or more times, and there are no restrictions on the occurrences of any future events.

FSA and DNL for Property BT.B.1
Event alphabet:

- **A**: pick up unit of blood product i from blood bank
- **START**: unit of blood product i expires
- **END**: infuse unit of blood product i

**SCOPE:**

1. A restricted interval in the event sequence can have both a starting delimiter, **unit of blood product i expires**, and an ending delimiter, **infuse unit of blood product i**.
2. The behavior is required to hold from an occurrence of **unit of blood product i expires**, if it ever occurs, through to the first subsequent occurrence of **infuse unit of blood product i**, if it ever occurs.
3. If there are multiple occurrences of **unit of blood product i expires** without an occurrence of **infuse unit of blood product i** in between them, only the last of those occurrences of **unit of blood product i expires** potentially starts a restricted interval; each of those occurrences of **unit of blood product i expires** resets the beginning of this restricted interval.
4. **unit of blood product i expires** is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if **unit of blood product i expires** does occur, **infuse unit of blood product i** is not required to occur subsequently and if it does not occur subsequently, then the behavior is not required to hold for the remainder of the event sequence.
5. There might be many restricted intervals in the event sequence. If **unit of blood product i expires** occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of **unit of blood product i expires**, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a potential new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

**BEHAVIOR:**

1. The event of primary interest in this behavior is **pick up unit of blood product i from blood bank**.
2. There are no events of secondary interest in this behavior.
3. **pick up unit of blood product i from blood bank** is required to occur at least once.

**FSA and DNL for Property BT.B.10**
Event alphabet:
- A: first 15 minutes of infusion passes
- B: assess Patient’s 15-minute single-unit status

SCOPE:
1. From the start of any event sequence through to the end of that event sequence, the behavior must hold.

BEHAVIOR:
1. The events of primary interest in this behavior are **first 15 minutes of infusion passes** and **assess Patient’s 15-minute single-unit status**.
2. There are no events of secondary interest in this behavior.
3. If **first 15 minutes of infusion passes** occurs, **assess Patient’s 15-minute single-unit status** is required to occur subsequently.
4. Before the first **first 15 minutes of infusion passes** occurs, **assess Patient’s 15-minute single-unit status** is not allowed to occur.
5. **First 15 minutes of infusion passes** is not required to occur.
6. After **first 15 minutes of infusion passes** occurs, but before the first subsequent **assess Patient’s 15-minute single-unit status** occurs, **first 15 minutes of infusion passes** is not allowed to occur again.
7. After **first 15 minutes of infusion passes** and the first subsequent **assess Patient’s 15-minute single-unit status** occur:
   - **assess Patient’s 15-minute single-unit status** is allowed to occur again, zero or more times, before another **first 15 minutes of infusion passes** occurs; **first 15 minutes of infusion passes** is allowed to occur again and, if it does, then the situation is the same as when the first **first 15 minutes of infusion passes** occurred, meaning that the restrictions described in parts 3, 6, and 7 would again apply.

FSA and DNL for Property BT.B.11
Event alphabet:
- A: infuse of a unit of blood product
- B: assess Patient’s post-single-unit status
- C: *, discard transfusion materials, record infusion information

SCOPE:
1. From the start of any event sequence through to the end of that event sequence, the behavior must hold.

BEHAVIOR:
1. The events of primary interest in this behavior are infuse of a unit of blood product and assess Patient’s post-single-unit status.
2. The events of secondary interest in this behavior are *, discard transfusion materials, and record infusion information.
3. If infuse of a unit of blood product occurs, assess Patient’s post-single-unit status is required to occur subsequently.
4. Before the first infuse of a unit of blood product occurs:
   - assess Patient’s post-single-unit status is not allowed to occur;
   - All the events of secondary interest are allowed to occur zero or more times.
5. Infuse of a unit of blood product is not required to occur.
6. After infuse of a unit of blood product occurs, but before the first subsequent assess Patient’s post-single-unit status occurs:
   - infuse of a unit of blood product is not allowed to occur again;
   - None of the events of secondary interest are allowed to occur.
7. After infuse of a unit of blood product and the first subsequent assess Patient’s post-single-unit status occur:
   - All the events of secondary interest are allowed to occur zero or more times.
   - assess Patient’s post-single-unit status is allowed to occur again, zero or more times, before another infuse of a unit of blood product occurs; infuse of a unit of blood product is allowed to occur again and, if it does, then the situation is the same as when the first infuse of a unit of blood product occurred, meaning that the restrictions described in parts 3, 6, and 7 would again apply.

FSA and DNL for Property BT.B.12
Event alphabet:
- A: suspect infusion will exceed infusion time limit OR discover that infusion has exceeded infusion time limit
- B: ask blood bank for further instructions
- START: unit of blood product arrives
- END: record infusion information

SCOPE:
1. A restricted interval in the event sequence can have both a starting delimiter, unit of blood product arrives, and an ending delimiter, record infusion information.
2. The behavior is required to hold from an occurrence of unit of blood product arrives, if it ever occurs, through to the first subsequent occurrence of record infusion information, if it ever occurs.
3. If there are multiple occurrences of unit of blood product arrives without an occurrence of record infusion information in between them, only the first of those occurrences of unit of blood product arrives starts a restricted interval; later occurrences of unit of blood product arrives within this restricted interval do not have an effect.
4. Unit of blood product arrives is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if unit of blood product arrives does occur, record infusion information is not required to occur subsequently. Even if record infusion information does not occur subsequently, the behavior is still required to hold, until the end of the event sequence.
5. There might be many restricted intervals in the event sequence. If unit of blood product arrives occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of unit of blood product arrives, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

BEHAVIOR:
1. The events of primary interest in this behavior are suspect infusion will exceed infusion time limit OR discover that infusion has exceeded infusion time limit and ask blood bank for further instructions.
2. There are no events of secondary interest in this behavior.
3. If suspect infusion will exceed infusion time limit OR discover that infusion has exceeded infusion time limit occurs, ask blood bank for further instructions is required to occur subsequently.
4. Before the first suspect infusion will exceed infusion time limit OR discover that infusion has exceeded infusion time limit occurs, ask blood bank for further instructions is allowed to occur zero or more times.
5. Suspect infusion will exceed infusion time limit OR discover that infusion has exceeded infusion time limit is not required to occur.
6. After suspect infusion will exceed infusion time limit OR discover that infusion has exceeded infusion time limit occurs, but before the first subsequent ask blood bank for further instructions occurs, suspect infusion will exceed infusion time limit OR discover that infusion has exceeded infusion time limit is allowed to occur again, zero or more times.
7. After suspect infusion will exceed infusion time limit OR discover that infusion has exceeded infusion time limit and the first subsequent ask blood bank for further instructions occur:
   - Both suspect infusion will exceed infusion time limit OR discover that infusion has exceeded infusion time limit and ask blood bank for further instructions are allowed to occur again, zero or more times, and there are no restrictions on the occurrences of any future events.

FSA and DNL for Property BT.B.14
**Event alphabet:**
- A: receive blood bank OR physician instruction to resume the infusion of a unit of blood product
- B: resume the infusion of a unit of blood product

**SCOPE:**
1. From the start of any event sequence through to the end of that event sequence, the behavior must hold.

**BEHAVIOR:**
1. The events of primary interest in this behavior are receive blood bank OR physician instruction to resume the infusion of a unit of blood product and resume the infusion of a unit of blood product.
2. There are no events of secondary interest in this behavior.
3. If receive blood bank OR physician instruction to resume the infusion of a unit of blood product occurs, resume the infusion of a unit of blood product is required to occur subsequently.
4. Before the first receive blood bank OR physician instruction to resume the infusion of a unit of blood product occurs, resume the infusion of a unit of blood product is allowed to occur zero or more times.
5. If receive blood bank OR physician instruction to resume the infusion of a unit of blood product occurs, resume the infusion of a unit of blood product is not required to occur.
6. After receive blood bank OR physician instruction to resume the infusion of a unit of blood product occurs, but before the first subsequent resume the infusion of a unit of blood product occurs, receive blood bank OR physician instruction to resume the infusion of a unit of blood product is allowed to occur again, zero or more times.
7. After receive blood bank OR physician instruction to resume the infusion of a unit of blood product and the first subsequent resume the infusion of a unit of blood product occur:
   - resume the infusion of a unit of blood product is allowed to occur again, zero or more times, before another receive blood bank OR physician instruction to resume the infusion of a unit of blood product occurs; receive blood bank OR physician instruction to resume the infusion of a unit of blood product is allowed to occur again and, if it does, then the situation is the same as when the first receive blood bank OR physician instruction to resume the infusion of a unit of blood product occurred, meaning that the restrictions described in parts 3, 6, and 7 would again apply.

FSA and DNL for Property BT.B.15
Event alphabet:
- A: stop the infusion of a unit of blood product
- B: resume OR discontinue the infusion of a unit of blood product

SCOPE:
1. From the start of any event sequence through to the end of that event sequence, the behavior must hold.

BEHAVIOR:
1. The events of primary interest in this behavior are stop the infusion of a unit of blood product and resume OR discontinue the infusion of a unit of blood product.
2. There are no events of secondary interest in this behavior.
3. If stop the infusion of a unit of blood product occurs, resume OR discontinue the infusion of a unit of blood product is required to occur subsequently.
4. Before the first stop the infusion of a unit of blood product occurs, resume OR discontinue the infusion of a unit of blood product is not allowed to occur.
5. stop the infusion of a unit of blood product is not required to occur.
6. After stop the infusion of a unit of blood product occurs, but before the first subsequent resume OR discontinue the infusion of a unit of blood product occurs, stop the infusion of a unit of blood product is not allowed to occur again.
7. After stop the infusion of a unit of blood product and the first subsequent resume OR discontinue the infusion of a unit of blood product occur:
   - resume OR discontinue the infusion of a unit of blood product is not allowed to occur again until after another stop the infusion of a unit of blood product occurs; stop the infusion of a unit of blood product is allowed to occur again and, if it does, then the situation is the same as when the first stop the infusion of a unit of blood product occurred, meaning that the restrictions described in parts 3, 6, and 7 would again apply.

FSA and DNL for Property BT.B.16
Event alphabet:
- A: review Patient history pre-assessment
- B: perform blood transfusion

SCOPE:
1. From the start of any event sequence through to the end of that event sequence, the behavior must hold.

BEHAVIOR:
1. The events of primary interest in this behavior are review Patient history pre-assessment and perform blood transfusion.
2. There are no events of secondary interest in this behavior.
3. perform blood transfusion is not allowed to occur until after review Patient history pre-assessment occurs.
4. review Patient history pre-assessment is required to occur, whether or not perform blood transfusion eventually occurs.
5. perform blood transfusion is not required to occur after review Patient history pre-assessment occurs.
6. After review Patient history pre-assessment occurs, but before the first subsequent perform blood transfusion occurs, review Patient history pre-assessment is allowed to occur again, zero or more times.
7. After review Patient history pre-assessment and the first subsequent perform blood transfusion occur:
   - Both review Patient history pre-assessment and perform blood transfusion are allowed to occur again, zero or more times, and there are no restrictions on the occurrences of any future events.

FSA and DNL for Property BT.B.2
SCOPE:
1. A restricted interval in the event sequence can have both a starting delimiter, `find unknown prior adverse reactions`, and an ending delimiter, `perform blood transfusion`.
2. The behavior is required to hold from an occurrence of `find unknown prior adverse reactions`, if it ever occurs, through to the first subsequent occurrence of `perform blood transfusion`, if it ever occurs.
3. If there are multiple occurrences of `find unknown prior adverse reactions` without an occurrence of `perform blood transfusion` in between them, only the last of those occurrences of `find unknown prior adverse reactions` starts a restricted interval; each of those occurrences of `find unknown prior adverse reactions` resets the beginning of this restricted interval.
4. `find unknown prior adverse reactions` is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if `find unknown prior adverse reactions` does occur, `perform blood transfusion` is not required to occur subsequently. Even if `perform blood transfusion` does not occur subsequently, the behavior is still required to hold, until the end of the event sequence.
5. There might be many restricted intervals in the event sequence. If `find unknown prior adverse reactions` occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of `find unknown prior adverse reactions`, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

BEHAVIOR:
1. The event of primary interest in this behavior is `ask physician for instructions`.
2. There are no events of secondary interest in this behavior.
3. `ask physician for instructions` is required to occur at least once.

FSA and DNL for Property BT.B.3
Event alphabet:
- A: infuse unit of blood product i OR return unit of blood product i to blood bank
- START: pick up unit of blood product i from blood bank
- END: 30-minute deadline passes for unit of blood product i

SCOPE:
1. A restricted interval in the event sequence can have both a starting delimiter, pick up unit of blood product i from blood bank, and an ending delimiter, 30-minute deadline passes for unit of blood product i.
2. The behavior is required to hold from an occurrence of pick up unit of blood product i from blood bank, if it ever occurs, through to the first subsequent occurrence of 30-minute deadline passes for unit of blood product i, if it ever occurs.
3. If there are multiple occurrences of pick up unit of blood product i from blood bank without an occurrence of 30-minute deadline passes for unit of blood product i in between them, only the first of those occurrences of pick up unit of blood product i from blood bank starts a restricted interval; later occurrences of pick up unit of blood product i from blood bank within this restricted interval do not have an effect.
4. pick up unit of blood product i from blood bank is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if pick up unit of blood product i from blood bank does occur, 30-minute deadline passes for unit of blood product i is not required to occur subsequently. Even if 30-minute deadline passes for unit of blood product i does not occur subsequently, the behavior is still required to hold, until the end of the event sequence.
5. There might be many restricted intervals in the event sequence. If pick up unit of blood product i from blood bank occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of pick up unit of blood product i from blood bank, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

BEHAVIOR:
1. The event of primary interest in this behavior is infuse unit of blood product i OR return unit of blood product i to blood bank.
2. There are no events of secondary interest in this behavior.
3. infuse unit of blood product i OR return unit of blood product i to blood bank is required to occur at least once.

FSA and DNL for Property BT.B.9a
Event alphabet:
- A: pick up unit of blood product i from blood bank
- B: 30-minute deadline passes for unit of blood product i
- C: infuse unit of blood product i OR return unit of blood product i to blood bank

SCOPE:
1. From the start of any event sequence through to the end of that event sequence, the behavior must hold.

BEHAVIOR:
1. The events of primary interest in this behavior are pick up unit of blood product i from blood bank and 30-minute deadline passes for unit of blood product i.
2. The event of secondary interest in this behavior is infuse unit of blood product i OR return unit of blood product i to blood bank.
3. If pick up unit of blood product i from blood bank occurs, 30-minute deadline passes for unit of blood product i is required to occur subsequently.
4. Before the first pick up unit of blood product i from blood bank occurs:
   - 30-minute deadline passes for unit of blood product i is not allowed to occur;
   - infuse unit of blood product i OR return unit of blood product i to blood bank is allowed to occur zero or more times.
5. pick up unit of blood product i from blood bank is not required to occur.
6. After pick up unit of blood product i from blood bank occurs, but before the first subsequent 30-minute deadline passes for unit of blood product i occurs:
   - pick up unit of blood product i from blood bank is allowed to occur again, zero or more times;
   - infuse unit of blood product i OR return unit of blood product i to blood bank is allowed to occur zero or more times.
7. After pick up unit of blood product i from blood bank and the first subsequent 30-minute deadline passes for unit of blood product i occur:
   - infuse unit of blood product i OR return unit of blood product i to blood bank is allowed to occur zero or more times;
   - 30-minute deadline passes for unit of blood product i is not allowed to occur again until after another pick up unit of blood product i from blood bank occurs; pick up unit of blood product i from blood bank is allowed to occur again and, if it does, then the situation is the same as when the first pick up unit of blood product i from blood bank occurred, meaning that the restrictions described in parts 3, 6, and 7 would again apply.

FSA and DNL for Property BT.B.9b

310
Event alphabet:
- A: suspect Patient is having a transfusion reaction
- B: stop the infusion of a unit of blood product
- C: *

**SCOPE:**
1. From the start of any event sequence through to the end of that event sequence, the behavior must hold.

**BEHAVIOR:**
1. The events of primary interest in this behavior are **suspect Patient is having a transfusion reaction** and **stop the infusion of a unit of blood product**.
2. The event of secondary interest in this behavior is **C**.
3. If **suspect Patient is having a transfusion reaction** occurs, **stop the infusion of a unit of blood product** is required to occur subsequently.
4. Before the first **suspect Patient is having a transfusion reaction** occurs:
   - **stop the infusion of a unit of blood product** is allowed to occur zero or more times;
   - **C** is allowed to occur zero or more times.
5. **suspect Patient is having a transfusion reaction** is not required to occur.
6. After **suspect Patient is having a transfusion reaction** occurs, but before the first subsequent **stop the infusion of a unit of blood product** occurs:
   - **suspect Patient is having a transfusion reaction** is not allowed to occur again;
   - **C** is not allowed to occur.
7. After **suspect Patient is having a transfusion reaction** and the first subsequent **stop the infusion of a unit of blood product** occur:
   - **C** is allowed to occur zero or more times;
   - **stop the infusion of a unit of blood product** is allowed to occur again, zero or more times, before another **suspect Patient is having a transfusion reaction** occurs; **suspect Patient is having a transfusion reaction** is allowed to occur again and, if it does, then the situation is the same as when the first **suspect Patient is having a transfusion reaction** occurred, meaning that the restrictions described in parts 3, 6, and 7 would again apply.

**FSA and DNL for Property BT.C.1**
Event alphabet:
- A: suspect Patient is having a transfusion reaction
- B: confirm presence of ID band

**SCOPE:**
1. From the start of any event sequence through to the end of that event sequence, the behavior must hold.

**BEHAVIOR:**
1. The events of primary interest in this behavior are **suspect Patient is having a transfusion reaction** and **confirm presence of ID band**.
2. There are no events of secondary interest in this behavior.
3. If **suspect Patient is having a transfusion reaction** occurs, **confirm presence of ID band** is required to occur subsequently.
4. Before the first **suspect Patient is having a transfusion reaction** occurs, **confirm presence of ID band** is allowed to occur zero or more times.
5. **suspect Patient is having a transfusion reaction** is not required to occur.
6. After **suspect Patient is having a transfusion reaction** occurs, but before the first subsequent **confirm presence of ID band** occurs, **suspect Patient is having a transfusion reaction** is not allowed to occur again.
7. After **suspect Patient is having a transfusion reaction** and the first subsequent **confirm presence of ID band** occur:
   - **confirm presence of ID band** is allowed to occur again, zero or more times, before another **suspect Patient is having a transfusion reaction** occurs; **suspect Patient is having a transfusion reaction** is allowed to occur again and, if it does, then the situation is the same as when the first **suspect Patient is having a transfusion reaction** occurred, meaning that the restrictions described in parts 3, 6, and 7 would again apply.

FSA and DNL for Property BT.C.2.1
Event alphabet:
- A: suspect Patient is having a transfusion reaction
- B: obtain Patient’s stated name and birth date

SCOPE:
1. From the start of any event sequence through to the end of that event sequence, the behavior must hold.

BEHAVIOR:
1. The events of primary interest in this behavior are {
   suspect Patient is having a transfusion reaction} and {
   obtain Patient’s stated name and birth date}.
2. There are no events of secondary interest in this behavior.
3. If {
   suspect Patient is having a transfusion reaction} occurs, {
   obtain Patient’s stated name and birth date} is required to occur subsequently.
4. Before the first {
   suspect Patient is having a transfusion reaction} occurs, {
   obtain Patient’s stated name and birth date} is allowed to occur
   zero or more times.
5. {
   suspect Patient is having a transfusion reaction} is not required to occur.
6. After {
   suspect Patient is having a transfusion reaction} occurs, but before the first subsequent {
   obtain Patient’s stated name and birth date} occurs, {
   suspect Patient is having a transfusion reaction} is not allowed to occur again.
7. After {
   suspect Patient is having a transfusion reaction} and the first subsequent {
   obtain Patient’s stated name and birth date} occur:
   - {
     obtain Patient’s stated name and birth date} is allowed to occur again, zero or more times, before another {
     suspect Patient is having a transfusion reaction} occurs; {
     suspect Patient is having a transfusion reaction} is allowed to occur again and, if it does, then the situation is the same as when the first {
     suspect Patient is having a transfusion reaction} occurred, meaning that the restrictions
     described in parts 3, 6, and 7 would again apply.

FSA and DNL for Property BT.C.2.2
### Event alphabet:

- **A**: suspect Patient is having a transfusion reaction
- **B**: make sure that ID band and Patient’s stated name and birth date match

### SCOPE:

1. From the start of any event sequence through to the end of that event sequence, the behavior must hold.

### BEHAVIOR:

1. The events of primary interest in this behavior are suspect Patient is having a transfusion reaction and make sure that ID band and Patient’s stated name and birth date match.
2. There are no events of secondary interest in this behavior.
3. If suspect Patient is having a transfusion reaction occurs, make sure that ID band and Patient’s stated name and birth date match is required to occur subsequently.
4. Before the first suspect Patient is having a transfusion reaction occurs, make sure that ID band and Patient’s stated name and birth date match is allowed to occur zero or more times.
5. If suspect Patient is having a transfusion reaction is not required to occur.
6. After suspect Patient is having a transfusion reaction occurs, but before the first subsequent make sure that ID band and Patient’s stated name and birth date match occurs, suspect Patient is having a transfusion reaction is not allowed to occur again.
7. After suspect Patient is having a transfusion reaction and the first subsequent make sure that ID band and Patient’s stated name and birth date match occur:
   - make sure that ID band and Patient’s stated name and birth date match is allowed to occur again, zero or more times, before another suspect Patient is having a transfusion reaction occurs; suspect Patient is having a transfusion reaction is allowed to occur again and, if it does, then the situation is the same as when the first suspect Patient is having a transfusion reaction occurred, meaning that the restrictions described in parts 3, 6, and 7 would again apply.

---

**FSA and DNL for Property BT.C.2.3**
Event alphabet:
- A: confirm presence of ID band
- B: obtain Patient’s stated name and birth date
- START: suspect Patient is having a transfusion reaction
- END: resume OR discontinue the infusion of a unit of blood product
- C: make sure that ID band and Patient’s stated name and birth date match

SCOPE:
1. A restricted interval in the event sequence can have both a starting delimiter, suspect Patient is having a transfusion reaction, and an ending delimiter, resume OR discontinue the infusion of a unit of blood product.
2. The behavior is required to hold from an occurrence of suspect Patient is having a transfusion reaction, if it ever occurs, through to the first subsequent occurrence of resume OR discontinue the infusion of a unit of blood product, if it ever occurs.
3. If there are multiple occurrences of suspect Patient is having a transfusion reaction without an occurrence of resume OR discontinue the infusion of a unit of blood product in between them, only the first of those occurrences of suspect Patient is having a transfusion reaction starts a restricted interval; later occurrences of suspect Patient is having a transfusion reaction within this restricted interval do not have an effect.
4. suspect Patient is having a transfusion reaction is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if suspect Patient is having a transfusion reaction does occur, resume OR discontinue the infusion of a unit of blood product is not required to occur subsequently. Even if resume OR discontinue the infusion of a unit of blood product does not occur subsequently, the behavior is still required to hold, until the end of the event sequence.
5. There might be many restricted intervals in the event sequence. If suspect Patient is having a transfusion reaction occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of suspect Patient is having a transfusion reaction, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

BEHAVIOR:
1. The events of primary interest in this behavior are confirm presence of ID band and obtain Patient’s stated name and birth date.
2. The event of secondary interest in this behavior is make sure that ID band and Patient’s stated name and birth date match.
3. obtain Patient’s stated name and birth date is not allowed to occur until after confirm presence of ID band occurs.
4. Before the first confirm presence of ID band occurs, make sure that ID band and Patient’s stated name and birth date match is allowed to occur zero or more times.
5. confirm presence of ID band is required to occur, whether or not obtain Patient’s stated name and birth date eventually occurs.
6. obtain Patient’s stated name and birth date is not required to occur after confirm presence of ID band occurs.
7. After confirm presence of ID band occurs, but before the first subsequent obtain Patient’s stated name and birth date occurs:
   - confirm presence of ID band is allowed to occur again, zero or more times;
   - make sure that ID band and Patient’s stated name and birth date match is not allowed to occur.
8. After confirm presence of ID band and the first subsequent obtain Patient’s stated name and birth date occur:
   - make sure that ID band and Patient’s stated name and birth date match is allowed to occur zero or more times;
   - Both confirm presence of ID band and obtain Patient’s stated name and birth date are allowed to occur again, zero or more times, and there are no restrictions on the occurrences of any future events.

FSA and DNL for Property BT.C.2.4a (C.2.1 → C.2.2)
Event alphabet:
- A: obtain Patient’s stated name and birth date
- B: make sure that ID band and Patient’s stated name and birth date match
- C: confirm presence of ID band
- END: resume OR discontinue the infusion of a unit of blood product

**SCOPE:**
1. A restricted interval in the event sequence can have both a starting delimiter, **suspect Patient is having a transfusion reaction**, and an ending delimiter, **resume OR discontinue the infusion of a unit of blood product**.
2. The behavior is required to hold from an occurrence of **suspect Patient is having a transfusion reaction**, if it ever occurs, through to the first subsequent occurrence of **resume OR discontinue the infusion of a unit of blood product**, if it ever occurs.
3. If there are multiple occurrences of **suspect Patient is having a transfusion reaction** without an occurrence of **resume OR discontinue the infusion of a unit of blood product**, in between them, only the first of those occurrences of **suspect Patient is having a transfusion reaction** starts a restricted interval; later occurrences of **suspect Patient is having a transfusion reaction** within this restricted interval do not have an effect.
4. **suspect Patient is having a transfusion reaction** is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if **suspect Patient is having a transfusion reaction** does occur, **resume OR discontinue the infusion of a unit of blood product** is not required to occur subsequently. Even if **resume OR discontinue the infusion of a unit of blood product** does not occur subsequently, the behavior is still required to hold, until the end of the event sequence.
5. There might be many restricted intervals in the event sequence. If **suspect Patient is having a transfusion reaction** occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of **suspect Patient is having a transfusion reaction**, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

**BEHAVIOR:**
1. The events of primary interest in this behavior are **obtain Patient’s stated name and birth date** and **make sure that ID band and Patient’s stated name and birth date match**.
2. The event of secondary interest in this behavior is **confirm presence of ID band**.
3. **make sure that ID band and Patient’s stated name and birth date match** is not allowed to occur until after **obtain Patient’s stated name and birth date** occurs.
4. Before the first **obtain Patient’s stated name and birth date** occurs, **confirm presence of ID band** is allowed to occur zero or more times.
5. **obtain Patient’s stated name and birth date** is required to occur, whether or not **make sure that ID band and Patient’s stated name and birth date match** eventually occurs.
6. **make sure that ID band and Patient’s stated name and birth date match** is not required to occur after **obtain Patient’s stated name and birth date** occurs.
7. After **obtain Patient’s stated name and birth date** occurs, but before the first subsequent **make sure that ID band and Patient’s stated name and birth date match** occurs:
   - **obtain Patient’s stated name and birth date** is allowed to occur again, zero or more times;
   - **confirm presence of ID band** is not allowed to occur.
8. After **obtain Patient’s stated name and birth date** and the first subsequent **make sure that ID band and Patient’s stated name and birth date match** occur:
   - **confirm presence of ID band** is allowed to occur zero or more times;
   - Both **obtain Patient’s stated name and birth date** and **make sure that ID band and Patient’s stated name and birth date match** are allowed to occur again, zero or more times, and there are no restrictions on the occurrences of any future events.

FSA and DNL for Property BT.C.2.4a (C.2.2 → C.2.3)
Event alphabet:
- A: suspect Patient is having a transfusion reaction
- B: resume OR discontinue the infusion of a unit of blood product
- C: confirm presence of ID band, make sure that ID band and Patient’s stated name and birth date match, obtain Patient’s stated name and birth date

**SCOPE:**
1. From the start of any event sequence through to the end of that event sequence, the behavior must hold.

**BEHAVIOR:**
1. The events of primary interest in this behavior are suspect Patient is having a transfusion reaction and resume OR discontinue the infusion of a unit of blood product.
2. The events of secondary interest in this behavior are confirm presence of ID band, make sure that ID band and Patient’s stated name and birth date match, and obtain Patient’s stated name and birth date.
3. If suspect Patient is having a transfusion reaction occurs, resume OR discontinue the infusion of a unit of blood product is required to occur subsequently.
4. Before the first suspect Patient is having a transfusion reaction occurs:
   - resume OR discontinue the infusion of a unit of blood product is allowed to occur zero or more times;
   - All the events of secondary interest are allowed to occur zero or more times.
5. suspect Patient is having a transfusion reaction is not required to occur.
6. After suspect Patient is having a transfusion reaction occurs, but before the first subsequent resume OR discontinue the infusion of a unit of blood product occurs:
   - suspect Patient is having a transfusion reaction is not allowed to occur again;
   - All the events of secondary interest are allowed to occur zero or more times.
7. After suspect Patient is having a transfusion reaction and the first subsequent resume OR discontinue the infusion of a unit of blood product occur:
   - All the events of secondary interest are allowed to occur zero or more times;
   - resume OR discontinue the infusion of a unit of blood product is allowed to occur again, zero or more times, before another suspect Patient is having a transfusion reaction occurs; suspect Patient is having a transfusion reaction is allowed to occur again and, if it does, then the situation is the same as when the first suspect Patient is having a transfusion reaction occurred, meaning that the restrictions described in parts 3, 6, and 7 would again apply.

FSA and DNL for Property BT.C.2.4b
Event alphabet:
- A: suspect Patient is having a transfusion reaction
- B: make sure that ID band and tag affixed to unit of blood product i match

SCOPE:
1. From the start of any event sequence through to the end of that event sequence, the behavior must hold.

BEHAVIOR:
1. The events of primary interest in this behavior are suspect Patient is having a transfusion reaction and make sure that ID band and tag affixed to unit of blood product i match.
2. There are no events of secondary interest in this behavior.
3. If suspect Patient is having a transfusion reaction occurs, make sure that ID band and tag affixed to unit of blood product i match is required to occur subsequently.
4. Before the first suspect Patient is having a transfusion reaction occurs, make sure that ID band and tag affixed to unit of blood product i match is allowed to occur zero or more times.
5. suspect Patient is having a transfusion reaction is not required to occur.
6. After suspect Patient is having a transfusion reaction occurs, but before the first subsequent make sure that ID band and tag affixed to unit of blood product i match occurs, suspect Patient is having a transfusion reaction is not allowed to occur again.
7. After suspect Patient is having a transfusion reaction and the first subsequent make sure that ID band and tag affixed to unit of blood product i match occur:
   - make sure that ID band and tag affixed to unit of blood product i match is allowed to occur again, zero or more times, before another suspect Patient is having a transfusion reaction occurs; suspect Patient is having a transfusion reaction is allowed to occur again and, if it does, then the situation is the same as when the first suspect Patient is having a transfusion reaction occurred, meaning that the restrictions described in parts 3, 6, and 7 would again apply.

FSA and DNL for Property BT.C.3.1
**Event alphabet:**
- A: suspect Patient is having a transfusion reaction
- B: make sure that tag affixed to unit of blood product i and unit of blood product i match

**SCOPE:**
1. From the start of any event sequence through to the end of that event sequence, the behavior must hold.

**BEHAVIOR:**
1. The events of primary interest in this behavior are **suspect Patient is having a transfusion reaction** and **make sure that tag affixed to unit of blood product i and unit of blood product i match**.
2. There are no events of secondary interest in this behavior.
3. If **suspect Patient is having a transfusion reaction** occurs, **make sure that tag affixed to unit of blood product i and unit of blood product i match** is required to occur subsequently.
4. Before the first **suspect Patient is having a transfusion reaction** occurs, **make sure that tag affixed to unit of blood product i and unit of blood product i match** is allowed to occur zero or more times.
5. **suspect Patient is having a transfusion reaction** is not required to occur.
6. After **suspect Patient is having a transfusion reaction** occurs, but before the first subsequent **make sure that tag affixed to unit of blood product i and unit of blood product i match** occurs, **suspect Patient is having a transfusion reaction** is not allowed to occur again.
7. After **suspect Patient is having a transfusion reaction** and the first subsequent **make sure that tag affixed to unit of blood product i and unit of blood product i match** occur:
   - **make sure that tag affixed to unit of blood product i and unit of blood product i match** is allowed to occur again, zero or more times, before another **suspect Patient is having a transfusion reaction** occurs; **suspect Patient is having a transfusion reaction** is allowed to occur again and, if it does, then the situation is the same as when the first **suspect Patient is having a transfusion reaction** occurred, meaning that the restrictions described in parts 3, 6, and 7 would again apply.
Event alphabet:
- A: suspect Patient is having a transfusion reaction
- B: make sure that unit of blood product i has not expired

Scope:
1. From the start of any event sequence through to the end of that event sequence, the behavior must hold.

Behavior:
1. The events of primary interest in this behavior are suspect Patient is having a transfusion reaction and make sure that unit of blood product i has not expired.
2. There are no events of secondary interest in this behavior.
3. If suspect Patient is having a transfusion reaction occurs, make sure that unit of blood product i has not expired is required to occur subsequently.
4. Before the first suspect Patient is having a transfusion reaction occurs, make sure that unit of blood product i has not expired is allowed to occur zero or more times.
5. suspect Patient is having a transfusion reaction is not required to occur.
6. After suspect Patient is having a transfusion reaction occurs, but before the first subsequent make sure that unit of blood product i has not expired occurs, suspect Patient is having a transfusion reaction is not allowed to occur again.
7. After suspect Patient is having a transfusion reaction and the first subsequent make sure that unit of blood product i has not expired occur:
   - make sure that unit of blood product i has not expired is allowed to occur again, zero or more times, before another suspect Patient is having a transfusion reaction occurs; suspect Patient is having a transfusion reaction is allowed to occur again and, if it does, then the situation is the same as when the first suspect Patient is having a transfusion reaction occurred, meaning that the restrictions described in parts 3, 6, and 7 would again apply.

FSA and DNL for Property BT.C.3.3
Event alphabet:

- A: make sure that ID band and tag affixed to unit of blood product i match
- B: make sure that tag affixed to unit of blood product i and unit of blood product i match
- END: resume OR discontinue the infusion of unit of blood product i
- C: make sure that unit of blood product i has not expired

**SCOPE:**

1. A restricted interval in the event sequence can have both a starting delimiter, suspect Patient is having a transfusion reaction, and an ending delimiter, resume OR discontinue the infusion of unit of blood product i.
2. The behavior is required to hold from an occurrence of suspect Patient is having a transfusion reaction, if it ever occurs, through to the first subsequent occurrence of resume OR discontinue the infusion of unit of blood product i, if it ever occurs.

3. If there are multiple occurrences of suspect Patient is having a transfusion reaction without an occurrence of resume OR discontinue the infusion of unit of blood product i in between them, only the first of those occurrences of suspect Patient is having a transfusion reaction starts a restricted interval; later occurrences of suspect Patient is having a transfusion reaction within this restricted interval do not have an effect.

4. If suspect Patient is having a transfusion reaction is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if suspect Patient is having a transfusion reaction does occur, resume OR discontinue the infusion of unit of blood product i is not required to occur subsequently. Even if resume OR discontinue the infusion of unit of blood product i does not occur subsequently, the behavior is still required to hold, until the end of the event sequence.

5. There might be many restricted intervals in the event sequence. If suspect Patient is having a transfusion reaction occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of suspect Patient is having a transfusion reaction, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a new restricted interval.

6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

**BEHAVIOR:**

1. The events of primary interest in this behavior are make sure that ID band and tag affixed to unit of blood product i match and make sure that tag affixed to unit of blood product i and unit of blood product i match.
2. The event of secondary interest in this behavior is make sure that unit of blood product i has not expired.
3. make sure that tag affixed to unit of blood product i and unit of blood product i match is not allowed to occur until after make sure that ID band and tag affixed to unit of blood product i match occurs.
4. Before the first make sure that ID band and tag affixed to unit of blood product i match occurs, make sure that unit of blood product i has not expired is allowed to occur zero or more times.
5. make sure that ID band and tag affixed to unit of blood product i match is required to occur, whether or not make sure that tag affixed to unit of blood product i and unit of blood product i match eventually occurs.
6. make sure that tag affixed to unit of blood product i and unit of blood product i match is not required to occur after make sure that ID band and tag affixed to unit of blood product i match occurs.
7. After make sure that ID band and tag affixed to unit of blood product i match occurs, but before the first subsequent make sure that tag affixed to unit of blood product i and unit of blood product i match occurs:
   - make sure that ID band and tag affixed to unit of blood product i match is allowed to occur again, zero or more times;
   - make sure that unit of blood product i has not expired is not allowed to occur.
8. After make sure that ID band and tag affixed to unit of blood product i match and the first subsequent make sure that tag affixed to unit of blood product i and unit of blood product i match occurs:
   - make sure that ID band and tag affixed to unit of blood product i match is allowed to occur zero or more times;
   - both make sure that ID band and tag affixed to unit of blood product i match and make sure that tag affixed to unit of blood product i and unit of blood product i match are allowed to occur again, zero or more times, and there are no restrictions on the occurrences of any future events.
7. After make sure that tag affixed to unit of blood product i and unit of blood product i match
6. make sure that unit of blood product i has not expired
5. There might be many restricted intervals in the event sequence. If suspect Patient is having a transfusion reaction does occur, resume OR discontinue the infusion of unit of blood product i does not occur subsequently, the behavior is still required to hold, until the end of the event sequence.
4. Before the first make sure that tag affixed to unit of blood product i and unit of blood product i match occurs, make sure that ID band and tag affixed to unit of blood product i match is allowed to occur zero or more times.
3. If there are multiple occurrences of suspect Patient is having a transfusion reaction without an occurrence of resume OR discontinue the infusion of unit of blood product i in between them, only the first of those occurrences of suspect Patient is having a transfusion reaction starts a restricted interval; later occurrences of suspect Patient is having a transfusion reaction within this restricted interval do not have an effect.
2. The behavior is required to hold from an occurrence of suspect Patient is having a transfusion reaction, if it ever occurs, through to the first subsequent occurrence of resume OR discontinue the infusion of unit of blood product i, if it ever occurs.
1. A restricted interval in the event sequence can have both a starting delimiter, suspect Patient is having a transfusion reaction, and an ending delimiter, resume OR discontinue the infusion of unit of blood product i

SCOPE:
1. The events of primary interest in this behavior are make sure that tag affixed to unit of blood product i and unit of blood product i match and make sure that unit of blood product i has not expired.
2. The event of secondary interest in this behavior is make sure that ID band and tag affixed to unit of blood product i match.
3. make sure that unit of blood product i has not expired is not allowed to occur until after make sure that tag affixed to unit of blood product i and unit of blood product i match occurs.
4. Before the first make sure that tag affixed to unit of blood product i and unit of blood product i match occurs, make sure that ID band and tag affixed to unit of blood product i match is allowed to occur zero or more times.
5. make sure that tag affixed to unit of blood product i and unit of blood product i match is required to occur, whether or not make sure that unit of blood product i has not expired eventually occurs.
6. make sure that unit of blood product i has not expired is not required to occur after make sure that tag affixed to unit of blood product i and unit of blood product i match occurs.
7. After make sure that tag affixed to unit of blood product i and unit of blood product i match occurs, but before the first subsequent make sure that unit of blood product i has not expired occurs:
   - make sure that tag affixed to unit of blood product i and unit of blood product i match is allowed to occur again, zero or more times;
   - make sure that ID band and tag affixed to unit of blood product i match is not allowed to occur.
8. After make sure that tag affixed to unit of blood product i and unit of blood product i match and the first subsequent make sure that unit of blood product i has not expired occurs:
   - make sure that ID band and tag affixed to unit of blood product i match is allowed to occur zero or more times;
   - Both make sure that tag affixed to unit of blood product i and unit of blood product i match and make sure that unit of blood product i has not expired are allowed to occur again, zero or more times, and there are no restrictions on the occurrences of any future events.

BEHAVIOR:
1. The events of primary interest in this behavior are make sure that tag affixed to unit of blood product i and unit of blood product i match and make sure that unit of blood product i has not expired.
2. The event of secondary interest in this behavior is make sure that ID band and tag affixed to unit of blood product i match.
3. make sure that unit of blood product i has not expired is not allowed to occur until after make sure that tag affixed to unit of blood product i and unit of blood product i match occurs.
4. Before the first make sure that tag affixed to unit of blood product i and unit of blood product i match occurs, make sure that ID band and tag affixed to unit of blood product i match is allowed to occur zero or more times.
5. make sure that tag affixed to unit of blood product i and unit of blood product i match is required to occur, whether or not make sure that unit of blood product i has not expired eventually occurs.
6. make sure that unit of blood product i has not expired is not required to occur after make sure that tag affixed to unit of blood product i and unit of blood product i match occurs.
7. After make sure that tag affixed to unit of blood product i and unit of blood product i match occurs, but before the first subsequent make sure that unit of blood product i has not expired occurs:
   - make sure that tag affixed to unit of blood product i and unit of blood product i match is allowed to occur again, zero or more times;
   - make sure that ID band and tag affixed to unit of blood product i match is not allowed to occur.
8. After make sure that tag affixed to unit of blood product i and unit of blood product i match and the first subsequent make sure that unit of blood product i has not expired occurs:
   - make sure that ID band and tag affixed to unit of blood product i match is allowed to occur zero or more times;
   - Both make sure that tag affixed to unit of blood product i and unit of blood product i match and make sure that unit of blood product i has not expired are allowed to occur again, zero or more times, and there are no restrictions on the occurrences of any future events.

FSA and DNL for Property BT.C.3.4 (C.3.2 → C.3.3)
Event alphabet:
- A: suspect Patient is having a transfusion reaction
- B: ask physician and blood bank for further instructions

SCOPE:
1. From the start of any event sequence through to the end of that event sequence, the behavior must hold.

BEHAVIOR:
1. The events of primary interest in this behavior are suspect Patient is having a transfusion reaction and ask physician and blood bank for further instructions.
2. There are no events of secondary interest in this behavior.
3. If suspect Patient is having a transfusion reaction occurs, ask physician and blood bank for further instructions is required to occur subsequently.
4. Before the first suspect Patient is having a transfusion reaction occurs, ask physician and blood bank for further instructions is allowed to occur zero or more times.
5. suspect Patient is having a transfusion reaction is not required to occur.
6. After suspect Patient is having a transfusion reaction occurs, but before the first subsequent ask physician and blood bank for further instructions occurs, suspect Patient is having a transfusion reaction is not allowed to occur again.
7. After suspect Patient is having a transfusion reaction and the first subsequent ask physician and blood bank for further instructions occur:
   - ask physician and blood bank for further instructions is allowed to occur again, zero or more times, before another suspect Patient is having a transfusion reaction occurs; suspect Patient is having a transfusion reaction is allowed to occur again and, if it does, then the situation is the same as when the first suspect Patient is having a transfusion reaction occurred, meaning that the restrictions described in parts 3, 6, and 7 would again apply.

FSA and DNL for Property BT.C.4
Event alphabet:

- A: stop the infusion of a unit of blood product i
- B: confirm presence of ID band
- START: suspect Patient is having a transfusion reaction
- END: resume OR discontinue the infusion of unit of blood product i
- C: ask physician and blood bank for further instructions, make sure ID band and Patient’s stated name and birth date match, make sure that ID band and tag affixed to the unit of blood product i match, make sure that unit of blood product i has not expired

SCOPE:

1. A restricted interval in the event sequence can have both a starting delimiter, **suspect Patient is having a transfusion reaction**, and an ending delimiter, **resume OR discontinue the infusion of unit of blood product i**.
2. The behavior is required to hold from an occurrence of **suspect Patient is having a transfusion reaction**, if it ever occurs, through to the first subsequent occurrence of **resume OR discontinue the infusion of unit of blood product i**, if it ever occurs.
3. If there are multiple occurrences of **suspect Patient is having a transfusion reaction** without an occurrence of **resume OR discontinue the infusion of unit of blood product i** in between them, only the first of those occurrences of **suspect Patient is having a transfusion reaction** starts a restricted interval; later occurrences of **suspect Patient is having a transfusion reaction** within this restricted interval do not have an effect.
4. **suspect Patient is having a transfusion reaction** is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if **suspect Patient is having a transfusion reaction** does occur, **resume OR discontinue the infusion of unit of blood product i** is not required to occur subsequently. Even if **resume OR discontinue the infusion of unit of blood product i** does not occur subsequently, the behavior is still required to hold, until the end of the event sequence.
5. There might be many restricted intervals in the event sequence. If **suspect Patient is having a transfusion reaction** occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of **suspect Patient is having a transfusion reaction**, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

BEHAVIOR:

1. The events of interest in this behavior are **stop the infusion of a unit of blood product i** and **confirm presence of ID band**.
2. The events of secondary interest in this behavior are ask physician and blood bank for further instructions, make sure ID band and Patient’s stated name and birth date match, make sure that ID band and tag affixed to the unit of blood product i match, and make sure that unit of blood product i has not expired.
3. **confirm presence of ID band** is not allowed to occur until after **stop the infusion of a unit of blood product i** occurs.
4. Before the first **stop the infusion of a unit of blood product i** occurs, all the events of secondary interest are allowed to occur zero or more times.
5. **Stop the infusion of a unit of blood product i** is required to occur, whether or not **confirm presence of ID band** eventually occurs.
6. **confirm presence of ID band** is not required to occur after **stop the infusion of a unit of blood product i** occurs.
7. After **stop the infusion of a unit of blood product i** occurs, but before the first subsequent **confirm presence of ID band** occurs:
   - **Stop the infusion of a unit of blood product i** is not allowed to occur again;
   - None of the events of secondary interest are allowed to occur.
8. After **stop the infusion of a unit of blood product i** and the first subsequent **confirm presence of ID band** occur:
   - All the events of secondary interest are allowed to occur zero or more times;
   - Both **stop the infusion of a unit of blood product i** and **confirm presence of ID band** are allowed to occur again, zero or more times, and there are no restrictions on the occurrences of any future events.

FSA and DNL for Property BT.C.5 (C.1 → C.2.1)
Event alphabet:
- A: make sure ID band and Patient’s stated name and birth date match
- B: make sure that ID band and tag affixed to the unit of blood product i match
- END: resume OR discontinue the infusion of unit of blood product i
- C: ask physician and blood bank for further instructions, confirm presence of ID band, make sure that unit of blood product i has not expired, stop the infusion of a unit of blood product i

SCOPE:
1. A restricted interval in the event sequence can have both a starting delimiter, suspect Patient is having a transfusion reaction, and an ending delimiter, resume OR discontinue the infusion of unit of blood product i.
2. The behavior is required to hold from an occurrence of suspect Patient is having a transfusion reaction, if it ever occurs, through to the first subsequent occurrence of resume OR discontinue the infusion of unit of blood product i, if it ever occurs.
3. If there are multiple occurrences of suspect Patient is having a transfusion reaction without an occurrence of resume OR discontinue the infusion of unit of blood product i in between, only the first of those occurrences of suspect Patient is having a transfusion reaction starts a restricted interval; later occurrences of suspect Patient is having a transfusion reaction within this restricted interval do not have an effect.
4. suspect Patient is having a transfusion reaction is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if suspect Patient is having a transfusion reaction occurs, resume OR discontinue the infusion of unit of blood product i is not required to occur subsequently. Even if resume OR discontinue the infusion of unit of blood product i does not occur subsequently, the behavior is still required to hold, until the end of the event sequence.
5. There might be many restricted intervals in the event sequence. If suspect Patient is having a transfusion reaction occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of suspect Patient is having a transfusion reaction, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

BEHAVIOR:
1. The events of primary interest in this behavior are make sure ID band and Patient’s stated name and birth date match and make sure that ID band and tag affixed to the unit of blood product i match.
2. The events of secondary interest in this behavior are ask physician and blood bank for further instructions, confirm presence of ID band, make sure that unit of blood product i has not expired, and stop the infusion of a unit of blood product i.
3. make sure that ID band and tag affixed to the unit of blood product i match is not allowed to occur until after make sure ID band and Patient’s stated name and birth date match occurs.
4. Before the first make sure ID band and Patient’s stated name and birth date match occurs, all the events of secondary interest are allowed to occur zero or more times.
5. make sure ID band and Patient’s stated name and birth date match is required to occur, whether or not make sure that ID band and tag affixed to the unit of blood product i match occurs.
6. make sure that ID band and tag affixed to the unit of blood product i match is not required to occur after make sure ID band and Patient’s stated name and birth date match occurs.
7. After make sure ID band and Patient’s stated name and birth date match occurs, but before the first subsequent make sure that ID band and tag affixed to the unit of blood product i match occurs:
   • make sure ID band and Patient’s stated name and birth date match is allowed to occur again, zero or more times;
   • None of the events of secondary interest are allowed to occur.
8. After make sure ID band and Patient’s stated name and birth date match and the first subsequent make sure that ID band and tag affixed to the unit of blood product i match occur:
   • All the events of secondary interest are allowed to occur zero or more times;
   • Both make sure ID band and Patient’s stated name and birth date match and make sure that ID band and tag affixed to the unit of blood product i match are allowed to occur again, zero or more times, and there are no restrictions on the occurrences of any future events.

FSA and DNL for Property BT.C.5 (C.2.3 → C.3.1)
make sure that unit of blood product i has not expired
7. After the event occurs, all the events of secondary interest are allowed to occur zero or more times.

ask physician and blood bank for further instructions
3. If there are multiple occurrences of suspect Patient is having a transfusion reaction starts a restricted interval; later occurrences of suspect Patient is having a transfusion reaction within this restricted interval do not have an effect.

ask physician and blood bank for further instructions
4. The behavior is still required to hold, until the end of the event sequence.

suspect Patient is having a transfusion reaction
5. There might be many restricted intervals in the event sequence. If suspect Patient is having a transfusion reaction occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of suspect Patient is having a transfusion reaction, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a new restricted interval.

BEHAVIOR:
1. The events of primary interest in this behavior are make sure that unit of blood product i has not expired and ask physician and blood bank for further instructions.

make sure that unit of blood product i has not expired
2. The events of secondary interest in this behavior are confirm presence of ID band, make sure ID band and Patient’s stated name and birth date match, make sure that ID band and tag affixed to the unit of blood product i match, and stop the infusion of a unit of blood product i.

ask physician and blood bank for further instructions
3. If it ever occurs.

suspect Patient is having a transfusion reaction
4. The behavior is not required to occur subsequently. Even if resume OR discontinue the infusion of unit of blood product i does not occur subsequently, the behavior is still required to hold, until the end of the event sequence.

ask physician and blood bank for further instructions
5. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

BEHAVIOR:
1. The events of primary interest in this behavior are make sure that unit of blood product i has not expired and ask physician and blood bank for further instructions.

make sure that unit of blood product i has not expired
2. The events of secondary interest in this behavior are confirm presence of ID band, make sure ID band and Patient’s stated name and birth date match, make sure that ID band and tag affixed to the unit of blood product i match, and stop the infusion of a unit of blood product i.

ask physician and blood bank for further instructions
3. If it ever occurs.

suspect Patient is having a transfusion reaction
4. The behavior is not required to occur subsequently. Even if resume OR discontinue the infusion of unit of blood product i does not occur subsequently, the behavior is still required to hold, until the end of the event sequence.

suspect Patient is having a transfusion reaction
5. There might be many restricted intervals in the event sequence. If suspect Patient is having a transfusion reaction occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of suspect Patient is having a transfusion reaction, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a new restricted interval.

BEHAVIOR:
1. The events of primary interest in this behavior are make sure that unit of blood product i has not expired and ask physician and blood bank for further instructions.

make sure that unit of blood product i has not expired
2. The events of secondary interest in this behavior are confirm presence of ID band, make sure ID band and Patient’s stated name and birth date match, make sure that ID band and tag affixed to the unit of blood product i match, and stop the infusion of a unit of blood product i.

ask physician and blood bank for further instructions
3. If it ever occurs.

suspect Patient is having a transfusion reaction
4. The behavior is not required to occur subsequently. Even if resume OR discontinue the infusion of unit of blood product i does not occur subsequently, the behavior is still required to hold, until the end of the event sequence.

ask physician and blood bank for further instructions
5. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

BEHAVIOR:
1. The events of primary interest in this behavior are make sure that unit of blood product i has not expired and ask physician and blood bank for further instructions.

make sure that unit of blood product i has not expired
2. The events of secondary interest in this behavior are confirm presence of ID band, make sure ID band and Patient’s stated name and birth date match, make sure that ID band and tag affixed to the unit of blood product i match, and stop the infusion of a unit of blood product i.

ask physician and blood bank for further instructions
3. If it ever occurs.

suspect Patient is having a transfusion reaction
4. The behavior is not required to occur subsequently. Even if resume OR discontinue the infusion of unit of blood product i does not occur subsequently, the behavior is still required to hold, until the end of the event sequence.

suspect Patient is having a transfusion reaction
5. There might be many restricted intervals in the event sequence. If suspect Patient is having a transfusion reaction occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of suspect Patient is having a transfusion reaction, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a new restricted interval.

BEHAVIOR:
1. The events of primary interest in this behavior are make sure that unit of blood product i has not expired and ask physician and blood bank for further instructions.

make sure that unit of blood product i has not expired
2. The events of secondary interest in this behavior are confirm presence of ID band, make sure ID band and Patient’s stated name and birth date match, make sure that ID band and tag affixed to the unit of blood product i match, and stop the infusion of a unit of blood product i.

ask physician and blood bank for further instructions
3. If it ever occurs.

suspect Patient is having a transfusion reaction
4. The behavior is not required to occur subsequently. Even if resume OR discontinue the infusion of unit of blood product i does not occur subsequently, the behavior is still required to hold, until the end of the event sequence.

suspect Patient is having a transfusion reaction
5. There might be many restricted intervals in the event sequence. If suspect Patient is having a transfusion reaction occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of suspect Patient is having a transfusion reaction, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a new restricted interval.

BEHAVIOR:
1. The events of primary interest in this behavior are make sure that unit of blood product i has not expired and ask physician and blood bank for further instructions.

make sure that unit of blood product i has not expired
2. The events of secondary interest in this behavior are confirm presence of ID band, make sure ID band and Patient’s stated name and birth date match, make sure that ID band and tag affixed to the unit of blood product i match, and stop the infusion of a unit of blood product i.

ask physician and blood bank for further instructions
3. If it ever occurs.

suspect Patient is having a transfusion reaction
4. The behavior is not required to occur subsequently. Even if resume OR discontinue the infusion of unit of blood product i does not occur subsequently, the behavior is still required to hold, until the end of the event sequence.

suspect Patient is having a transfusion reaction
5. There might be many restricted intervals in the event sequence. If suspect Patient is having a transfusion reaction occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of suspect Patient is having a transfusion reaction, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a new restricted interval.

suspect Patient is having a transfusion reaction
5. There might be many restricted intervals in the event sequence. If suspect Patient is having a transfusion reaction occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of suspect Patient is having a transfusion reaction, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a new restricted interval.

BEHAVIOR:
1. The events of primary interest in this behavior are make sure that unit of blood product i has not expired and ask physician and blood bank for further instructions.

make sure that unit of blood product i has not expired
2. The events of secondary interest in this behavior are confirm presence of ID band, make sure ID band and Patient’s stated name and birth date match, make sure that ID band and tag affixed to the unit of blood product i match, and stop the infusion of a unit of blood product i.

ask physician and blood bank for further instructions
3. If it ever occurs.

suspect Patient is having a transfusion reaction
4. The behavior is not required to occur subsequently. Even if resume OR discontinue the infusion of unit of blood product i does not occur subsequently, the behavior is still required to hold, until the end of the event sequence.

suspect Patient is having a transfusion reaction
5. There might be many restricted intervals in the event sequence. If suspect Patient is having a transfusion reaction occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of suspect Patient is having a transfusion reaction, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a new restricted interval.

suspect Patient is having a transfusion reaction
5. There might be many restricted intervals in the event sequence. If suspect Patient is having a transfusion reaction occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of suspect Patient is having a transfusion reaction, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a new restricted interval.

suspect Patient is having a transfusion reaction
5. There might be many restricted intervals in the event sequence. If suspect Patient is having a transfusion reaction occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of suspect Patient is having a transfusion reaction, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a new restricted interval.
**Event alphabet:**

- A: receive blood bank orders to obtain specimen
- B: obtain specimen
- START: stop the infusion of a unit of blood product because of suspected transfusion reaction
- END: record infusion information

**SCOPE:**

1. A restricted interval in the event sequence can have both a starting delimiter, stop the infusion of a unit of blood product because of suspected transfusion reaction, and an ending delimiter, record infusion information.
2. The behavior is required to hold from an occurrence of stop the infusion of a unit of blood product because of suspected transfusion reaction, if it ever occurs, through to the first subsequent occurrence of record infusion information, if it ever occurs.
3. If there are multiple occurrences of stop the infusion of a unit of blood product because of suspected transfusion reaction without an occurrence of record infusion information in between them, only the first of those occurrences of stop the infusion of a unit of blood product because of suspected transfusion reaction starts a restricted interval; later occurrences of stop the infusion of a unit of blood product because of suspected transfusion reaction do not start a new restricted interval if they occur without an occurrence of record infusion information in between them.
4. stop the infusion of a unit of blood product because of suspected transfusion reaction is not required to occur if it never occurs. Even if stop the infusion of a unit of blood product because of suspected transfusion reaction occurs, record infusion information is not required to occur subsequently. Even if record infusion information does not occur subsequently, the behavior is still required to hold, until the end of the event sequence.
5. There might be many restricted intervals in the event sequence. If stop the infusion of a unit of blood product because of suspected transfusion reaction occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of stop the infusion of a unit of blood product because of suspected transfusion reaction, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

**BEHAVIOR:**

1. The events of primary interest in this behavior are receive blood bank orders to obtain specimen and obtain specimen.
2. There are no events of secondary interest in this behavior.
3. If receive blood bank orders to obtain specimen occurs, obtain specimen is required to occur subsequently.
4. Before the first receive blood bank orders to obtain specimen occurs, obtain specimen is not allowed to occur.
5. receive blood bank orders to obtain specimen is not required to occur.
6. After receive blood bank orders to obtain specimen occurs, but before the first subsequent obtain specimen occurs, receive blood bank orders to obtain specimen is allowed to occur again, zero or more times.
7. After receive blood bank orders to obtain specimen and the first subsequent obtain specimen occur:
   - obtain specimen is allowed to occur again, zero or more times, before another receive blood bank orders to obtain specimen occurs; receive blood bank orders to obtain specimen is allowed to occur again and, if it does, then the situation is the same as when the first receive blood bank orders to obtain specimen occurred, meaning that the restrictions described in parts 3, 6, and 7 would again apply.

**FSA and DNL for Property BT.C.6a**
Event alphabet:
- A: stop the infusion of a unit of blood product because of suspected transfusion reaction
- B: record infusion information
- C: obtain specimen, receive blood bank orders to obtain specimen

**SCOPE:**
1. From the start of any event sequence through to the end of that event sequence, the behavior must hold.

**BEHAVIOR:**
1. The events of primary interest in this behavior are stop the infusion of a unit of blood product because of suspected transfusion reaction and record infusion information.
2. The events of secondary interest in this behavior are obtain specimen and receive blood bank orders to obtain specimen.
3. If stop the infusion of a unit of blood product because of suspected transfusion reaction occurs, record infusion information is required to occur subsequently.
4. Before the first stop the infusion of a unit of blood product because of suspected transfusion reaction occurs:
   - record infusion information is allowed to occur zero or more times;
   - All the events of secondary interest are allowed to occur zero or more times.
5. stop the infusion of a unit of blood product because of suspected transfusion reaction is not required to occur.
6. After stop the infusion of a unit of blood product because of suspected transfusion reaction occurs, but before the first subsequent record infusion information occurs:
   - stop the infusion of a unit of blood product because of suspected transfusion reaction is allowed to occur again, zero or more times;
   - All the events of secondary interest are allowed to occur zero or more times.
7. After stop the infusion of a unit of blood product because of suspected transfusion reaction and the first subsequent record infusion information occur:
   - All the events of secondary interest are allowed to occur zero or more times;
   - record infusion information is allowed to occur again, zero or more times, before another stop the infusion of a unit of blood product because of suspected transfusion reaction occurs; stop the infusion of a unit of blood product because of suspected transfusion reaction is allowed to occur again and, if it does, then the situation is the same as when the first stop the infusion of a unit of blood product because of suspected transfusion reaction occurred, meaning that the restrictions described in parts 3, 6, and 7 would again apply.

FSA and DNL for Property BT.C.6b
Event alphabet:

- **A**: confirm presence of ID band
- **B**: obtain specimen i
- **START**: receive order to obtain a specimen
- **END**: send specimen i to blood bank
- **C**: apply specimen container label i, make sure that ID band and order to obtain a specimen match, make sure that ID band and Patient’s stated name and birth date match, make sure that ID band and specimen container label i match, obtain Patient’s stated name and birth date

**SCOPE:**

1. A restricted interval in the event sequence can have both a starting delimiter, **receive order to obtain a specimen**, and an ending delimiter, **send specimen i to blood bank**.
2. The behavior is required to hold from an occurrence of **receive order to obtain a specimen**, if it ever occurs, through to the first subsequent occurrence of **send specimen i to blood bank**, if it ever occurs.
3. If there are multiple occurrences of **receive order to obtain a specimen** without an occurrence of **send specimen i to blood bank** in between them, only the first of those occurrences of **receive order to obtain a specimen** potentially starts a restricted interval; later occurrences of **receive order to obtain a specimen** within this restricted interval do not have an effect.
4. **receive order to obtain a specimen** is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if **receive order to obtain a specimen** does occur, **send specimen i to blood bank** is not required to occur subsequently and if it does not occur subsequently, then the behavior is not required to hold for the remainder of the event sequence.
5. There might be many restricted intervals in the event sequence. If **receive order to obtain a specimen** occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of **receive order to obtain a specimen**, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a potential new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

**BEHAVIOR:**

1. The events of primary interest in this behavior are **confirm presence of ID band** and **obtain specimen i**.
2. The events of secondary interest in this behavior are **apply specimen container label i**, make sure that ID band and order to obtain a specimen match, make sure that ID band and Patient’s stated name and birth date match, make sure that ID band and specimen container label i match, and obtain Patient’s stated name and birth date.
3. **obtain specimen i** is not allowed to occur until after **confirm presence of ID band** occurs.
4. Before the first **confirm presence of ID band** occurs, all the events of secondary interest are allowed to occur zero or more times.
5. **confirm presence of ID band** is required to occur, whether or not **obtain specimen i** eventually occurs.
6. **obtain specimen i** is not required to occur after **confirm presence of ID band** occurs.
7. After **confirm presence of ID band** occurs, but before the first subsequent **obtain specimen i** occurs:
   - **confirm presence of ID band** is allowed to occur again, zero or more times;
   - All the events of secondary interest are allowed to occur zero or more times.
8. After **confirm presence of ID band** and the first subsequent **obtain specimen i** occur:
   - All the events of secondary interest are allowed to occur zero or more times;
   - Both **confirm presence of ID band** and **obtain specimen i** are allowed to occur again, zero or more times, and there are no restrictions on the occurrences of any future events.

FSA and DNL for Property BT.D.1a
Event alphabet:
- A: confirm presence of ID band
- B: apply specimen container label i
- START: receive order to obtain a specimen
- END: send specimen i to blood bank
- C: make sure that ID band and order to obtain a specimen match, make sure that ID band and Patient’s stated name and birth date match, make sure that ID band and specimen container label i match, obtain Patient’s stated name and birth date, obtain specimen i

SCOPE:
1. A restricted interval in the event sequence can have both a starting delimiter, receive order to obtain a specimen, and an ending delimiter, send specimen i to blood bank.
2. The behavior is required to hold from an occurrence of receive order to obtain a specimen, if it ever occurs, through to the first subsequent occurrence of send specimen i to blood bank, if it ever occurs.
3. If there are multiple occurrences of receive order to obtain a specimen without an occurrence of send specimen i to blood bank in between them, only the first of those occurrences of receive order to obtain a specimen potentially starts a restricted interval; later occurrences of receive order to obtain a specimen within this restricted interval do not have an effect.
4. receive order to obtain a specimen is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if receive order to obtain a specimen does occur, send specimen i to blood bank is not required to occur subsequently and if it does not occur subsequently, then the behavior is not required to hold for the remainder of the event sequence.
5. There might be many restricted intervals in the event sequence. If receive order to obtain a specimen occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of receive order to obtain a specimen, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a potential new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

BEHAVIOR:
1. The events of primary interest in this behavior are confirm presence of ID band and apply specimen container label i.
2. The events of secondary interest in this behavior are make sure that ID band and order to obtain a specimen match, make sure that ID band and Patient’s stated name and birth date match, make sure that ID band and specimen container label i match, obtain Patient’s stated name and birth date, and obtain specimen i.
3. apply specimen container label i is not allowed to occur until after confirm presence of ID band occurs.
4. Before the first confirm presence of ID band occurs, all the events of secondary interest are allowed to occur zero or more times.
5. confirm presence of ID band is required to occur, whether or not apply specimen container label i eventually occurs.
6. apply specimen container label i is not required to occur after confirm presence of ID band occurs.
7. After confirm presence of ID band occurs, but before the first subsequent apply specimen container label i occurs:
   - confirm presence of ID band is allowed to occur again, zero or more times;
   - All the events of secondary interest are allowed to occur zero or more times.
8. After confirm presence of ID band and the first subsequent apply specimen container label i occur:
   - All the events of secondary interest are allowed to occur zero or more times,
   - Both confirm presence of ID band and apply specimen container label i are allowed to occur again, zero or more times, and there are no restrictions on the occurrences of any future events.

FSA and DNL for Property BT.D.1b
Event alphabet:

- **A**: obtain Patient’s stated name and birth date
- **B**: obtain specimen i
- **START**: receive order to obtain a specimen
- **END**: send specimen i to blood bank
- **C**: apply specimen container label i, confirm presence of ID band, make sure that ID band and order to obtain a specimen match, make sure that ID band and Patient’s stated name and birth date match, make sure that ID band and specimen container label i match

**SCOPE:**
1. A restricted interval in the event sequence can have both a starting delimiter, receive order to obtain a specimen, and an ending delimiter, send specimen i to blood bank.
2. The behavior is required to hold from an occurrence of receive order to obtain a specimen, if it ever occurs, through to the first subsequent occurrence of send specimen i to blood bank, if it ever occurs.
3. If there are multiple occurrences of receive order to obtain a specimen without an occurrence of send specimen i to blood bank in between them, only the first of those occurrences of receive order to obtain a specimen potentially starts a restricted interval; later occurrences of receive order to obtain a specimen within this restricted interval do not have an effect.
4. receive order to obtain a specimen is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if receive order to obtain a specimen does occur, send specimen i to blood bank is not required to occur subsequently and if it does not occur subsequently, then the behavior is not required to hold for the remainder of the event sequence.
5. There might be many restricted intervals in the event sequence. If receive order to obtain a specimen occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of receive order to obtain a specimen, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a potential new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

**BEHAVIOR:**
1. The events of primary interest in this behavior are obtain Patient’s stated name and birth date and obtain specimen i.
2. The events of secondary interest in this behavior are apply specimen container label i, confirm presence of ID band, make sure that ID band and order to obtain a specimen match, make sure that ID band and Patient’s stated name and birth date match, and make sure that ID band and specimen container label i match.
3. obtain specimen i is not allowed to occur until after obtain Patient’s stated name and birth date occurs.
4. Before the first obtain Patient’s stated name and birth date occurs, all the events of secondary interest are allowed to occur zero or more times.
5. obtain Patient’s stated name and birth date is required to occur, whether or not obtain specimen i eventually occurs.
6. obtain specimen i is not required to occur after obtain Patient’s stated name and birth date occurs.
7. After obtain Patient’s stated name and birth date occurs, but before the first subsequent obtain specimen i occurs:
   - obtain Patient’s stated name and birth date is allowed to occur again, zero or more times;
   - All the events of secondary interest are allowed to occur zero or more times;
8. After obtain Patient’s stated name and birth date and the first subsequent obtain specimen i occur:
   - All the events of secondary interest are allowed to occur zero or more times;
   - Both obtain Patient’s stated name and birth date and obtain specimen i are allowed to occur again, zero or more times, and there are no restrictions on the occurrences of any future events.

**FSA and DNL for Property BT.D.2a**
Event alphabet:
- A: obtain Patient’s stated name and birth date
- B: apply specimen container label i
- START: receive order to obtain a specimen
- END: send specimen i to blood bank
- C: confirm presence of ID band, make sure that ID band and order to obtain a specimen match, make sure that ID band and Patient’s stated name and birth date match, make sure that ID band and specimen container label i match, obtain specimen i

SCOPE:
1. A restricted interval in the event sequence can have both a starting delimiter, receive order to obtain a specimen, and an ending delimiter, send specimen i to blood bank.
2. The behavior is required to hold from an occurrence of receive order to obtain a specimen, if it ever occurs, through to the first subsequent occurrence of send specimen i to blood bank, if it ever occurs.
3. If there are multiple occurrences of receive order to obtain a specimen without an occurrence of send specimen i to blood bank in between them, only the first of those occurrences of receive order to obtain a specimen potentially starts a restricted interval; later occurrences of receive order to obtain a specimen within this restricted interval do not have an effect.
4. receive order to obtain a specimen is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if receive order to obtain a specimen does occur, send specimen i to blood bank is not required to occur subsequently and if it does not occur subsequently, then the behavior is not required to hold for the remainder of the event sequence.
5. There might be many restricted intervals in the event sequence. If receive order to obtain a specimen occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of receive order to obtain a specimen, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a potential new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

BEHAVIOR:
1. The events of primary interest in this behavior are obtain Patient’s stated name and birth date and apply specimen container label i.
2. The events of secondary interest in this behavior are confirm presence of ID band, make sure that ID band and order to obtain a specimen match, make sure that ID band and Patient’s stated name and birth date match, make sure that ID band and specimen container label i match, and obtain specimen i.
3. apply specimen container label i is not allowed to occur until after obtain Patient’s stated name and birth date occurs.
4. Before the first obtain Patient’s stated name and birth date occurs, all the events of secondary interest are allowed to occur zero or more times.
5. obtain Patient’s stated name and birth date is required to occur, whether or not apply specimen container label i eventually occurs.
6. apply specimen container label i is not required to occur after obtain Patient’s stated name and birth date occurs.
7. After obtain Patient’s stated name and birth date occurs, but before the first subsequent apply specimen container label i occurs:
   - obtain Patient’s stated name and birth date is allowed to occur again, zero or more times;
   - All the events of secondary interest are allowed to occur zero or more times.
8. After obtain Patient’s stated name and birth date and the first subsequent apply specimen container label i occur:
   - All the events of secondary interest are allowed to occur zero or more times;
   - Both obtain Patient’s stated name and birth date and apply specimen container label i are allowed to occur again, zero or more times, and there are no restrictions on the occurrences of any future events.

FSA and DNL for Property BT.D.2b
SCOPE:
1. A restricted interval in the event sequence can have both a starting delimiter, receive order to obtain a specimen, and an ending delimiter, send specimen i to blood bank.
2. The behavior is required to hold from an occurrence of receive order to obtain a specimen, if it ever occurs, through to the first subsequent occurrence of send specimen i to blood bank, if it ever occurs.
3. If there are multiple occurrences of receive order to obtain a specimen without an occurrence of send specimen i to blood bank in between them, only the first of those occurrences of receive order to obtain a specimen potentially starts a restricted interval; later occurrences of receive order to obtain a specimen within this restricted interval do not have an effect.
4. receive order to obtain a specimen is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if receive order to obtain a specimen does occur, send specimen i to blood bank is not required to occur subsequently and if it does not occur subsequently, then the behavior is not required to hold for the remainder of the event sequence.
5. There might be many restricted intervals in the event sequence. If receive order to obtain a specimen occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of receive order to obtain a specimen, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a potential new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

BEHAVIOR:
1. The events of primary interest in this behavior are make sure that ID band and Patient’s stated name and birth date match and obtain specimen i.
2. The events of secondary interest in this behavior are apply specimen container label i, confirm presence of ID band, make sure that ID band and specimen match, make sure that ID band and specimen container label i match, and obtain Patient’s stated name and birth date.
3. obtain specimen i is not allowed to occur until after make sure that ID band and Patient’s stated name and birth date match occurs.
4. Before the first make sure that ID band and Patient’s stated name and birth date match occurs, all the events of secondary interest are allowed to occur zero or more times.
5. make sure that ID band and Patient’s stated name and birth date match is required to occur, whether or not obtain specimen i eventually occurs.
6. obtain specimen i is not required to occur after make sure that ID band and Patient’s stated name and birth date match occurs.
7. After make sure that ID band and Patient’s stated name and birth date match occurs, but before the first subsequent obtain specimen i occurs:
   • make sure that ID band and Patient’s stated name and birth date match is allowed to occur again, zero or more times;
   • All the events of secondary interest are allowed to occur zero or more times.
8. After make sure that ID band and Patient’s stated name and birth date match and the first subsequent obtain specimen i occurs:
   • All the events of secondary interest are allowed to occur zero or more times;
   • Both make sure that ID band and Patient’s stated name and birth date match and obtain specimen i are allowed to occur again, zero or more times, and there are no restrictions on the occurrences of any future events.

FSA and DNL for Property BT.D.3a
Event alphabet:

- A: make sure that ID band and Patient’s stated name and birth date match
- B: apply specimen container label i
- START: receive order to obtain a specimen
- END: send specimen i to blood bank
- C: confirm presence of ID band, make sure that ID band and order to obtain a specimen match, make sure that ID band and specimen container label i match, obtain Patient’s stated name and birth date, obtain specimen i

**SCOPE:**
1. A restricted interval in the event sequence can have both a starting delimiter, receive order to obtain a specimen, and an ending delimiter, send specimen i to blood bank.
2. The behavior is required to hold from an occurrence of receive order to obtain a specimen, if it ever occurs, through to the first subsequent occurrence of send specimen i to blood bank, if it ever occurs.
3. If there are multiple occurrences of receive order to obtain a specimen without an occurrence of send specimen i to blood bank in between them, only the first of those occurrences of receive order to obtain a specimen potentially starts a restricted interval; later occurrences of receive order to obtain a specimen within this restricted interval do not have an effect.
4. receive order to obtain a specimen is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if receive order to obtain a specimen does occur, send specimen i to blood bank is not required to occur subsequently and if it does not occur subsequently, then the behavior is not required to hold for the remainder of the event sequence.
5. There might be many restricted intervals in the event sequence. If receive order to obtain a specimen occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of receive order to obtain a specimen, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a potential new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

**BEHAVIOR:**
1. The events of primary interest in this behavior are make sure that ID band and Patient’s stated name and birth date match and apply specimen container label i.
2. The events of secondary interest in this behavior are confirm presence of ID band, make sure that ID band and order to obtain a specimen match, make sure that ID band and specimen container label i match, obtain Patient’s stated name and birth date, and obtain specimen i.
3. apply specimen container label i is not allowed to occur until after make sure that ID band and Patient’s stated name and birth date match occurs.
4. Before the first make sure that ID band and Patient’s stated name and birth date match occurs, all the events of secondary interest are allowed to occur zero or more times.
5. make sure that ID band and Patient’s stated name and birth date match is required to occur, whether or not apply specimen container label i eventually occurs.
6. apply specimen container label i is not required to occur after make sure that ID band and Patient’s stated name and birth date match occurs.
7. After make sure that ID band and Patient’s stated name and birth date match occurs, but before the first subsequent apply specimen container label i occurs:
   - make sure that ID band and Patient’s stated name and birth date match is allowed to occur again, zero or more times;
   - All the events of secondary interest are allowed to occur zero or more times.
8. After make sure that ID band and Patient’s stated name and birth date match and the first subsequent apply specimen container label i occur:
   - All the events of secondary interest are allowed to occur zero or more times;
   - Both make sure that ID band and Patient’s stated name and birth date match and apply specimen container label i are allowed to occur again, zero or more times, and there are no restrictions on the occurrences of any future events.

FSA and DNL for Property BT.D.3b
Event alphabet:

- A: make sure that ID band and order to obtain a specimen match
- B: obtain specimen i
- START: receive order to obtain a specimen
- END: send specimen i to blood bank
- C: apply specimen container label i, confirm presence of ID band, make sure that ID band and Patient’s stated name and birth date match, make sure that ID band and specimen container label i match, obtain Patient’s stated name and birth date

SCOPE:
1. A restricted interval in the event sequence can have both a starting delimiter, receive order to obtain a specimen, and an ending delimiter, send specimen i to blood bank.
2. The behavior is required to hold from an occurrence of receive order to obtain a specimen, if it ever occurs, through to the first subsequent occurrence of send specimen i to blood bank, if it ever occurs.
3. If there are multiple occurrences of receive order to obtain a specimen without an occurrence of send specimen i to blood bank in between them, only the first of those occurrences of receive order to obtain a specimen potentially starts a restricted interval; later occurrences of receive order to obtain a specimen within this restricted interval do not have an effect.
4. receive order to obtain a specimen is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if receive order to obtain a specimen does occur, send specimen i to blood bank is not required to occur subsequently and if it does not occur subsequently, then the behavior is not required to hold for the remainder of the event sequence.
5. There might be many restricted intervals in the event sequence. If receive order to obtain a specimen occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of receive order to obtain a specimen, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a potential new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

BEHAVIOR:
1. The events of primary interest in this behavior are make sure that ID band and order to obtain a specimen match and obtain specimen i.
2. The events of secondary interest in this behavior are apply specimen container label i, confirm presence of ID band, make sure that ID band and Patient’s stated name and birth date match, make sure that ID band and specimen container label i match, and obtain Patient’s stated name and birth date.
3. obtain specimen i is not allowed to occur until after make sure that ID band and order to obtain a specimen match occurs.
4. Before the first make sure that ID band and order to obtain a specimen match occurs, all the events of secondary interest are allowed to occur zero or more times.
5. make sure that ID band and order to obtain a specimen match is required to occur, whether or not obtain specimen i eventually occurs.
6. obtain specimen i is not required to occur after make sure that ID band and order to obtain a specimen match occurs.
7. After make sure that ID band and order to obtain a specimen match occurs, but before the first subsequent obtain specimen i occurs:
   - make sure that ID band and order to obtain a specimen match is allowed to occur again, zero or more times;
   - All the events of secondary interest are allowed to occur zero or more times;
8. After make sure that ID band and order to obtain a specimen match and the first subsequent obtain specimen i occur:
   - All the events of secondary interest are allowed to occur zero or more times;
   - Both make sure that ID band and order to obtain a specimen match and obtain specimen i are allowed to occur again, zero or more times, and there are no restrictions on the occurrences of any future events.

FSA and DNL for Property BT.D.4a
Event alphabet:
• A: make sure that ID band and order to obtain a specimen match
• B: apply specimen container label i
• START: receive order to obtain a specimen
• END: send specimen i to blood bank
• C: confirm presence of ID band, make sure that ID band and Patient’s stated name and birth date match, make sure that ID band and specimen container label i match, obtain Patient’s stated name and birth date, obtain specimen i

SCOPE:
1. A restricted interval in the event sequence can have both a starting delimiter, receive order to obtain a specimen, and an ending delimiter, send specimen i to blood bank.
2. The behavior is required to hold from an occurrence of receive order to obtain a specimen, if it ever occurs, through to the first subsequent occurrence of send specimen i to blood bank, if it ever occurs.
3. If there are multiple occurrences of receive order to obtain a specimen without an occurrence of send specimen i to blood bank in between them, only the first of those occurrences of receive order to obtain a specimen potentially starts a restricted interval; later occurrences of receive order to obtain a specimen within this restricted interval do not have an effect.
4. receive order to obtain a specimen is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if receive order to obtain a specimen does occur, send specimen i to blood bank is not required to occur subsequently and if it does not occur subsequently, then the behavior is not required to hold for the remainder of the event sequence.
5. There might be many restricted intervals in the event sequence. If receive order to obtain a specimen occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of receive order to obtain a specimen, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a potential new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

BEHAVIOR:
1. The events of primary interest in this behavior are make sure that ID band and order to obtain a specimen match and apply specimen container label i.
2. The events of secondary interest in this behavior are confirm presence of ID band, make sure that ID band and specimen container label i match, obtain Patient’s stated name and birth date, and obtain specimen i.
3. apply specimen container label i is not allowed to occur until after make sure that ID band and order to obtain a specimen match occurs.
4. Before the first make sure that ID band and order to obtain a specimen match occurs, all the events of secondary interest are allowed to occur zero or more times.
5. make sure that ID band and order to obtain a specimen match is required to occur, whether or not apply specimen container label i eventually occurs.
6. apply specimen container label i is not required to occur after make sure that ID band and order to obtain a specimen match occurs.
7. After make sure that ID band and order to obtain a specimen match occurs, but before the first subsequent apply specimen container label i occurs:
   • make sure that ID band and order to obtain a specimen match is allowed to occur again, zero or more times;
   • All the events of secondary interest are allowed to occur zero or more times.
8. After make sure that ID band and order to obtain a specimen match and the first subsequent apply specimen container label i occur:
   • Both make sure that ID band and order to obtain a specimen match and apply specimen container label i are allowed to occur again, zero or more times, and there are no restrictions on the occurrences of any future events.

FSA and DNL for Property BT.D.4b
Event alphabet:
- A: make sure that ID band and specimen container label i match
- B: obtain specimen i
- START: receive order to obtain a specimen
- END: send specimen i to blood bank
- C: apply specimen container label i, confirm presence of ID band, make sure that ID band and order to obtain a specimen match, make sure that ID band and Patient’s stated name and birth date match, obtain Patient’s stated name and birth date

SCOPE:
1. A restricted interval in the event sequence can have both a starting delimiter, receive order to obtain a specimen, and an ending delimiter, send specimen i to blood bank.
2. The behavior is required to hold from an occurrence of receive order to obtain a specimen, if it ever occurs, through to the first subsequent occurrence of send specimen i to blood bank, if it ever occurs.
3. If there are multiple occurrences of receive order to obtain a specimen without an occurrence of send specimen i to blood bank in between them, only the first of those occurrences of receive order to obtain a specimen potentially starts a restricted interval; later occurrences of receive order to obtain a specimen within this restricted interval do not have an effect.
4. receive order to obtain a specimen is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if receive order to obtain a specimen does occur, send specimen i to blood bank is not required to occur subsequently and if it does not occur subsequently, then the behavior is not required to hold for the remainder of the event sequence.
5. There might be many restricted intervals in the event sequence. If receive order to obtain a specimen occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of receive order to obtain a specimen, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a potential new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

BEHAVIOR:
1. The events of primary interest in this behavior are make sure that ID band and specimen container label i match and obtain specimen i.
2. The events of secondary interest in this behavior are apply specimen container label i, confirm presence of ID band, make sure that ID band and order to obtain a specimen match, make sure that ID band and Patient’s stated name and birth date match, and obtain Patient’s stated name and birth date.
3. obtain specimen i is not allowed to occur until after make sure that ID band and specimen container label i match occurs.
4. Before the first make sure that ID band and specimen container label i match occurs, all the events of secondary interest are allowed to occur zero or more times.
5. make sure that ID band and specimen container label i match is required to occur, whether or not obtain specimen i eventually occurs.
6. obtain specimen i is not required to occur after make sure that ID band and specimen container label i match occurs.
7. After make sure that ID band and specimen container label i match occurs, but before the first subsequent obtain specimen i occurs:
   - All the events of secondary interest are allowed to occur zero or more times.
8. After make sure that ID band and specimen container label i match and the first subsequent obtain specimen i occur:
   - All the events of secondary interest are allowed to occur zero or more times.
   - obtain specimen i is not allowed to occur again until another make sure that ID band and specimen container label i match occurs; make sure that ID band and specimen container label i match is allowed to occur again and, if it does, then the situation is the same as when the first make sure that ID band and specimen container label i match occurred, meaning that the restrictions described in parts 6, 7, and 8 would again apply.
Event alphabet:
- A: make sure that ID band and specimen container label i match
- B: apply specimen container label i
- START: receive order to obtain a specimen
- END: send specimen i to blood bank
- C: confirm presence of ID band, make sure that ID band and order to obtain a specimen match, make sure that ID band and Patient’s stated name and birth date match, obtain Patient’s stated name and birth date, obtain specimen i

Scope:
1. A restricted interval in the event sequence can have both a starting delimiter, receive order to obtain a specimen, and an ending delimiter, send specimen i to blood bank.
2. The behavior is required to hold from an occurrence of receive order to obtain a specimen, if it ever occurs, through to the first subsequent occurrence of send specimen i to blood bank, if it ever occurs.
3. If there are multiple occurrences of receive order to obtain a specimen without an occurrence of send specimen i to blood bank in between them, only the first of those occurrences of receive order to obtain a specimen potentially starts a restricted interval; later occurrences of receive order to obtain a specimen within this restricted interval do not have an effect.
4. receive order to obtain a specimen is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if receive order to obtain a specimen does occur, send specimen i to blood bank is not required to occur subsequently and if it does not occur subsequently, then the behavior is not required to hold for the remainder of the event sequence.
5. There might be many restricted intervals in the event sequence. If receive order to obtain a specimen occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of receive order to obtain a specimen, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a potential new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

Behavior:
1. The events of primary interest in this behavior are make sure that ID band and specimen container label i match and apply specimen container label i.
2. The events of secondary interest in this behavior are confirm presence of ID band, make sure that ID band and order to obtain a specimen match, make sure that ID band and Patient’s stated name and birth date match, obtain Patient’s stated name and birth date, and obtain specimen i.
3. apply specimen container label i is not allowed to occur until after make sure that ID band and specimen container label i match occurs.
4. Before the first make sure that ID band and specimen container label i match occurs, all the events of secondary interest are allowed to occur zero or more times.
5. make sure that ID band and specimen container label i match is required to occur, whether or not apply specimen container label i eventually occurs.
6. apply specimen container label i is not required to occur after make sure that ID band and specimen container label i match occurs.
7. After make sure that ID band and specimen container label i match occurs, but before the first subsequent apply specimen container label i occurs:
   - make sure that ID band and specimen container label i match is allowed to occur again, zero or more times;
   - All the events of secondary interest are allowed to occur zero or more times.
8. After make sure that ID band and specimen container label i match and the first subsequent apply specimen container label i occur:
   - All the events of secondary interest are allowed to occur zero or more times;
   - apply specimen container label i is not allowed to occur again until after another make sure that ID band and specimen container label i match occurs, make sure that ID band and specimen container label i match is allowed to occur again and, if it does, then the situation is the same as when the first make sure that ID band and specimen container label i match occurred, meaning that the restrictions described in parts 6, 7, and 8 would again apply.

FSA and DNL for Property BT.D.5b
Event alphabet:
- A: confirm presence of ID band
- B: obtain Patient’s stated name and birth date
- START: receive an order to obtain a specimen
- END: obtain specimen OR apply specimen container label
- C: *

SCOPE:
1. A restricted interval in the event sequence can have both a starting delimiter, receive an order to obtain a specimen, and an ending delimiter, obtain specimen OR apply specimen container label.
2. The behavior is required to hold from an occurrence of receive an order to obtain a specimen, if it ever occurs, through to the first subsequent occurrence of obtain specimen OR apply specimen container label, if it ever occurs.
3. If there are multiple occurrences of receive an order to obtain a specimen without an occurrence of obtain specimen OR apply specimen container label in between them, only the first of those occurrences of receive an order to obtain a specimen potentially starts a restricted interval; later occurrences of receive an order to obtain a specimen within this restricted interval do not have an effect.
4. receive an order to obtain a specimen is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if receive an order to obtain a specimen does occur, obtain specimen OR apply specimen container label is not required to occur subsequently and if it does not occur subsequently, then the behavior is not required to hold for the remainder of the event sequence.
5. There might be many restricted intervals in the event sequence. If receive an order to obtain a specimen occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of receive an order to obtain a specimen, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a potential new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

BEHAVIOR:
1. The events of primary interest in this behavior are confirm presence of ID band and obtain Patient’s stated name and birth date.
2. The event of secondary interest in this behavior is *.
3. obtain Patient’s stated name and birth date is not allowed to occur until after confirm presence of ID band occurs.
4. Before the first confirm presence of ID band occurs, * is allowed to occur zero or more times.
5. confirm presence of ID band is required to occur, whether or not obtain Patient’s stated name and birth date eventually occurs.
6. obtain Patient’s stated name and birth date is not required to occur after confirm presence of ID band occurs.
7. After confirm presence of ID band occurs, but before the first subsequent obtain Patient’s stated name and birth date occurs:
   - confirm presence of ID band is allowed to occur again, zero or more times;
   - * is not allowed to occur.
8. After confirm presence of ID band and the first subsequent obtain Patient’s stated name and birth date occur:
   - * is allowed to occur zero or more times;
   - Both confirm presence of ID band and obtain Patient’s stated name and birth date are allowed to occur again, zero or more times, and there are no restrictions on the occurrences of any future events.

FSA and DNL for Property BT.D.6 (D.1 → D.2)
Event alphabet:
- A: obtain Patient’s stated name and birth date
- B: make sure that ID band and Patient’s stated name and birth date match
- START: receive an order to obtain a specimen
- END: obtain specimen OR apply specimen container label
- C: confirm presence of ID band

SCOPE:
1. A restricted interval in the event sequence can have both a starting delimiter, receive an order to obtain a specimen, and an ending delimiter, obtain specimen OR apply specimen container label.
2. The behavior is required to hold from an occurrence of receive an order to obtain a specimen, if it ever occurs, through to the first subsequent occurrence of obtain specimen OR apply specimen container label, if it ever occurs.
3. If there are multiple occurrences of receive an order to obtain a specimen without an occurrence of obtain specimen OR apply specimen container label in between them, only the first of those occurrences of receive an order to obtain a specimen potentially starts a restricted interval; later occurrences of receive an order to obtain a specimen within this restricted interval do not have an effect.
4. receive an order to obtain a specimen is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if receive an order to obtain a specimen does occur, obtain specimen OR apply specimen container label is not required to occur subsequently and if it does not occur subsequently, then the behavior is not required to hold for the remainder of the event sequence.
5. There might be many restricted intervals in the event sequence. If receive an order to obtain a specimen occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of receive an order to obtain a specimen, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a potential new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

BEHAVIOR:
1. The events of primary interest in this behavior are obtain Patient’s stated name and birth date and make sure that ID band and Patient’s stated name and birth date match.
2. The event of secondary interest in this behavior is confirm presence of ID band.
3. make sure that ID band and Patient’s stated name and birth date match is not allowed to occur until after obtain Patient’s stated name and birth date occurs.
4. Before the first obtain Patient’s stated name and birth date occurs, confirm presence of ID band is allowed to occur zero or more times.
5. obtain Patient’s stated name and birth date occurs, confirm presence of ID band is allowed to occur zero or more times.
6. make sure that ID band and Patient’s stated name and birth date match is not required to occur after obtain Patient’s stated name and birth date occurs.
7. After obtain Patient’s stated name and birth date occurs, but before the first subsequent make sure that ID band and Patient’s stated name and birth date match occurs:
   - obtain Patient’s stated name and birth date is allowed to occur again, zero or more times;
   - confirm presence of ID band is allowed to occur zero or more times.
8. After obtain Patient’s stated name and birth date and the first subsequent make sure that ID band and Patient’s stated name and birth date match occur:
   - confirm presence of ID band is allowed to occur zero or more times;
   - Both obtain Patient’s stated name and birth date and make sure that ID band and Patient’s stated name and birth date match are allowed to occur again, zero or more times, and there are no restrictions on the occurrences of any future events.

FSA and DNL for Property BT.D.6 (D.2 → D.3)
Event alphabet:

- **A**: make sure that ID band and Patient’s stated name and birth date match
- **B**: make sure that ID band and order to obtain a specimen match
- **START**: receive an order to obtain a specimen
- **END**: obtain specimen OR apply specimen container label
- **C**: confirm presence of ID band, obtain Patient’s stated name and birth date

**SCOPE:**

1. A restricted interval in the event sequence can have both a starting delimiter, receive an order to obtain a specimen, and an ending delimiter, obtain specimen OR apply specimen container label.
2. The behavior is required to hold from an occurrence of receive an order to obtain a specimen, if it ever occurs, through to the first subsequent occurrence of obtain specimen OR apply specimen container label, if it ever occurs.
3. If there are multiple occurrences of receive an order to obtain a specimen without an occurrence of obtain specimen OR apply specimen container label in between them, only the first of those occurrences of receive an order to obtain a specimen potentially starts a restricted interval; later occurrences of receive an order to obtain a specimen within this restricted interval do not have an effect.
4. receive an order to obtain a specimen is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if receive an order to obtain a specimen does occur, obtain specimen OR apply specimen container label is not required to occur subsequently and if it does not occur subsequently, then the behavior is not required to hold for the remainder of the event sequence.
5. There might be many restricted intervals in the event sequence. If receive an order to obtain a specimen occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of receive an order to obtain a specimen, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a potential new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

**BEHAVIOR:**

1. The events of primary interest in this behavior are make sure that ID band and Patient’s stated name and birth date match and make sure that ID band and order to obtain a specimen match.
2. The events of secondary interest in this behavior are confirm presence of ID band and obtain Patient’s stated name and birth date.
3. make sure that ID band and order to obtain a specimen match is not allowed to occur until after make sure that ID band and Patient’s stated name and birth date match occurs.
4. Before the first make sure that ID band and Patient’s stated name and birth date match occurs, all the events of secondary interest are allowed to occur zero or more times.
5. make sure that ID band and Patient’s stated name and birth date match is required to occur, whether or not make sure that ID band and Patient’s stated name and birth date match occurs.
6. make sure that ID band and order to obtain a specimen match is not required to occur after make sure that ID band and Patient’s stated name and birth date match occurs.
7. After make sure that ID band and Patient’s stated name and birth date match occurs, but before the first subsequent make sure that ID band and order to obtain a specimen match occurs:
   - All the events of secondary interest are allowed to occur zero or more times;
   - Both make sure that ID band and Patient’s stated name and birth date match and make sure that ID band and order to obtain a specimen match are allowed to occur again, zero or more times, and there are no restrictions on the occurrences of any future events.
8. After make sure that ID band and Patient’s stated name and birth date match and the first subsequent make sure that ID band and order to obtain a specimen match occur:
   - All the events of secondary interest are allowed to occur zero or more times;

FSA and DNL for Property BT.D.6 (D.3 → D.4)
Event alphabet:
- A: make sure that ID band and order to obtain a specimen match
- B: make sure that ID band and specimen container label i match
- START: receive an order to obtain a specimen
- END: obtain specimen i OR apply specimen container label i
- C: confirm presence of ID band, make sure that ID band and Patient’s stated name and birth date match, obtain Patient’s stated name and birth date

**SCOPE:**
1. A restricted interval in the event sequence can have both a starting delimiter, receive an order to obtain a specimen, and an ending delimiter, obtain specimen i OR apply specimen container label i.
2. The behavior is required to hold from an occurrence of receive an order to obtain a specimen, if it ever occurs, through to the first subsequent occurrence of obtain specimen i OR apply specimen container label i, if it ever occurs.
3. If there are multiple occurrences of receive an order to obtain a specimen without an occurrence of obtain specimen i OR apply specimen container label i in between them, only the first of those occurrences of receive an order to obtain a specimen potentially starts a restricted interval; later occurrences of receive an order to obtain a specimen within this restricted interval do not have an effect.
4. receive an order to obtain a specimen is not required to occur and if it never occurs, the behavior is not required to hold anywhere in the event sequence. Even if receive an order to obtain a specimen does occur, obtain specimen i OR apply specimen container label i is not required to occur subsequently and if it does not occur subsequently, then the behavior is not required to hold for the remainder of the event sequence.
5. There might be many restricted intervals in the event sequence. If receive an order to obtain a specimen occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of receive an order to obtain a specimen, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a potential new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

**BEHAVIOR:**
1. The events of primary interest in this behavior are make sure that ID band and order to obtain a specimen match and make sure that ID band and specimen container label i match.
2. The events of secondary interest in this behavior are confirm presence of ID band, make sure that ID band and Patient’s stated name and birth date match, and obtain Patient’s stated name and birth date.
3. make sure that ID band and specimen container label i match is not allowed to occur until after make sure that ID band and order to obtain a specimen match occurs.
4. Before the first make sure that ID band and order to obtain a specimen match occurs, all the events of secondary interest are allowed to occur zero or more times.
5. make sure that ID band and order to obtain a specimen match is required to occur, whether or not make sure that ID band and specimen container label i match eventually occurs.
6. make sure that ID band and specimen container label i match is not required to occur after make sure that ID band and order to obtain a specimen match occurs.
7. After make sure that ID band and order to obtain a specimen match occurs, but before the first subsequent make sure that ID band and specimen container label i match occurs:
   - make sure that ID band and order to obtain a specimen match is allowed to occur again, zero or more times; 
   - All the events of secondary interest are allowed to occur zero or more times.
8. After make sure that ID band and order to obtain a specimen match and the first subsequent make sure that ID band and specimen container label i match occur:
   - All the events of secondary interest are allowed to occur zero or more times; 
   - Both make sure that ID band and order to obtain a specimen match and make sure that ID band and specimen container label i match are allowed to occur again, zero or more times, and there are no restrictions on the occurrences of any future events.

FSA and DNL for Property BT.D.6 (D.4 → D.5)
C.2 Chemotherapy

C.2.1 Description of the Domain

The Chemotherapy (Chemo) case study is part of a larger medical safety project that our research group is participating in. In this case study, the domain experts were an M.D., an R.N., a Pharm.D., and an M.A. We elicited all of the properties through interviews with them. Two of the domain experts were trained in how to interpret the FSA template view, and although both were able to quickly learn the basics of that formalism, one preferred not to use it to reason about the properties. Although we, the computer scientists, were primarily the ones who worked with PROPEL to formally specify the properties, one of the domain experts used the FSA template and QT property views to specify 1-2 of them himself, with us observing in those cases. In addition, all of these domain experts worked with us to vet all of the informal and formal property specifications against their domain expertise. None of these domain experts had any prior knowledge of property specification formalisms or RE.

Due to resource limitations, we focused on a subset of the Chemo domain, specifically covering only out-patient clinical situations, not including protocols for experimental chemotherapy research. In addition, all properties in this case study are assumed to be in the context of one patient, a simplified chemotherapy case (e.g., an AC breast cancer patient), and only the activities necessary for a single course of chemotherapy, where a single course can contain one or more cycles, and a single cycle can contain one or more administration episodes.

There are 59 informal property specifications in this case study, and they are given in Appendix C.2.2. See Section 5.4.3.2 for an explanation of the notation used in the informal specifications. Appendix C.2.3 gives the glossary for all of the bolded terms used in the informal specifications; see Section 5.4.3.1 for an explanation of the glossary structure. There are 119 formal property specifications in this case study, and they are given in Appendix C.2.4. The FSA and DNL specifications are given for each property, with some modifications made to these two views to enable the property specifications to fit on a single page. The FSA specification is shown in two parts: at the top is a mapping from parameter names to specifier-specified events, labeled “Event alphabet:”, and directly below that mapping is an FSA specification of the property. The FSA is assumed to be total and deterministic\(^2\) and its transitions are labeled with the parameters, rather than with the specifier-specified events. The DNL specification is also shown in two parts, the scope and the behavior, but a third part, the preamble, is elided. The preamble is the same for every DNL property specification, and can be found in Section 3.2.4.

\(^2\)For brevity, we do not show the transitions that go to a non-accepting trap state. When no transition is provided that explicitly allows an event to occur, it should be assumed that an occurrence of that event puts the FSA into a non-accepting trap state.
C.2.2 Organization of the Informal Property Specifications

Chemotherapy Properties

A. PATIENT ELIGIBILITY

A Patient must have a cancer diagnosis that is verified by qualified medical staff before chemotherapy can be administered to that Patient.

Before chemotherapy can be administered to a Patient:

MP A.1 • a Baystate Pathologist must review that Patient’s pathology.

MP A.2 • that Patient must have a consult with a Baystate Attending MD.

MP A.3 If the Attending MD decides that a Patient’s Baystate pathology report does not indicate a cancer diagnosis, chemotherapy cannot be administered to that Patient.

B. LEGAL CONSTRAINTS

A Patient must give informed consent before chemotherapy can be administered to that Patient.

Before chemotherapy can be administered to a Patient:

MP B.1 • that Patient’s consult note must be present in that Patient’s record.

MP B.2 • that Patient must sign a consent form for chemotherapy.

MP B.3 • that Patient’s signed consent form for chemotherapy must be present in that Patient’s record.

MP B.4 • that Patient’s treatment plan must be present in that Patient’s record.

MP B.5 • after that Patient signs a consent form for chemotherapy, if a critical change occurs in the treatment plan, that Patient must sign a consent form for the chemotherapy in the new treatment plan.

If any of these cases of B.5 occur, they must be satisfied for B.5 to be satisfied. These cases enumerate what “a critical change occurs in the treatment plan” could mean.

After that Patient signs a consent form for chemotherapy:

-- B.5.1 • if the chemotherapy drugs in the treatment plan change, that Patient must sign a consent form for the chemotherapy in the new treatment plan.

-- B.5.2 • if a new treatment plan is created, that Patient must sign a consent form for the chemotherapy in the new treatment plan.

Before a Patient can sign a consent form for chemotherapy:

MP B.6 • that Patient must have a consult with a Baystate Attending MD.

MP B.7 • that Patient must have chemotherapy teaching.

MP B.8 • the activities described above must occur in that order.

Figure C.9. Chemotherapy Case Study Informal Specifications - Page 1
Chemotherapy Properties

C. DEVELOPMENT OF TREATMENT PLAN AND CHEMOTHERAPY ORDERS

A treatment plan for a course of chemotherapy for a Patient must be created and approved by qualified medical staff before chemotherapy can be administered to that Patient. The chemotherapy orders for each cycle in the course must be entered and verified by qualified medical staff before chemotherapy for that cycle can be administered to that Patient. Each set of chemotherapy orders must be consistent with the treatment plan, must contain only up-to-date Patient data, and must be consistent with that Patient data.

MP C.1 A Patient’s treatment plan must be approved before chemotherapy can be administered to that Patient for the first time.

Approvals required: Attending MD (if Fellow MD created the treatment plan), Practice RN or second Clinic RN (Practice RN preferred), Pharmacy, and Clinic RN.

MP C.2 All critical changes made to a Patient’s treatment plan after its creation must be approved before chemotherapy can be administered to that Patient for the first time.

Approvals required: Attending MD (if Fellow MD created the new treatment plan), Pharmacy, and Clinic RN.

MP C.3 A Patient’s chemotherapy orders for a cycle must be verified before chemotherapy can be administered to that Patient in that cycle.

Verifications required: Practice RN or second Clinic RN (first cycle only, Practice RN preferred), Pharmacy and Clinic RN.

MP C.4 All critical changes made to a Patient’s chemotherapy orders for a cycle must be approved before a verification of that Patient’s chemotherapy orders for that cycle can be completed.

Approvals required: Attending MD or Fellow MD.

Verifications required: Pharmacy and Clinic RN.

MP C.5 If a critical change is made to a Patient’s chemotherapy orders, those changed chemotherapy orders must be verified before chemotherapy can be administered to that Patient.

Verifications required: Pharmacy and Clinic RN.

Before a Patient’s treatment plan can be approved:

MP C.6

- that Patient’s chemotherapy orders must be consistent with that Patient’s treatment plan.
- all of the chemotherapy drugs in that Patient’s treatment plan must be:
  - consistent with that Patient’s cancer diagnosis.
  - in doses that are consistent with that Patient’s data (i.e., height, weight, laboratory results, consult note).

Approvals required: Attending MD (if Fellow MD created the new treatment plan), Practice RN or second Clinic RN (Practice RN preferred), Pharmacy, and Clinic RN.
Chemotherapy Properties

MP C.9  ● all the Patient data (i.e., height, weight, laboratory results, consult note) in that Patient’s treatment plan must not be stale or disparate.

Approvals required: Attending MD (if Fellow MD created the new treatment plan), Practice RN or second Clinic RN (Practice RN preferred), Pharmacy, and Clinic RN.

MP C.10  No treatment plan that is based on a Patient’s stale or disparate data (i.e., height, weight, laboratory results, consult note) can be created for that Patient.

Before the verifications of each of that Patient’s chemotherapy orders can be completed:

MP C.11  ● that Patient’s chemotherapy orders must be consistent with that Patient’s treatment plan.

MP C.12  ● all of the chemotherapy drugs in that Patient’s chemotherapy orders must be consistent with that Patient’s cancer diagnosis.

MP C.13  ● in doses that are consistent with that Patient’s data (i.e., height, weight, laboratory results, previous chemotherapy administration information (only after first cycle)).

Verifications required: Practice RN or second Clinic RN (first cycle only, Practice RN preferred), Pharmacy and Clinic RN

MP C.14  ● all the Patient data (i.e., height, weight, laboratory results, previous chemotherapy administration information (only after first cycle)) in a Patient’s chemotherapy orders must not be stale or disparate.

Verifications required: Practice RN or second Clinic RN (first cycle only, Practice RN preferred), Pharmacy and Clinic RN

D. ACTIVITIES REQUIRED RIGHT BEFORE CHEMOTHERAPY IS ADMINISTERED

Before chemotherapy can be administered, it must be confirmed that the right patient is present and that the right drugs are administered to that patient at the right time and for the right reason.

Before pre-chemotherapy supportive care medications or chemotherapy can be administered to a Patient:

MP D.1  ● that Patient must be correctly identified.

MP D.2  ● that Patient must be well enough to receive treatment.

MP D.3  ● that Patient must have appropriate I.V. access.
Chemotherapy Properties

**MP D.4**
- the confirmations of the Patient information described above must occur in that order.

If a Patient’s data (i.e., height, weight, laboratory results, previous chemotherapy administration information (only after first cycle)) becomes stale or disparate, the problem must be resolved before:

**MP D.5**
- chemotherapy drugs can be prepared for that Patient.

**MP D.6**
- pre-chemotherapy supportive care medications can be administered to that Patient.

**MP D.7**
- chemotherapy can be administered to that Patient.

Before chemotherapy can be administered to a Patient, all of that Patient’s chemotherapy drugs must:

**MP D.8**
- be physically suitable for administration.

**MP D.9**
- be consistent with that Patient’s cancer diagnosis.

**MP D.10**
- match that Patient’s approved treatment plan and verified chemotherapy orders.

**MP D.11**
- be in doses that are consistent with that Patient’s data (i.e., height, weight, laboratory results, previous chemotherapy administration information (only after first cycle)).

**MP D.12**
- be correctly prepared for that Patient.

**MP D.13**
- be assigned to that Patient.

Before pre-chemotherapy supportive care medications can be administered to a Patient, all of that Patient’s pre-chemotherapy supportive care medications must:

**MP D.14**
- be physically suitable for administration.

**MP D.15**
- be consistent with that Patient’s chemotherapy drugs.

**MP D.16**
- match that Patient’s verified chemotherapy orders.

**MP D.17**
- be in doses that are consistent with that Patient’s data (i.e., height, weight, laboratory results, previous chemotherapy administration information (only after first cycle)).

**MP D.18**
- be correctly prepared for that Patient.

**MP D.19**
- be assigned to that Patient.

**MP D.20**

Before chemotherapy can be administered to a Patient, all pre-chemotherapy supportive care medications in that Patient’s verified chemotherapy orders must be administered to that Patient.

If a Patient has an adverse reaction to an administration of pre-chemotherapy supportive care medications:

**MP D.21**
- that administration of pre-chemotherapy supportive care medications must be stopped immediately.
Chemotherapy Properties

D. ACTIVITIES REQUIRED RIGHT BEFORE CHEMOTHERAPY IS ADMINISTERED

Before chemotherapy can be administered, it must be confirmed that the right patient is present and that the right drugs are administered to that patient at the right time and for the right reason.

If a Patient has an adverse reaction to an administration of pre-chemotherapy supportive care medications:

MP D.21
- that administration of pre-chemotherapy supportive care medications must be stopped immediately.

MP D.22
- that Patient's condition must be stabilized.

MP D.23
- that Patient's disposition must be decided.
   This can only be done by that Patient's Attending MD or one of the Attending MD's delegates (i.e. a Fellow MD that works under the auspices of the Attending MD or a Nurse Practitioner).

MP D.24
- the activities described above must occur in that order.

E. ACTIVITIES REQUIRED WHILE CHEMOTHERAPY IS BEING ADMINISTERED

Chemotherapy must be safely administered to a patient.

If a Patient has an adverse reaction to an administration of chemotherapy:

MP E.1
- that administration of chemotherapy must be stopped immediately.

MP E.2
- that Patient's condition must be stabilized.

MP E.3
- that Patient's disposition must be decided.
   This can only be done by that Patient's Attending MD or one of the Attending MD's delegates (i.e. a Fellow MD that works under the auspices of the Attending MD or a Nurse Practitioner).

MP E.4
- the activities described above must occur in that order.

F. ACTIVITIES REQUIRED AFTER CHEMOTHERAPY HAS BEEN ADMINISTERED

After chemotherapy has been administered to a patient, that patient must be given support for safely handling the outcome of that administration of chemotherapy and all administration data must be recorded.

After chemotherapy has been administered to a Patient and before that Patient can be discharged:

MP F.1
- that Patient must be well enough to be discharged.

MP F.2
- if that Patient needs observation, then that Patient must be observed.

MP F.3
- prescriptions for all the necessary post-chemotherapy supportive care medications must be given to that Patient.

MP F.4
- all post-chemotherapy instructions must be given to that Patient.

MP F.5
- a follow-up appointment must be scheduled for that Patient.

MP F.6
- After chemotherapy has been administered to a Patient, all of that Patient's laboratory results and all of that Patient's chemotherapy administration data must be entered into that Patient's record on the same business day.
### C.2.3 Glossaries

#### C.2.3.1 Chemotherapy Term Glossary

<table>
<thead>
<tr>
<th>Term</th>
<th>Definition</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>adverse reaction</td>
<td>have in response to administration of chemotherapy</td>
<td></td>
</tr>
<tr>
<td></td>
<td>have in response to administration of pre-chemotherapy supportive medications</td>
<td></td>
</tr>
<tr>
<td>cancer diagnosis</td>
<td>be consistent with</td>
<td>The cancer diagnosis must (to a lesser degree) be consistent with the treatment plan and (to a lesser degree) with the chemotherapy drugs in the chemotherapy orders, though what specific checks are done to establish consistency differs between these two artifacts. For the treatment plan, specific checks are done. For the chemotherapy drugs in the chemotherapy orders, specific checks are done.</td>
</tr>
<tr>
<td></td>
<td>be indicated by pathology report</td>
<td></td>
</tr>
<tr>
<td></td>
<td>identify</td>
<td></td>
</tr>
<tr>
<td></td>
<td>verify</td>
<td></td>
</tr>
<tr>
<td>change</td>
<td>Specifically, edits made to a Patient's treatment plan or chemotherapy orders.</td>
<td></td>
</tr>
<tr>
<td>approve</td>
<td></td>
<td>Only critical changes to the chemotherapy orders must be approved, by the Attending MD or Fellow MD. All changes made to the treatment plan prompt the creation of a new treatment plan by the Attending MD and/or Fellow MD.</td>
</tr>
<tr>
<td>make</td>
<td></td>
<td>Chemotherapy orders can have both critical and non-critical changes. All changes made to a treatment plan are considered critical changes.</td>
</tr>
<tr>
<td>• critical</td>
<td>Any change made to a treatment plan, and every change made to chemotherapy orders that is not in the set of non-critical changes.</td>
<td></td>
</tr>
<tr>
<td>• non-critical</td>
<td>A change that a Pharmacist can make to chemotherapy orders that does not require an Attending MD or Fellow MD to approve, and does not require re-approval of the treatment plan (and/or chemotherapy orders). Specifically, these are: a change in solution/dilution, a rounding off of dose (i.e., for the purposes of fixing obvious typographical errors), or a change in rates of administration.</td>
<td></td>
</tr>
<tr>
<td>chemotherapy</td>
<td></td>
<td></td>
</tr>
<tr>
<td>administer</td>
<td>Put chemotherapy drugs into the Patient's body.</td>
<td>syn. “administer chemotherapy drug.”</td>
</tr>
<tr>
<td>be for a cycle</td>
<td></td>
<td></td>
</tr>
<tr>
<td>be safely administered</td>
<td></td>
<td></td>
</tr>
<tr>
<td>have adverse reaction to an administration of chemotherapeutic drug</td>
<td></td>
<td></td>
</tr>
<tr>
<td>stop administration of chemotherapy drug</td>
<td></td>
<td>syn. “anti-cancer medication” but not “supportive-care medication.”</td>
</tr>
<tr>
<td>administer</td>
<td>Put drugs into the Patient’s body.</td>
<td></td>
</tr>
</tbody>
</table>

*Figure C.14. Chemotherapy Case Study Term Glossary - Page 1*
## Chemotherapy Property Glossary

<table>
<thead>
<tr>
<th>Term</th>
<th>Definition</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>chemotherapy drug</td>
<td>syn. &quot;anti-cancer medication&quot; but not &quot;supportive-care medication&quot;</td>
<td></td>
</tr>
<tr>
<td>administer</td>
<td>Put drugs into the Patient's body.</td>
<td></td>
</tr>
<tr>
<td>chemotherapy orders</td>
<td>Directions for which chemotherapy drugs, at which amounts, and at what rates and intervals the chemotherapy should be administered. Also includes similar directions for administering pre-chemotherapy supportive care medications.</td>
<td>Must verify chemotherapy orders before administer, enter, and prepare. Note that chemotherapy orders are not themselves approved, but that a verification of them is required before the treatment plan can be approved.</td>
</tr>
<tr>
<td>administer</td>
<td>Put chemotherapy drugs into the Patient's body.</td>
<td></td>
</tr>
<tr>
<td>be assigned to Patient</td>
<td>Adhering to physical drug quality standards.</td>
<td></td>
</tr>
<tr>
<td>be consistently prepared for Patient</td>
<td>Adhering to physical drug quality standards.</td>
<td></td>
</tr>
<tr>
<td>be in chemotherapy orders</td>
<td>All of the chemotherapy drug's data values must exactly match the associated values in the approved treatment plan and chemotherapy orders.</td>
<td>This is meant to imply that all chemotherapy drugs in the treatment plan and chemotherapy orders must be prepared (i.e., no missing drugs) and that all prepared chemotherapy drugs must be in the treatment plan and chemotherapy orders (i.e., no extra drugs).</td>
</tr>
<tr>
<td>be in treatment plan</td>
<td>All of the chemotherapy drug's data values must exactly match the associated values in the approved treatment plan and chemotherapy orders.</td>
<td>This is meant to imply that all chemotherapy drugs in the treatment plan and chemotherapy orders must be prepared (i.e., no missing drugs) and that all prepared chemotherapy drugs must be in the treatment plan and chemotherapy orders (i.e., no extra drugs).</td>
</tr>
<tr>
<td>be physically suitable for administration</td>
<td>Adhering to physical drug quality standards.</td>
<td></td>
</tr>
<tr>
<td>change in treatment plan</td>
<td>Adhering to physical drug quality standards.</td>
<td></td>
</tr>
<tr>
<td>chemotherapy drug dose</td>
<td>Directions for which chemotherapy drugs, at which amounts, and at what rates and intervals the chemotherapy should be administered. Also includes similar directions for administering pre-chemotherapy supportive care medications.</td>
<td>Must verify chemotherapy orders before administer, enter, and prepare. Note that chemotherapy orders are not themselves approved, but that a verification of them is required before the treatment plan can be approved.</td>
</tr>
<tr>
<td>administer</td>
<td>Put chemotherapy drugs into the Patient's body.</td>
<td></td>
</tr>
<tr>
<td>be consistent with Patient data</td>
<td></td>
<td></td>
</tr>
<tr>
<td>be changed</td>
<td>Can have critical or non-critical changes.</td>
<td></td>
</tr>
<tr>
<td>be consistent with</td>
<td>The chemotherapy orders must be consistent with the cancer diagnosis and the treatment plan, though what specific checks are done to establish consistency differs between these two artifacts.</td>
<td></td>
</tr>
<tr>
<td>be for a cycle of chemotherapy</td>
<td>For the cancer diagnosis, specific checks are done.</td>
<td></td>
</tr>
<tr>
<td>be verified</td>
<td>For the treatment plan, specific checks are done.</td>
<td></td>
</tr>
<tr>
<td>be consistent with Patient data</td>
<td></td>
<td></td>
</tr>
<tr>
<td>be consistent with</td>
<td>The chemotherapy orders must be consistent with the cancer diagnosis and the treatment plan, though what specific checks are done to establish consistency differs between these two artifacts.</td>
<td></td>
</tr>
<tr>
<td>be consistent with</td>
<td>For the cancer diagnosis, specific checks are done.</td>
<td></td>
</tr>
<tr>
<td>be consistent with</td>
<td>For the treatment plan, specific checks are done.</td>
<td></td>
</tr>
<tr>
<td>be consistent with</td>
<td>The chemotherapy orders must be consistent with the cancer diagnosis and the treatment plan, though what specific checks are done to establish consistency differs between these two artifacts.</td>
<td></td>
</tr>
<tr>
<td>be consistent with</td>
<td>For the cancer diagnosis, specific checks are done.</td>
<td></td>
</tr>
<tr>
<td>be consistent with</td>
<td>For the treatment plan, specific checks are done.</td>
<td></td>
</tr>
<tr>
<td>be consistent with</td>
<td>The chemotherapy orders must be consistent with the cancer diagnosis and the treatment plan, though what specific checks are done to establish consistency differs between these two artifacts.</td>
<td></td>
</tr>
<tr>
<td>be consistent with</td>
<td>For the cancer diagnosis, specific checks are done.</td>
<td></td>
</tr>
<tr>
<td>be consistent with</td>
<td>For the treatment plan, specific checks are done.</td>
<td></td>
</tr>
<tr>
<td>be consistent with</td>
<td>The chemotherapy orders must be consistent with the cancer diagnosis and the treatment plan, though what specific checks are done to establish consistency differs between these two artifacts.</td>
<td></td>
</tr>
<tr>
<td>be consistent with</td>
<td>For the cancer diagnosis, specific checks are done.</td>
<td></td>
</tr>
<tr>
<td>be consistent with</td>
<td>For the treatment plan, specific checks are done.</td>
<td></td>
</tr>
<tr>
<td>be consistent with</td>
<td>The chemotherapy orders must be consistent with the cancer diagnosis and the treatment plan, though what specific checks are done to establish consistency differs between these two artifacts.</td>
<td></td>
</tr>
<tr>
<td>be consistent with</td>
<td>For the cancer diagnosis, specific checks are done.</td>
<td></td>
</tr>
<tr>
<td>be consistent with</td>
<td>For the treatment plan, specific checks are done.</td>
<td></td>
</tr>
<tr>
<td>be consistent with</td>
<td>The chemotherapy orders must be consistent with the cancer diagnosis and the treatment plan, though what specific checks are done to establish consistency differs between these two artifacts.</td>
<td></td>
</tr>
<tr>
<td>be consistent with</td>
<td>For the cancer diagnosis, specific checks are done.</td>
<td></td>
</tr>
<tr>
<td>be consistent with</td>
<td>For the treatment plan, specific checks are done.</td>
<td></td>
</tr>
<tr>
<td>be consistent with</td>
<td>The chemotherapy orders must be consistent with the cancer diagnosis and the treatment plan, though what specific checks are done to establish consistency differs between these two artifacts.</td>
<td></td>
</tr>
<tr>
<td>be consistent with</td>
<td>For the cancer diagnosis, specific checks are done.</td>
<td></td>
</tr>
<tr>
<td>be consistent with</td>
<td>For the treatment plan, specific checks are done.</td>
<td></td>
</tr>
<tr>
<td>be consistent with</td>
<td>The chemotherapy orders must be consistent with the cancer diagnosis and the treatment plan, though what specific checks are done to establish consistency differs between these two artifacts.</td>
<td></td>
</tr>
<tr>
<td>be consistent with</td>
<td>For the cancer diagnosis, specific checks are done.</td>
<td></td>
</tr>
<tr>
<td>be consistent with</td>
<td>For the treatment plan, specific checks are done.</td>
<td></td>
</tr>
<tr>
<td>be consistent with</td>
<td>The chemotherapy orders must be consistent with the cancer diagnosis and the treatment plan, though what specific checks are done to establish consistency differs between these two artifacts.</td>
<td></td>
</tr>
<tr>
<td>be consistent with</td>
<td>For the cancer diagnosis, specific checks are done.</td>
<td></td>
</tr>
<tr>
<td>be consistent with</td>
<td>For the treatment plan, specific checks are done.</td>
<td></td>
</tr>
<tr>
<td>be consistent with</td>
<td>The chemotherapy orders must be consistent with the cancer diagnosis and the treatment plan, though what specific checks are done to establish consistency differs between these two artifacts.</td>
<td></td>
</tr>
<tr>
<td>be consistent with</td>
<td>For the cancer diagnosis, specific checks are done.</td>
<td></td>
</tr>
<tr>
<td>be consistent with</td>
<td>For the treatment plan, specific checks are done.</td>
<td></td>
</tr>
<tr>
<td>be consistent with</td>
<td>The chemotherapy orders must be consistent with the cancer diagnosis and the treatment plan, though what specific checks are done to establish consistency differs between these two artifacts.</td>
<td></td>
</tr>
<tr>
<td>be consistent with</td>
<td>For the cancer diagnosis, specific checks are done.</td>
<td></td>
</tr>
<tr>
<td>be consistent with</td>
<td>For the treatment plan, specific checks are done.</td>
<td></td>
</tr>
<tr>
<td>be consistent with</td>
<td>The chemotherapy orders must be consistent with the cancer diagnosis and the treatment plan, though what specific checks are done to establish consistency differs between these two artifacts.</td>
<td></td>
</tr>
<tr>
<td>be consistent with</td>
<td>For the cancer diagnosis, specific checks are done.</td>
<td></td>
</tr>
<tr>
<td>be consistent with</td>
<td>For the treatment plan, specific checks are done.</td>
<td></td>
</tr>
<tr>
<td>be consistent with</td>
<td>The chemotherapy orders must be consistent with the cancer diagnosis and the treatment plan, though what specific checks are done to establish consistency differs between these two artifacts.</td>
<td></td>
</tr>
<tr>
<td>be consistent with</td>
<td>For the cancer diagnosis, specific checks are done.</td>
<td></td>
</tr>
<tr>
<td>be consistent with</td>
<td>For the treatment plan, specific checks are done.</td>
<td></td>
</tr>
<tr>
<td>be consistent with</td>
<td>The chemotherapy orders must be consistent with the cancer diagnosis and the treatment plan, though what specific checks are done to establish consistency differs between these two artifacts.</td>
<td></td>
</tr>
<tr>
<td>be consistent with</td>
<td>For the cancer diagnosis, specific checks are done.</td>
<td></td>
</tr>
<tr>
<td>be consistent with</td>
<td>For the treatment plan, specific checks are done.</td>
<td></td>
</tr>
<tr>
<td>be consistent with</td>
<td>The chemotherapy orders must be consistent with the cancer diagnosis and the treatment plan, though what specific checks are done to establish consistency differs between these two artifacts.</td>
<td></td>
</tr>
<tr>
<td>be consistent with</td>
<td>For the cancer diagnosis, specific checks are done.</td>
<td></td>
</tr>
<tr>
<td>be consistent with</td>
<td>For the treatment plan, specific checks are done.</td>
<td></td>
</tr>
<tr>
<td>be consistent with</td>
<td>The chemotherapy orders must be consistent with the cancer diagnosis and the treatment plan, though what specific checks are done to establish consistency differs between these two artifacts.</td>
<td></td>
</tr>
<tr>
<td>be consistent with</td>
<td>For the cancer diagnosis, specific checks are done.</td>
<td></td>
</tr>
<tr>
<td>be consistent with</td>
<td>For the treatment plan, specific checks are done.</td>
<td></td>
</tr>
</tbody>
</table>

**critical**

- Every change not listed in the set of non-critical changes.
- Includes anything that introduces inconsistency between the chemotherapy orders and the treatment plan or between the chemotherapy drugs and the cancer diagnosis, and any significant changes to Patient data that are related to the chemotherapy orders.

**non-critical**

- A change that a Pharmacist can make to chemotherapy orders that does not require an Attending MD or Fellow MD to approve, and does not require re-approval of the treatment plan (and/or re-verification chemotherapy orders).
- Specifically, these are: a change in solution/dilution, a rounding off of dose (i.e., fixing obvious typographical errors), or a change in rates of administration.
<table>
<thead>
<tr>
<th>Term</th>
<th>Definition</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>chemotherapy orders</td>
<td>Directions for which chemotherapy drugs, at which amounts, and at what rates and intervals the chemotherapy should be administered. Also includes similar directions for administering pre-chemotherapy supportive care medications.</td>
<td>Must verify chemotherapy orders before administer, enter, and prepare. Note that chemotherapy orders are not themselves approved, but that a verification of them is required before the treatment plan can be approved.</td>
</tr>
<tr>
<td>contain Patient data</td>
<td></td>
<td></td>
</tr>
<tr>
<td>enter</td>
<td>Put chemotherapy orders into the computer system.</td>
<td>syn. &quot;write&quot; or &quot;generate&quot;. Never &quot;create&quot;.</td>
</tr>
<tr>
<td>match</td>
<td></td>
<td>The chemotherapy orders must exactly match the chemotherapy drug labels. Specific checks are done for: drug names, amounts, etc.</td>
</tr>
<tr>
<td>verify</td>
<td>Make sure that the chemotherapy orders are consistent with that Patient’s cancer diagnosis; consistent with the treatment plan; consistent with previous Patient chemotherapy administration data; consistent with that Patient’s up-to-date lab results; that the height and weight data in that Patient’s computer record are not stale and do not have known disparities; that the chemotherapy drugs’ doses on that Patient’s chemotherapy orders are correctly calculated for that Patient’s height and weight data; that the chemotherapy orders exactly match the chemotherapy drugs labels.</td>
<td>The Attending MD and Fellow MD do not verify chemotherapy orders, they only enter chemotherapy orders. The Practice RN, Pharmacy, &amp; Clinic RN verify chemotherapy orders. The influence of the various pieces of Patient data changes between the first cycle of chemotherapy and all subsequent cycles of chemotherapy. In subsequent cycles, stale height and weight is less significant.</td>
</tr>
<tr>
<td>chemotherapy teaching</td>
<td></td>
<td>A Patient can be taught about the chemotherapy that will be administered.</td>
</tr>
<tr>
<td>have</td>
<td></td>
<td></td>
</tr>
<tr>
<td>chemotherapy, course of</td>
<td>One or more cycles of chemotherapy, as defined by a treatment plan.</td>
<td>There is a 1:1 relationship between courses and treatment plans. An alternative definition (which is not used here) of a course of chemotherapy could be based on a Patient’s status and on the professional judgement of the MD(s) involved in that Patient’s care, and could span multiple treatment plans.</td>
</tr>
<tr>
<td>chemotherapy, cycle of</td>
<td>One cycle of chemotherapy includes one or more chemotherapy administration episodes. Each cycle is defined by a set of chemotherapy orders.</td>
<td>First cycle in a course of chemotherapy can also be thought of as &quot;prescribing chemotherapy&quot;, whereas all subsequent cycles aren’t prescriptions: they are continuations of the first prescription.</td>
</tr>
<tr>
<td>consent form for chemotherapy</td>
<td>The informed consent form for the Patient to receive chemotherapy.</td>
<td>There are other types of consent forms, such as consent to be seen by an Attending MD for a consult, and consent to release records for use by those in the chemotherapy process who need the information.</td>
</tr>
<tr>
<td>be for a particular treatment plan</td>
<td></td>
<td></td>
</tr>
<tr>
<td>be present in Patient record</td>
<td></td>
<td></td>
</tr>
<tr>
<td>be signed</td>
<td></td>
<td></td>
</tr>
<tr>
<td>sign</td>
<td></td>
<td></td>
</tr>
<tr>
<td>consult</td>
<td></td>
<td></td>
</tr>
<tr>
<td>have with Attending MD</td>
<td></td>
<td></td>
</tr>
<tr>
<td>follow-up appointment</td>
<td></td>
<td></td>
</tr>
<tr>
<td>schedule</td>
<td></td>
<td></td>
</tr>
<tr>
<td>pathology</td>
<td></td>
<td></td>
</tr>
<tr>
<td>be reviewed</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Patient condition</td>
<td></td>
<td></td>
</tr>
<tr>
<td>stabilize</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Term</td>
<td>Definition</td>
<td>Notes</td>
</tr>
<tr>
<td>-------------------------------------</td>
<td>---------------------------------------------------------------------------</td>
<td>----------------------------------------------------------------------</td>
</tr>
<tr>
<td>Patient data</td>
<td>A Patient's previous and current chemotherapy administration information, their height, their weight, and their laboratory results.</td>
<td>Also includes that Patient's first and last name, birth date, and medical record number.</td>
</tr>
<tr>
<td>be consistent with chemotherapy drug doses</td>
<td></td>
<td></td>
</tr>
<tr>
<td>be disparate</td>
<td>Have a problem or irregularity that is not consistent with the treatment plan and/or chemotherapy orders, including a copy, historical, or observed disparity.</td>
<td></td>
</tr>
<tr>
<td>be entered into Patient record</td>
<td></td>
<td></td>
</tr>
<tr>
<td>be recorded</td>
<td></td>
<td></td>
</tr>
<tr>
<td>be resolved</td>
<td></td>
<td></td>
</tr>
<tr>
<td>be stale / be up-to-date</td>
<td>The current data in a Patient’s record is more than one month old and has not yet been updated.</td>
<td></td>
</tr>
<tr>
<td>become disparate</td>
<td></td>
<td></td>
</tr>
<tr>
<td>become stale</td>
<td></td>
<td></td>
</tr>
<tr>
<td>chemotherapy administration information</td>
<td></td>
<td>Includes the dates when chemotherapy has been administered. the appropriate amount of time must pass between administration episodes. This check is only required in subsequent cycle.</td>
</tr>
<tr>
<td>• be previous</td>
<td></td>
<td></td>
</tr>
<tr>
<td>consult note</td>
<td></td>
<td></td>
</tr>
<tr>
<td>• be present in Patient record</td>
<td></td>
<td></td>
</tr>
<tr>
<td>height and/or weight</td>
<td></td>
<td></td>
</tr>
<tr>
<td>laboratory results</td>
<td>The results of tests ordered for the Patient. syn. &quot;Patient lab work&quot;</td>
<td>The only laboratory result necessary for approval of the treatment plan is the pathology report. The only laboratory result(s) necessary for verification of the chemotherapy orders are the blood counts and (optionally) the creatinine.</td>
</tr>
<tr>
<td>pathology report</td>
<td></td>
<td></td>
</tr>
<tr>
<td>• (not) indicate cancer diagnosis</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Patient disposition</td>
<td>decide</td>
<td></td>
</tr>
<tr>
<td>Patient record</td>
<td>have consult note</td>
<td>have signed consent form</td>
</tr>
<tr>
<td></td>
<td>have treatment plan</td>
<td></td>
</tr>
<tr>
<td>post-chemotherapy instructions</td>
<td>give to Patient</td>
<td></td>
</tr>
<tr>
<td>post-chemotherapy supportive care medication prescription</td>
<td>give to Patient</td>
<td></td>
</tr>
</tbody>
</table>
## Chemotherapy Property Glossary

<table>
<thead>
<tr>
<th>Term</th>
<th>Definition</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>supportive care medication</td>
<td>Comes in both pre-chemotherapy and post-chemotherapy types.</td>
<td></td>
</tr>
<tr>
<td>administer</td>
<td>Usually on the pre-chemotherapy supportive care medications are administered by Baystate.</td>
<td></td>
</tr>
<tr>
<td>be assigned to Patient</td>
<td></td>
<td></td>
</tr>
<tr>
<td>be consistent with chemotherapy drugs</td>
<td></td>
<td></td>
</tr>
<tr>
<td>be correctly prepared</td>
<td></td>
<td></td>
</tr>
<tr>
<td>be physically suitable for administration</td>
<td>Adhering to physical drug quality standards.</td>
<td></td>
</tr>
<tr>
<td>give prescriptions for post-chemotherapy</td>
<td></td>
<td></td>
</tr>
<tr>
<td>have adverse reaction to an administration of</td>
<td></td>
<td></td>
</tr>
<tr>
<td>match chemotherapy orders</td>
<td>All of the supportive care medication's data values must be the same as the associated values in the approved chemotherapy orders.</td>
<td>syn. “reconcile” in medical usage, but the MP's prefer to use “match” This is meant to imply that all supportive care medications in the chemotherapy orders must be prepared (i.e., no missing medications) and that all prepared supportive care medications must be in the chemotherapy orders (i.e., no extra medications).</td>
</tr>
<tr>
<td>prepare for Patient</td>
<td></td>
<td></td>
</tr>
<tr>
<td>stop administration of supportive care medication dose</td>
<td></td>
<td></td>
</tr>
<tr>
<td>be consistent with Patient's data</td>
<td></td>
<td></td>
</tr>
<tr>
<td>treatment</td>
<td>The administration of pre-chemotherapy supportive care medications and chemotherapy.</td>
<td></td>
</tr>
<tr>
<td>be well enough to receive</td>
<td></td>
<td></td>
</tr>
<tr>
<td>treatment plan</td>
<td>Directions for which drugs the Patient should receive. Used largely for legal and insurance purposes. This is the document that the Practice RN, Pharmacy, and Clinic RN must approve and sign.</td>
<td></td>
</tr>
<tr>
<td>approve</td>
<td>Check all associated data for correctness, consistency, and understandability, and if all data is correct, sign the document. syn. “certify”</td>
<td></td>
</tr>
<tr>
<td>• explicit</td>
<td>An indication of approval of the treatment plan. An Attending MD must explicitly approve a treatment plan that has been created by a Fellow MD or that has been changed by Pharmacy. For Practice RN, Pharmacy, &amp; Clinic RN: an explicit approval of the treatment plan</td>
<td></td>
</tr>
<tr>
<td>• implicit</td>
<td>A creation of the treatment plan. If that Attending MD creates a treatment plan, that creation is considered an implicit approval of the treatment plan.</td>
<td></td>
</tr>
<tr>
<td>be approved</td>
<td>syn. “be certified”</td>
<td></td>
</tr>
<tr>
<td>be changed</td>
<td></td>
<td></td>
</tr>
<tr>
<td>be consistent with</td>
<td>The treatment plan must be consistent with the cancer diagnosis and (to a lesser degree) the chemotherapy orders, though what specific checks are done to establish consistency differs between these two artifacts.</td>
<td></td>
</tr>
</tbody>
</table>

---

**Figure C.18.** Chemotherapy Case Study Term Glossary - Page 5
<table>
<thead>
<tr>
<th>Term</th>
<th>Definition</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>treatment plan</td>
<td>Directions for which drugs the Patient should receive. Used largely for legal and insurance purposes.</td>
<td>This is the document that the Practice RN, Pharmacy, and Clinic RN must approve and sign.</td>
</tr>
<tr>
<td>be consistent with</td>
<td>The treatment plan must be consistent with the cancer diagnosis and (to a lesser degree) the chemotherapy orders, though what specific checks are done to establish consistency differs between these two artifacts.</td>
<td>For the cancer diagnosis, specific checks are done. For the chemotherapy orders, specific checks are done.</td>
</tr>
<tr>
<td>be for a course of chemotherapy</td>
<td></td>
<td></td>
</tr>
<tr>
<td>be present in Patient record</td>
<td></td>
<td></td>
</tr>
<tr>
<td>change</td>
<td></td>
<td>All changes made to a treatment plan are considered ‘critical’ and they require the approval of an Attending MD and a new signed consent form from the Patient.</td>
</tr>
<tr>
<td>chemotherapy drug</td>
<td></td>
<td></td>
</tr>
<tr>
<td>create</td>
<td></td>
<td>For Attending MD: the creation of the treatment plan is an implicit approval of the treatment plan.</td>
</tr>
</tbody>
</table>
## C.2.3.2 Chemotherapy Role Glossary

### Chemotherapy Agent Glossary

<table>
<thead>
<tr>
<th>Term</th>
<th>Definition</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>attending MD</td>
<td>Approve critical changes to treatment plan or chemotherapy orders</td>
<td>Give an explicit approval of a treatment plan that has been created by a Fellow MD (as opposed to implicitly approving that treatment plan by creating it directly) or that has been changed by Pharmacy.</td>
</tr>
<tr>
<td>approve treatment plan</td>
<td>Change treatment plan or chemotherapy orders</td>
<td></td>
</tr>
<tr>
<td>change treatment plan or chemotherapy orders</td>
<td>Consult for Patient</td>
<td></td>
</tr>
<tr>
<td>consult for patient</td>
<td>Create treatment plan</td>
<td></td>
</tr>
<tr>
<td>create treatment plan</td>
<td>Decide pathology report does not indicate cancer diagnosis</td>
<td></td>
</tr>
<tr>
<td>decide pathology report does not indicate cancer diagnosis</td>
<td>Decide patient disposition</td>
<td></td>
</tr>
<tr>
<td>decide patient disposition</td>
<td>Enter chemotherapy orders</td>
<td></td>
</tr>
<tr>
<td>enter chemotherapy orders</td>
<td>Give prescriptions to patient</td>
<td></td>
</tr>
<tr>
<td>give prescriptions to patient</td>
<td>Verify chemotherapy orders</td>
<td></td>
</tr>
<tr>
<td>verify chemotherapy orders</td>
<td>attending MD delegate Either a Fellow MD working under the auspices of the Attending MD, or a Nurse Practitioner.</td>
<td></td>
</tr>
<tr>
<td>decide patient disposition</td>
<td>Clinic RN syn. &quot;Infusion suite nurse&quot;, &quot;treatment nurse&quot;</td>
<td></td>
</tr>
<tr>
<td>administrate chemotherapy</td>
<td>Administrator pre-chemotherapy supportive care medication</td>
<td></td>
</tr>
<tr>
<td>administer pre-chemotherapy</td>
<td>Approve treatment plan</td>
<td></td>
</tr>
<tr>
<td>approve treatment plan</td>
<td>Confirm presence of consult note in Patient record</td>
<td></td>
</tr>
<tr>
<td>confirm presence of consult note in Patient record</td>
<td>Confirm presence of signed consent form in Patient record</td>
<td></td>
</tr>
<tr>
<td>confirm presence of signed consent form in Patient record</td>
<td>Confirm presence of treatment plan in Patient record</td>
<td></td>
</tr>
<tr>
<td>confirm presence of treatment plan in Patient record</td>
<td>Correctly identify patient</td>
<td></td>
</tr>
<tr>
<td>correctly identify patient</td>
<td>Discharge patient</td>
<td></td>
</tr>
<tr>
<td>discharge patient</td>
<td>Do chemotherapy teaching</td>
<td></td>
</tr>
<tr>
<td>do chemotherapy teaching</td>
<td>Enter all patient data into that patient’s record</td>
<td></td>
</tr>
</tbody>
</table>

### Figure C.20. Chemotherapy Case Study Role Glossary - Page 1
<table>
<thead>
<tr>
<th>Term</th>
<th>Definition</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Clinic RN</td>
<td>syn. “Infusion suite nurse”, “treatment nurse”</td>
<td>enter all Patient data into that Patient’s record on the same business day, give post-chemotherapy instructions to Patient, give post-chemotherapy supportive care medication prescriptions to Patient, make critical changes to treatment plan or chemotherapy orders, make sure chemotherapy orders and treatment plan are consistent with Patient cancer diagnosis, make sure Patient has appropriate I.V. access, make sure Patient is well enough to receive chemotherapy, observe Patient, schedule follow-up appointment for Patient, verify chemotherapy orders</td>
</tr>
<tr>
<td>Fellow MD</td>
<td>An oncology or hematology MD in training.</td>
<td>approve critical changes to treatment plan or chemotherapy orders, change treatment plan or chemotherapy orders, create treatment plan (This treatment plan must be approved by an Attending MD.), decide Patient disposition, enter chemotherapy orders (Enter chemotherapy orders into the computer system.), verify chemotherapy orders</td>
</tr>
<tr>
<td>medical staff</td>
<td></td>
<td>approve treatment plan (Attending MD, but only if a Fellow MD creates the treatment plan.), be qualified (Be a Baystate employee in the appropriate position for each of the actions that can be taken.), create treatment plan (Attending MD or Fellow MD.), enter chemotherapy orders (Attending MD or Fellow MD.), verify cancer diagnosis (Pathologist), verify chemotherapy orders (Practice RN, Pharmacy, Clinic RN.)</td>
</tr>
<tr>
<td>Term</td>
<td>Definition</td>
<td>Notes</td>
</tr>
<tr>
<td>-------------------------</td>
<td>---------------------------------------------------------------------------</td>
<td>----------------------------------------------------------------------</td>
</tr>
<tr>
<td>Nurse Practitioner</td>
<td>decide Patient disposition</td>
<td></td>
</tr>
<tr>
<td></td>
<td>do chemotherapy teaching</td>
<td></td>
</tr>
<tr>
<td>Pathologist</td>
<td>review pathology</td>
<td></td>
</tr>
<tr>
<td>Patient</td>
<td>administer chemotherapy drugs to</td>
<td></td>
</tr>
<tr>
<td></td>
<td>administer pre-chemotherapy supportive care medication to</td>
<td></td>
</tr>
<tr>
<td></td>
<td>be assessed</td>
<td></td>
</tr>
<tr>
<td></td>
<td>be correctly identified</td>
<td></td>
</tr>
<tr>
<td></td>
<td>be discharged</td>
<td></td>
</tr>
<tr>
<td></td>
<td>be given support for safely handling the outcome of an chemotherapy</td>
<td></td>
</tr>
<tr>
<td></td>
<td>be observed</td>
<td></td>
</tr>
<tr>
<td></td>
<td>be well enough to be discharged</td>
<td>The review of systems indicates that the Patient is well enough to be discharged.</td>
</tr>
<tr>
<td></td>
<td>be well enough to receive treatment</td>
<td>The review of systems and the laboratory test results indicate that the Patient is well enough to receive treatment.</td>
</tr>
<tr>
<td></td>
<td>give informed consent</td>
<td>This is different than signing the consent form: the signing is just the legal representation of the Patient having given informed consent.</td>
</tr>
<tr>
<td></td>
<td>give post-chemotherapy instructions to</td>
<td></td>
</tr>
<tr>
<td></td>
<td>give prescriptions to</td>
<td></td>
</tr>
<tr>
<td></td>
<td>have adverse reaction to chemotherapy drug</td>
<td></td>
</tr>
<tr>
<td></td>
<td>have adverse reaction to pre-chemotherapy supportive care medication</td>
<td></td>
</tr>
<tr>
<td></td>
<td>have appropriate I.V. access</td>
<td>The Patient must have a port-a-cath or a peripheral I.V. line access.</td>
</tr>
<tr>
<td></td>
<td>have chemotherapy administered to</td>
<td></td>
</tr>
<tr>
<td></td>
<td>have chemotherapy teaching</td>
<td>syn. &quot;be taught about chemotherapy&quot;</td>
</tr>
<tr>
<td></td>
<td>have consult with Attending MD</td>
<td></td>
</tr>
<tr>
<td></td>
<td>need observation</td>
<td></td>
</tr>
<tr>
<td></td>
<td>schedule follow-up appointment for</td>
<td></td>
</tr>
<tr>
<td></td>
<td>sign consent form</td>
<td></td>
</tr>
<tr>
<td>Pharmacy</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### Chemotherapy Agent Glossary

<table>
<thead>
<tr>
<th>Term</th>
<th>Definition</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>approve treatment plan</td>
<td></td>
<td></td>
</tr>
<tr>
<td>make critical changes to treatment plan or chemotherapy orders</td>
<td></td>
<td></td>
</tr>
<tr>
<td>make sure chemotherapy drug doses are consistent with Patient data</td>
<td></td>
<td></td>
</tr>
<tr>
<td>make sure chemotherapy orders and treatment plan are consistent with Patient cancer diagnosis</td>
<td></td>
<td></td>
</tr>
<tr>
<td>make sure Patient height and weight data are not stale</td>
<td></td>
<td></td>
</tr>
<tr>
<td>make sure Patient height and weight data do not have known disparities</td>
<td></td>
<td></td>
</tr>
<tr>
<td>make sure Patient is well enough to receive chemotherapy</td>
<td></td>
<td></td>
</tr>
<tr>
<td>verify chemotherapy orders</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Practice RN</strong></td>
<td>Nurse assigned to up to three Attending MDs. Responsible for a lightweight consistency and completeness check of the treatment plan and chemotherapy orders.</td>
<td></td>
</tr>
<tr>
<td>approve treatment plan</td>
<td></td>
<td></td>
</tr>
<tr>
<td>make critical changes to treatment plan or chemotherapy orders</td>
<td></td>
<td></td>
</tr>
<tr>
<td>make sure chemotherapy orders and treatment plan are consistent with Patient cancer diagnosis</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

*Figure C.23. Chemotherapy Case Study Role Glossary - Page 4*
C.2.4 Formal Property Specifications

Event alphabet:
- A: Baystate Pathologist reviews pathology
- B: administer chemotherapy

SCOPE:
1. From the start of any event sequence through to the end of that event sequence, the behavior must hold.

BEHAVIOR:
1. The events of primary interest in this behavior are Baystate Pathologist reviews pathology and administer chemotherapy.
2. There are no events of secondary interest in this behavior.
3. administer chemotherapy is not allowed to occur until after Baystate Pathologist reviews pathology occurs.
4. Baystate Pathologist reviews pathology is required to occur, whether or not administer chemotherapy eventually occurs.
5. administer chemotherapy is not required to occur after Baystate Pathologist reviews pathology occurs.
6. After Baystate Pathologist reviews pathology occurs, but before the first subsequent administer chemotherapy occurs, Baystate Pathologist reviews pathology is allowed to occur again, zero or more times.
7. After Baystate Pathologist reviews pathology and the first subsequent administer chemotherapy occur:
   - Both Baystate Pathologist reviews pathology and administer chemotherapy are allowed to occur again, zero or more times, and there are no restrictions on the occurrences of any future events.

FSA and DNL for Property Chemo.A.1
Event alphabet:
- A: Patient has a consult with a Baystate Attending MD
- B: administer chemotherapy

**SCOPE:**
1. From the start of any event sequence through to the end of that event sequence, the behavior must hold.

**BEHAVIOR:**
1. The events of primary interest in this behavior are Patient has a consult with a Baystate Attending MD and administer chemotherapy.
2. There are no events of secondary interest in this behavior.
3. administer chemotherapy is not allowed to occur until after Patient has a consult with a Baystate Attending MD occurs.
4. Patient has a consult with a Baystate Attending MD is required to occur, whether or not administer chemotherapy eventually occurs.
5. administer chemotherapy is not required to occur after Patient has a consult with a Baystate Attending MD occurs.
6. After Patient has a consult with a Baystate Attending MD occurs, but before the first subsequent administer chemotherapy occurs, Patient has a consult with a Baystate Attending MD is allowed to occur again, zero or more times.
7. After Patient has a consult with a Baystate Attending MD and the first subsequent administer chemotherapy occur:
   - Both Patient has a consult with a Baystate Attending MD and administer chemotherapy are allowed to occur again, zero or more times, and there are no restrictions on the occurrences of any future events.

FSA and DNL for Property Chemo.A.2
Event alphabet:
- A: administer chemotherapy
- **START**: Attending MD decides no cancer diagnosis

**SCOPE:**
1. There can be at most one restricted interval in the event sequence and it has a starting delimiter, **Attending MD decides no cancer diagnosis**.
2. The behavior is required to hold from the first occurrence of **Attending MD decides no cancer diagnosis** through to the end of the event sequence. Even if **Attending MD decides no cancer diagnosis** occurs more than once before the end of the event sequence, only the first occurrence of **Attending MD decides no cancer diagnosis** begins the restricted interval; later occurrences of **Attending MD decides no cancer diagnosis** do not have an effect.
3. **Attending MD decides no cancer diagnosis** is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence.
4. There are no restrictions imposed on the occurrences of any events before the first occurrence of **Attending MD decides no cancer diagnosis**, if it ever occurs.

**BEHAVIOR:**
1. The event of primary interest in this behavior is **administer chemotherapy**.
2. There are no events of secondary interest in this behavior.
3. **administer chemotherapy** is never allowed to occur.

FSA and DNL for Property Chemo.A.3
Event alphabet:
- A: Patient has a consult with a Baystate Attending MD
- B: Patient signs a consent form

SCOPE:
1. From the start of any event sequence through to the end of that event sequence, the behavior must hold.

BEHAVIOR:
1. The events of primary interest in this behavior are Patient has a consult with a Baystate Attending MD and Patient signs a consent form.
2. There are no events of secondary interest in this behavior.
3. Patient signs a consent form is not allowed to occur until after Patient has a consult with a Baystate Attending MD occurs.
4. Patient has a consult with a Baystate Attending MD is required to occur, whether or not Patient signs a consent form eventually occurs.
5. Patient signs a consent form is not required to occur after Patient has a consult with a Baystate Attending MD occurs.
6. After Patient has a consult with a Baystate Attending MD occurs, but before the first subsequent Patient signs a consent form occurs, Patient has a consult with a Baystate Attending MD is allowed to occur again, zero or more times.
7. After Patient has a consult with a Baystate Attending MD and the first subsequent Patient signs a consent form occur:
   - Both Patient has a consult with a Baystate Attending MD and Patient signs a consent form are allowed to occur again, zero or more times, and there are no restrictions on the occurrences of any future events.

FSA and DNL for Property Chemo.B.6
Event alphabet:
- A: Patient has chemotherapy teaching
- B: Patient signs consent form for new treatment plan

SCOPE:
1. From the start of any event sequence through to the end of that event sequence, the behavior must hold.

BEHAVIOR:
1. The events of primary interest in this behavior are Patient has chemotherapy teaching and Patient signs consent form for new treatment plan.
2. There are no events of secondary interest in this behavior.
3. Patient signs consent form for new treatment plan is not allowed to occur until after Patient has chemotherapy teaching occurs.
4. Patient has chemotherapy teaching is not required to occur.
5. Even if Patient has chemotherapy teaching does occur, Patient signs consent form for new treatment plan is not required to occur after Patient has chemotherapy teaching occurs.
6. After Patient has chemotherapy teaching occurs, but before the first subsequent Patient signs consent form for new treatment plan occurs, Patient has chemotherapy teaching is allowed to occur again, zero or more times.
7. After Patient has chemotherapy teaching and the first subsequent Patient signs consent form for new treatment plan occur:
   - Patient signs consent form for new treatment plan is not allowed to occur again until after another Patient has chemotherapy teaching occurs; Patient has chemotherapy teaching is allowed to occur again and, if it does, then the situation is the same as when the first Patient has chemotherapy teaching occurred, meaning that the restrictions described in parts 5, 6, and 7 would again apply.

FSA and DNL for Property Chemo.B.7
Event alphabet:

- A: Patient has a consult with a Baystate Attending MD
- B: Patient has chemotherapy teaching

SCOPE:
1. From the start of any event sequence through to the end of that event sequence, the behavior must hold.

BEHAVIOR:
1. The events of primary interest in this behavior are Patient has a consult with a Baystate Attending MD and Patient has chemotherapy teaching.
2. There are no events of secondary interest in this behavior.
3. Patient has chemotherapy teaching is not allowed to occur until after Patient has a consult with a Baystate Attending MD occurs.
4. Patient has a consult with a Baystate Attending MD is required to occur, whether or not Patient has chemotherapy teaching eventually occurs.
5. Patient has chemotherapy teaching is not required to occur after Patient has a consult with a Baystate Attending MD occurs.
6. After Patient has a consult with a Baystate Attending MD occurs, but before the first subsequent Patient has chemotherapy teaching occurs, Patient has a consult with a Baystate Attending MD is allowed to occur again, zero or more times.
7. After Patient has a consult with a Baystate Attending MD and the first subsequent Patient has chemotherapy teaching occur:
   - Both Patient has a consult with a Baystate Attending MD and Patient has chemotherapy teaching are allowed to occur again, zero or more times, and there are no restrictions on the occurrences of any future events.

FSA and DNL for Property Chemo.B.8
Event alphabet:
- A: consult note is put into that Patient’s record
- B: administer chemotherapy
- C: consult note is removed from that Patient’s record

SCOPE:
1. From the start of any event sequence through to the end of that event sequence, the behavior must hold.

BEHAVIOR:
1. The events of primary interest in this behavior are consult note is put into that Patient’s record and administer chemotherapy.
2. The event of secondary interest in this behavior is consult note is removed from that Patient’s record.
3. administer chemotherapy is not allowed to occur until after consult note is put into that Patient’s record occurs.
4. Before the first consult note is put into that Patient’s record occurs, consult note is removed from that Patient’s record is allowed to occur zero or more times.
5. consult note is put into that Patient’s record is required to occur, whether or not administer chemotherapy eventually occurs.
6. administer chemotherapy is not required to occur after consult note is put into that Patient’s record occurs.
7. After consult note is put into that Patient’s record occurs, but before the first subsequent administer chemotherapy occurs:
   - consult note is put into that Patient’s record is allowed to occur again, zero or more times;
   - consult note is removed from that Patient’s record is allowed to occur zero or more times.
8. After consult note is put into that Patient’s record and the first subsequent administer chemotherapy occur:
   - consult note is removed from that Patient’s record is allowed to occur zero or more times;
   - Both consult note is put into that Patient’s record and administer chemotherapy are allowed to occur again, zero or more times, and there are no restrictions on the occurrences of any future events.

FSA and DNL for Property Chemo.B.1a

365
Event alphabet:
- \( A \): consult note is put into that Patient’s record
- \( \text{START} \): consult note is removed from that Patient’s record
- \( \text{END} \): administer chemotherapy

**SCOPE:**
1. A restricted interval in the event sequence can have both a starting delimiter, consult note is removed from that Patient’s record, and an ending delimiter, administer chemotherapy.
2. The behavior is required to hold from an occurrence of consult note is removed from that Patient’s record, if it ever occurs, through to the first subsequent occurrence of administer chemotherapy, if it ever occurs.
3. If there are multiple occurrences of consult note is removed from that Patient’s record without an occurrence of administer chemotherapy in between them, only the last of those occurrences of consult note is removed from that Patient’s record starts a restricted interval; each of those occurrences of consult note is removed from that Patient’s record resets the beginning of this restricted interval.
4. Consult note is removed from that Patient’s record is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if consult note is removed from that Patient’s record does occur, administer chemotherapy is not required to occur subsequently. Even if administer chemotherapy does not occur subsequently, the behavior is still required to hold, until the end of the event sequence.
5. There might be many restricted intervals in the event sequence. If consult note is removed from that Patient’s record occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of consult note is removed from that Patient’s record, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

**BEHAVIOR:**
1. The event of primary interest in this behavior is consult note is put into that Patient’s record.
2. There are no events of secondary interest in this behavior.
3. Consult note is put into that Patient’s record is required to occur at least once.

FSA and DNL for Property Chemo.B.1b
Event alphabet:
- A: Patient signs consent form
- B: administer chemotherapy

**SCOPE:**
1. From the start of any event sequence through to the end of that event sequence, the behavior must hold.

**BEHAVIOR:**
1. The events of primary interest in this behavior are Patient signs consent form and administer chemotherapy.
2. There are no events of secondary interest in this behavior.
3. administer chemotherapy is not allowed to occur until after Patient signs consent form occurs.
4. Patient signs consent form is not required to occur.
5. Even if Patient signs consent form does occur, administer chemotherapy is not required to occur after Patient signs consent form occurs.
6. After Patient signs consent form occurs, but before the first subsequent administer chemotherapy occurs, Patient signs consent form is allowed to occur again, zero or more times.
7. After Patient signs consent form and the first subsequent administer chemotherapy occur:
   - Both Patient signs consent form and administer chemotherapy are allowed to occur again, zero or more times, and there are no restrictions on the occurrences of any future events.

FSA and DNL for Property Chemo.B.2
**Event alphabet:**
- A: signed consent form is put into that Patient’s record
- B: administer chemotherapy
- C: signed consent form is removed from that Patient’s record

**SCOPE:**
1. From the start of any event sequence through to the end of that event sequence, the behavior must hold.

**BEHAVIOR:**
1. The events of primary interest in this behavior are signed consent form is put into that Patient’s record and administer chemotherapy.
2. The event of secondary interest in this behavior is signed consent form is removed from that Patient’s record.
3. administer chemotherapy is not allowed to occur until after signed consent form is put into that Patient’s record occurs.
4. Before the first signed consent form is put into that Patient’s record occurs, signed consent form is removed from that Patient’s record is allowed to occur zero or more times.
5. After signed consent form is put into that Patient’s record occurs, signed consent form is put into that Patient’s record is not required to occur.
6. Even if signed consent form is put into that Patient’s record occurs, administer chemotherapy is not required to occur after signed consent form is put into that Patient’s record occurs.
7. After signed consent form is put into that Patient’s record occurs, but before the first subsequent administer chemotherapy occurs:
   - signed consent form is put into that Patient’s record is allowed to occur again, zero or more times;
   - signed consent form is removed from that Patient’s record is allowed to occur zero or more times.
8. After signed consent form is put into that Patient’s record and the first subsequent administer chemotherapy occur:
   - signed consent form is removed from that Patient’s record is allowed to occur zero or more times;
   - Both signed consent form is put into that Patient’s record and administer chemotherapy are allowed to occur again, zero or more times, and there are no restrictions on the occurrences of any future events.

**FSA and DNL for Property Chemo.B.3a**
SCAPE:
1. A restricted interval in the event sequence can have both a starting delimiter, signed consent form is removed from that Patient’s record, and an ending delimiter, administer chemotherapy.
2. The behavior is required to hold from an occurrence of signed consent form is removed from that Patient’s record, if it ever occurs, through to the first subsequent occurrence of administer chemotherapy, if it ever occurs.
3. If there are multiple occurrences of signed consent form is removed from that Patient’s record without an occurrence of administer chemotherapy in between them, only the last of those occurrences of signed consent form is removed from that Patient’s record potentially starts a restricted interval; each of those occurrences of signed consent form is removed from that Patient’s record resets the beginning of this restricted interval.
4. signed consent form is removed from that Patient’s record is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if signed consent form is removed from that Patient’s record does occur, administer chemotherapy is not required to occur subsequently and if it does not occur subsequently, then the behavior is not required to hold for the remainder of the event sequence.
5. There might be many restricted intervals in the event sequence. If signed consent form is removed from that Patient’s record occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of signed consent form is removed from that Patient’s record, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a potential new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

BEHAVIOR:
1. The event of primary interest in this behavior is signed consent form is put into that Patient’s record.
2. There are no events of secondary interest in this behavior.
3. signed consent form is put into that Patient’s record is required to occur at least once.

FSA and DNL for Property Chemo.B.3b
Event alphabet:
- A: treatment plan is put into that Patient’s record
- B: administer chemotherapy
- C: treatment plan is removed from that Patient’s record

SCOPE:
1. From the start of any event sequence through to the end of that event sequence, the behavior must hold.

BEHAVIOR:
1. The events of primary interest in this behavior are treatment plan is put into that Patient’s record and administer chemotherapy.
2. The event of secondary interest in this behavior is treatment plan is removed from that Patient’s record.
3. administer chemotherapy is not allowed to occur until after treatment plan is put into that Patient’s record occurs.
4. Before the first treatment plan is put into that Patient’s record occurs, treatment plan is removed from that Patient’s record is allowed to occur zero or more times.
5. treatment plan is put into that Patient’s record is not required to occur.
6. Even if treatment plan is put into that Patient’s record does occur, administer chemotherapy is not required to occur after treatment plan is put into that Patient’s record occurs.
7. After treatment plan is put into that Patient’s record occurs, but before the first subsequent administer chemotherapy occurs:
   - treatment plan is put into that Patient’s record is allowed to occur again, zero or more times;
   - treatment plan is removed from that Patient’s record is allowed to occur zero or more times.
8. After treatment plan is put into that Patient’s record and the first subsequent administer chemotherapy occur:
   - treatment plan is removed from that Patient’s record is allowed to occur zero or more times;
   - Both treatment plan is put into that Patient’s record and administer chemotherapy are allowed to occur again, zero or more times, and there are no restrictions on the occurrences of any future events.

FSA and DNL for Property Chemo.B.4a
Event alphabet:
- \( A \): treatment plan is put into that Patient’s record
- \( \text{START} \): treatment plan is removed from that Patient’s record
- \( \text{END} \): administer chemotherapy

**SCOPE:**
1. A restricted interval in the event sequence can have both a starting delimiter, \( \text{treatment plan is removed from that Patient’s record} \), and an ending delimiter, \( \text{administer chemotherapy} \).
2. The behavior is required to hold from an occurrence of \( \text{treatment plan is removed from that Patient’s record} \), if it ever occurs, through to the first subsequent occurrence of \( \text{administer chemotherapy} \), if it ever occurs.
3. If there are multiple occurrences of \( \text{treatment plan is removed from that Patient’s record} \) without an occurrence of \( \text{administer chemotherapy} \) in between them, only the last of those occurrences of \( \text{treatment plan is removed from that Patient’s record} \) potentially starts a restricted interval; each of those occurrences of \( \text{treatment plan is removed from that Patient’s record} \) resets the beginning of this restricted interval.
4. \( \text{treatment plan is removed from that Patient’s record} \) is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if \( \text{treatment plan is removed from that Patient’s record} \) does occur, \( \text{administer chemotherapy} \) is not required to occur subsequently and if it does not occur subsequently, then the behavior is not required to hold for the remainder of the event sequence.
5. There might be many restricted intervals in the event sequence. If \( \text{treatment plan is removed from that Patient’s record} \) occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of \( \text{treatment plan is removed from that Patient’s record} \), meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a potential new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

**BEHAVIOR:**
1. The event of primary interest in this behavior is \( \text{treatment plan is put into that Patient’s record} \).
2. There are no events of secondary interest in this behavior.
3. \( \text{treatment plan is put into that Patient’s record} \) is required to occur exactly once.

FSA and DNL for Property Chemo.B.4b
**Scope:**

1. A restricted interval in the event sequence can have both a starting delimiter, *chemotherapy drugs in the treatment plan change OR new treatment plan is created*, and an ending delimiter, *administer chemotherapy*.
2. The behavior is required to hold from an occurrence of *chemotherapy drugs in the treatment plan change OR new treatment plan is created*, if it ever occurs, through to the first subsequent occurrence of *administer chemotherapy*, if it ever occurs.
3. If there are multiple occurrences of *chemotherapy drugs in the treatment plan change OR new treatment plan is created* without an occurrence of *administer chemotherapy* in between them, only the last of those occurrences of *chemotherapy drugs in the treatment plan change OR new treatment plan is created* potentially starts the restricted interval; each of those occurrences of *chemotherapy drugs in the treatment plan change OR new treatment plan is created* resets the beginning of the restricted interval.
4. If chemotherapy drugs in the treatment plan change OR new treatment plan is created does occur, administer chemotherapy is not required to occur subsequently and if it does not occur subsequently, then the behavior is not required to hold for the remainder of the event sequence.
5. There can be at most one restricted interval in the event sequence.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

**Behavior:**

1. The event of primary interest in this behavior is *Patient signs consent form for new treatment plan*.
2. There are no events of secondary interest in this behavior.
3. *Patient signs consent form for new treatment plan* is required to occur at least once.

---

**FSA and DNL for Property Chemo.B.5**
Event alphabet:

- **A**: treatment plan and chemotherapy orders become inconsistent
- **B**: resolve inconsistency between treatment plan and chemotherapy orders
- **C**: Clinic RN verifies chemotherapy orders, Pharmacy verifies chemotherapy orders

**SCOPE:**

1. There can be at most one restricted interval in the event sequence and it has an ending delimiter, Practice RN OR second Clinic RN verifies chemotherapy orders.
2. The behavior is required to hold from the start of the event sequence through to the first occurrence of Practice RN OR second Clinic RN verifies chemotherapy orders, if it ever occurs.
3. Practice RN OR second Clinic RN verifies chemotherapy orders is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence.
4. There are no restrictions imposed on the occurrences of any events after the first occurrence of Practice RN OR second Clinic RN verifies chemotherapy orders, if it ever occurs.

**BEHAVIOR:**

1. The events of primary interest in this behavior are treatment plan and chemotherapy orders become inconsistent and resolve inconsistency between treatment plan and chemotherapy orders.
2. The events of secondary interest in this behavior are Clinic RN verifies chemotherapy orders and Pharmacy verifies chemotherapy orders.
3. If treatment plan and chemotherapy orders become inconsistent occurs, resolve inconsistency between treatment plan and chemotherapy orders is required to occur subsequently.
4. Before the first treatment plan and chemotherapy orders become inconsistent occurs:
   - resolve inconsistency between treatment plan and chemotherapy orders is not allowed to occur;
   - All the events of secondary interest are allowed to occur zero or more times.
5. treatment plan and chemotherapy orders become inconsistent is not required to occur.
6. After treatment plan and chemotherapy orders become inconsistent occurs, but before the first subsequent resolve inconsistency between treatment plan and chemotherapy orders occurs:
   - treatment plan and chemotherapy orders become inconsistent is allowed to occur again, zero or more times;
   - All the events of secondary interest are allowed to occur zero or more times.
7. After treatment plan and chemotherapy orders become inconsistent and the first subsequent resolve inconsistency between treatment plan and chemotherapy orders occur:
   - All the events of secondary interest are allowed to occur zero or more times;
   - resolve inconsistency between treatment plan and chemotherapy orders is not allowed to occur again until after another treatment plan and chemotherapy orders become inconsistent occurs, treatment plan and chemotherapy orders become inconsistent is allowed to occur again and, if it does, then the situation is the same as when the first treatment plan and chemotherapy orders become inconsistent occurred, meaning that the restrictions described in parts 3, 6, and 7 would again apply.

FSA and DNL for Property Chemo.C.11a
Event alphabet:

- **A**: resolve inconsistency between treatment plan and chemotherapy orders
- **START**: treatment plan and chemotherapy orders become inconsistent
- **END**: Pharmacy verifies chemotherapy orders
- **C**: Clinic RN verifies chemotherapy orders, Practice RN OR second Clinic RN verifies chemotherapy orders

**SCOPE:**

1. A restricted interval in the event sequence can have both a starting delimiter, treatment plan and chemotherapy orders become inconsistent, and an ending delimiter, Pharmacy verifies chemotherapy orders.
2. The behavior is required to hold from an occurrence of treatment plan and chemotherapy orders become inconsistent, if it ever occurs, through to the first subsequent occurrence of Pharmacy verifies chemotherapy orders, if it ever occurs.
3. If there are multiple occurrences of treatment plan and chemotherapy orders become inconsistent without an occurrence of Pharmacy verifies chemotherapy orders in between them, only the last of those occurrences of treatment plan and chemotherapy orders become inconsistent potentially starts a restricted interval; each of those occurrences of treatment plan and chemotherapy orders become inconsistent resets the beginning of this restricted interval.
4. treatment plan and chemotherapy orders become inconsistent is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if treatment plan and chemotherapy orders become inconsistent does occur, Pharmacy verifies chemotherapy orders is not required to occur subsequently and if it does not occur subsequently, then the behavior is not required to hold for the remainder of the event sequence.
5. There might be many restricted intervals in the event sequence. If treatment plan and chemotherapy orders become inconsistent occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of treatment plan and chemotherapy orders become inconsistent, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a potential new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

**BEHAVIOR:**

1. The event of primary interest in this behavior is resolve inconsistency between treatment plan and chemotherapy orders.
2. The events of secondary interest in this behavior are Clinic RN verifies chemotherapy orders and Practice RN OR second Clinic RN verifies chemotherapy orders.
3. resolve inconsistency between treatment plan and chemotherapy orders is required to occur at least once.

FSA and DNL for Property Chemo.C.11b
Event alphabet:
- A: resolve inconsistency between treatment plan and chemotherapy orders
- START: treatment plan and chemotherapy orders become inconsistent
- END: Clinic RN verifies chemotherapy orders
- C: Pharmacy verifies chemotherapy orders, Practice RN OR second Clinic RN verifies chemotherapy orders

**SCOPE:**
1. A restricted interval in the event sequence can have both a starting delimiter, treatment plan and chemotherapy orders become inconsistent, and an ending delimiter, Clinic RN verifies chemotherapy orders.
2. The behavior is required to hold from an occurrence of treatment plan and chemotherapy orders become inconsistent, if it ever occurs, through to the first subsequent occurrence of Clinic RN verifies chemotherapy orders, if it ever occurs.
3. If there are multiple occurrences of treatment plan and chemotherapy orders become inconsistent without an occurrence of Clinic RN verifies chemotherapy orders in between them, only the last of those occurrences of treatment plan and chemotherapy orders become inconsistent potentially starts a restricted interval; each of those occurrences of treatment plan and chemotherapy orders become inconsistent resets the beginning of this restricted interval.
4. treatment plan and chemotherapy orders become inconsistent is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if treatment plan and chemotherapy orders become inconsistent does occur, Clinic RN verifies chemotherapy orders is not required to occur subsequently and if it does not occur subsequently, then the behavior is not required to hold for the remainder of the event sequence.
5. There might be many restricted intervals in the event sequence. If treatment plan and chemotherapy orders become inconsistent occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of treatment plan and chemotherapy orders become inconsistent, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a potential new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

**BEHAVIOR:**
1. The event of primary interest in this behavior is resolve inconsistency between treatment plan and chemotherapy orders.
2. The events of secondary interest in this behavior are Pharmacy verifies chemotherapy orders and Practice RN OR second Clinic RN verifies chemotherapy orders.
3. resolve inconsistency between treatment plan and chemotherapy orders is required to occur at least once.
**Event alphabet:**
- **A**: put inappropriate chemotherapy drugs in the chemotherapy orders
- **B**: put only appropriate chemotherapy drugs in the chemotherapy orders
- **END**: Practice RN OR second Clinic RN verifies chemotherapy orders
- **C**: Clinic RN verifies chemotherapy orders, Pharmacy verifies chemotherapy orders

**SCOPE:**
1. There can be at most one restricted interval in the event sequence and it has an ending delimiter, Practice RN OR second Clinic RN verifies chemotherapy orders.
2. The behavior is required to hold from the start of the event sequence through to the first occurrence of Practice RN OR second Clinic RN verifies chemotherapy orders, if it ever occurs.
3. Practice RN OR second Clinic RN verifies chemotherapy orders is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence.
4. There are no restrictions imposed on the occurrences of any events after the first occurrence of Practice RN OR second Clinic RN verifies chemotherapy orders, if it ever occurs.

**BEHAVIOR:**
1. The events of primary interest in this behavior are put inappropriate chemotherapy drugs in the chemotherapy orders and put only appropriate chemotherapy drugs in the chemotherapy orders.
2. The events of secondary interest in this behavior are Clinic RN verifies chemotherapy orders and Pharmacy verifies chemotherapy orders.
3. If put inappropriate chemotherapy drugs in the chemotherapy orders occurs, put only appropriate chemotherapy drugs in the chemotherapy orders is required to occur subsequently.
4. Before the first put inappropriate chemotherapy drugs in the chemotherapy orders occurs:
   - put only appropriate chemotherapy drugs in the chemotherapy orders is allowed to occur zero or more times;
   - All the events of secondary interest are allowed to occur zero or more times.
5. put inappropriate chemotherapy drugs in the chemotherapy orders is not required to occur.
6. After put inappropriate chemotherapy drugs in the chemotherapy orders occurs, but before the first subsequent put only appropriate chemotherapy drugs in the chemotherapy orders occurs:
   - put inappropriate chemotherapy drugs in the chemotherapy orders is allowed to occur again, zero or more times;
   - None of the events of secondary interest are allowed to occur.
7. After put inappropriate chemotherapy drugs in the chemotherapy orders and the first subsequent put only appropriate chemotherapy drugs in the chemotherapy orders occur:
   - All the events of secondary interest are allowed to occur zero or more times;
   - put inappropriate chemotherapy drugs in the chemotherapy orders is allowed to occur again, zero or more times, before another put inappropriate chemotherapy drugs in the chemotherapy orders occurs; put inappropriate chemotherapy drugs in the chemotherapy orders is allowed to occur again and, if it does, then the situation is the same as when the first put inappropriate chemotherapy drugs in the chemotherapy orders occurred, meaning that the restrictions described in parts 3, 6, and 7 would again apply.

FSA and DNL for Property Chemo.C.12a
Event alphabet:
- A: put only appropriate chemotherapy drugs in the chemotherapy orders
- START: put inappropriate chemotherapy drugs in the chemotherapy orders
- END: Pharmacy verifies chemotherapy orders
- C: Clinic RN verifies chemotherapy orders, Practice RN OR second Clinic RN verifies chemotherapy orders

**SCOPE:**
1. A restricted interval in the event sequence can have both a starting delimiter, put inappropriate chemotherapy drugs in the chemotherapy orders, and an ending delimiter, Pharmacy verifies chemotherapy orders.
2. The behavior is required to hold from an occurrence of put inappropriate chemotherapy drugs in the chemotherapy orders, if it ever occurs, through to the first subsequent occurrence of Pharmacy verifies chemotherapy orders, if it ever occurs.
3. If there are multiple occurrences of put inappropriate chemotherapy drugs in the chemotherapy orders without an occurrence of Pharmacy verifies chemotherapy orders in between them, only the last of those occurrences of put inappropriate chemotherapy drugs in the chemotherapy orders potentially starts a restricted interval; each of those occurrences of put inappropriate chemotherapy drugs in the chemotherapy orders resets the beginning of this restricted interval.
4. put inappropriate chemotherapy drugs in the chemotherapy orders is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if put inappropriate chemotherapy drugs in the chemotherapy orders does occur, Pharmacy verifies chemotherapy orders is not required to occur subsequently and if it does not occur subsequently, then the behavior is not required to hold for the remainder of the event sequence.
5. There might be many restricted intervals in the event sequence. If put inappropriate chemotherapy drugs in the chemotherapy orders occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of put inappropriate chemotherapy drugs in the chemotherapy orders, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a potential new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

**BEHAVIOR:**
1. The event of primary interest in this behavior is put only appropriate chemotherapy drugs in the chemotherapy orders.
2. The events of secondary interest in this behavior are Clinic RN verifies chemotherapy orders and Practice RN OR second Clinic RN verifies chemotherapy orders.
3. put only appropriate chemotherapy drugs in the chemotherapy orders is required to occur at least once.

FSA and DNL for Property Chemo.C.12b
Event alphabet:
- A: put only appropriate chemotherapy drugs in the chemotherapy orders
- START: put inappropriate chemotherapy drugs in the chemotherapy orders
- END: Clinic RN verifies chemotherapy orders
- C: Pharmacy verifies chemotherapy orders, Practice RN OR second Clinic RN verifies chemotherapy orders

**SCOPE:**
1. A restricted interval in the event sequence can have both a starting delimiter, put inappropriate chemotherapy drugs in the chemotherapy orders, and an ending delimiter, Clinic RN verifies chemotherapy orders.
2. The behavior is required to hold from an occurrence of put inappropriate chemotherapy drugs in the chemotherapy orders, if it ever occurs, through to the first subsequent occurrence of Clinic RN verifies chemotherapy orders, if it ever occurs.
3. If there are multiple occurrences of put inappropriate chemotherapy drugs in the chemotherapy orders without an occurrence of Clinic RN verifies chemotherapy orders in between them, only the last of those occurrences of put inappropriate chemotherapy drugs in the chemotherapy orders potentially starts a restricted interval; each of those occurrences of put inappropriate chemotherapy drugs in the chemotherapy orders resets the beginning of this restricted interval.
4. put inappropriate chemotherapy drugs in the chemotherapy orders is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if put inappropriate chemotherapy drugs in the chemotherapy orders does occur, Clinic RN verifies chemotherapy orders is not required to occur subsequently and if it does not occur subsequently, then the behavior is not required to hold for the remainder of the event sequence.
5. There might be many restricted intervals in the event sequence. If put inappropriate chemotherapy drugs in the chemotherapy orders occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of put inappropriate chemotherapy drugs in the chemotherapy orders, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a potential new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

**BEHAVIOR:**
1. The event of primary interest in this behavior is put only appropriate chemotherapy drugs in the chemotherapy orders.
2. The events of secondary interest in this behavior are Pharmacy verifies chemotherapy orders and Practice RN OR second Clinic RN verifies chemotherapy orders.
3. put only appropriate chemotherapy drugs in the chemotherapy orders is required to occur at least once.

FSA and DNL for Property Chemo.C.12c
Event alphabet:

- A: put inappropriate chemotherapy drug doses in chemotherapy orders
- B: put only appropriate chemotherapy drugs dosages in chemotherapy orders
- C: Clinic RN verifies chemotherapy orders, Pharmacy verifies chemotherapy orders
- END: Practice RN OR second Clinic RN verifies chemotherapy orders

SCOPE:
1. There can be at most one restricted interval in the event sequence and it has an ending delimiter, Practice RN OR second Clinic RN verifies chemotherapy orders.
2. The behavior is required to hold from the start of the event sequence through to the first occurrence of Practice RN OR second Clinic RN verifies chemotherapy orders, if it ever occurs.
3. Practice RN OR second Clinic RN verifies chemotherapy orders is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence.
4. There are no restrictions imposed on the occurrences of any events after the first occurrence of Practice RN OR second Clinic RN verifies chemotherapy orders, if it ever occurs.

BEHAVIOR:
1. The events of primary interest in this behavior are put inappropriate chemotherapy drug doses in chemotherapy orders and put only appropriate chemotherapy drugs dosages in chemotherapy orders.
2. The events of secondary interest in this behavior are Clinic RN verifies chemotherapy orders and Pharmacy verifies chemotherapy orders.
3. If put inappropriate chemotherapy drug doses in chemotherapy orders occurs, put only appropriate chemotherapy drugs dosages in chemotherapy orders is required to occur subsequently.
4. Before the first put inappropriate chemotherapy drug doses in chemotherapy orders occurs:
   - put only appropriate chemotherapy drugs dosages in chemotherapy orders is allowed to occur zero or more times;
   - All the events of secondary interest are allowed to occur zero or more times.
5. put inappropriate chemotherapy drug doses in chemotherapy orders is not required to occur.
6. After put inappropriate chemotherapy drug doses in chemotherapy orders occurs, but before the first subsequent put only appropriate chemotherapy drugs dosages in chemotherapy orders occurs:
   - put inappropriate chemotherapy drug doses in chemotherapy orders is allowed to occur again, zero or more times;
   - None of the events of secondary interest are allowed to occur.
7. After put inappropriate chemotherapy drug doses in chemotherapy orders and the first subsequent put only appropriate chemotherapy drugs dosages in chemotherapy orders occur:
   - All the events of secondary interest are allowed to occur zero or more times;
   - put only appropriate chemotherapy drugs dosages in chemotherapy orders is allowed to occur again, zero or more times, before another put inappropriate chemotherapy drug doses in chemotherapy orders occurs: put inappropriate chemotherapy drug doses in chemotherapy orders is allowed to occur again and, if it does, then the situation is the same as when the first put inappropriate chemotherapy drug doses in chemotherapy orders occurred, meaning that the restrictions described in parts 3, 6, and 7 would again apply.

FSA and DNL for Property Chemo.C.13a
Event alphabet:

- **A**: put only appropriate chemotherapy drugs dosages in chemotherapy orders
- **START**: put inappropriate chemotherapy drug doses in chemotherapy orders
- **END**: Pharmacy verifies chemotherapy orders
- **C**: Clinic RN verifies chemotherapy orders, Practice RN OR second Clinic RN verifies chemotherapy orders

**SCOPE:**

1. A restricted interval in the event sequence can have both a starting delimiter, put inappropriate chemotherapy drug doses in chemotherapy orders, and an ending delimiter, Pharmacy verifies chemotherapy orders.
2. The behavior is required to hold from an occurrence of put inappropriate chemotherapy drug doses in chemotherapy orders, if it ever occurs, through to the first subsequent occurrence of Pharmacy verifies chemotherapy orders, if it ever occurs.
3. If there are multiple occurrences of put inappropriate chemotherapy drug doses in chemotherapy orders without an occurrence of Pharmacy verifies chemotherapy orders in between them, only the last of those occurrences of put inappropriate chemotherapy drug doses in chemotherapy orders potentially starts a restricted interval; each of those occurrences of put inappropriate chemotherapy drug doses in chemotherapy orders resets the beginning of this restricted interval.
4. put inappropriate chemotherapy drug doses in chemotherapy orders is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if put inappropriate chemotherapy drug doses in chemotherapy orders does occur, Pharmacy verifies chemotherapy orders is not required to occur subsequently and if it does not occur subsequently, then the behavior is not required to hold for the remainder of the event sequence.
5. There might be many restricted intervals in the event sequence. If put inappropriate chemotherapy drug doses in chemotherapy orders occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of put inappropriate chemotherapy drug doses in chemotherapy orders, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a potential new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

**BEHAVIOR:**

1. The event of primary interest in this behavior is put only appropriate chemotherapy drugs dosages in chemotherapy orders.
2. The events of secondary interest in this behavior are Clinic RN verifies chemotherapy orders and Practice RN OR second Clinic RN verifies chemotherapy orders.
3. put only appropriate chemotherapy drugs dosages in chemotherapy orders is required to occur at least once.
Event alphabet:
- A: put only appropriate chemotherapy drugs dosages in chemotherapy orders
- START: put inappropriate chemotherapy drug doses in chemotherapy orders
- END: Clinic RN verifies chemotherapy orders
- C: Pharmacy verifies chemotherapy orders, Practice RN OR second Clinic RN verifies chemotherapy orders

SCOPE:
1. A restricted interval in the event sequence can have both a starting delimiter, put inappropriate chemotherapy drug doses in chemotherapy orders, and an ending delimiter, Clinic RN verifies chemotherapy orders.
2. The behavior is required to hold from an occurrence of put inappropriate chemotherapy drug doses in chemotherapy orders, if it ever occurs, through to the first subsequent occurrence of Clinic RN verifies chemotherapy orders, if it ever occurs.
3. If there are multiple occurrences of put inappropriate chemotherapy drug doses in chemotherapy orders without an occurrence of Clinic RN verifies chemotherapy orders in between them, only the last of those occurrences of put inappropriate chemotherapy drug doses in chemotherapy orders potentially starts a restricted interval; each of those occurrences of put inappropriate chemotherapy drug doses in chemotherapy orders resets the beginning of this restricted interval.
4. put inappropriate chemotherapy drug doses in chemotherapy orders is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if put inappropriate chemotherapy drug doses in chemotherapy orders does occur, Clinic RN verifies chemotherapy orders is not required to occur subsequently and if it does not occur subsequently, then the behavior is not required to hold for the remainder of the event sequence.
5. There might be many restricted intervals in the event sequence. If put inappropriate chemotherapy drug doses in chemotherapy orders occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of put inappropriate chemotherapy drug doses in chemotherapy orders, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a potential new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

BEHAVIOR:
1. The event of primary interest in this behavior is put only appropriate chemotherapy drugs dosages in chemotherapy orders.
2. The events of secondary interest in this behavior are Pharmacy verifies chemotherapy orders and Practice RN OR second Clinic RN verifies chemotherapy orders.
3. put only appropriate chemotherapy drugs dosages in chemotherapy orders is required to occur at least once.

FSA and DNL for Property Chemo.C.13c
**Event alphabet:**
- A: Patient data in chemotherapy orders becomes stale or disparate
- B: resolve stale or disparate Patient data in chemotherapy orders
- END: Practice RN OR second Clinic RN verifies chemotherapy orders
- C: Clinic RN verifies chemotherapy orders, Pharmacy verifies chemotherapy orders

**SCOPE:**
1. There can be at most one restricted interval in the event sequence and it has an ending delimiter, Practice RN OR second Clinic RN verifies chemotherapy orders.
2. The behavior is required to hold from the start of the event sequence through to the first occurrence of Practice RN OR second Clinic RN verifies chemotherapy orders, if it ever occurs.
3. Practice RN OR second Clinic RN verifies chemotherapy orders is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence.
4. There are no restrictions imposed on the occurrences of any events after the first occurrence of Practice RN OR second Clinic RN verifies chemotherapy orders, if it ever occurs.

**BEHAVIOR:**
1. The events of primary interest in this behavior are Patient data in chemotherapy orders becomes stale or disparate and resolve stale or disparate Patient data in chemotherapy orders.
2. The events of secondary interest in this behavior are Clinic RN verifies chemotherapy orders and Pharmacy verifies chemotherapy orders.
3. If Patient data in chemotherapy orders becomes stale or disparate occurs, resolve stale or disparate Patient data in chemotherapy orders is required to occur subsequently.
4. Before the first Patient data in chemotherapy orders becomes stale or disparate occurs:
   - resolve stale or disparate Patient data in chemotherapy orders is not allowed to occur;
   - All the events of secondary interest are allowed to occur zero or more times.
5. Patient data in chemotherapy orders becomes stale or disparate is not required to occur.
6. After Patient data in chemotherapy orders becomes stale or disparate occurs, but before the first subsequent resolve stale or disparate Patient data in chemotherapy orders occurs:
   - Patient data in chemotherapy orders becomes stale or disparate is allowed to occur again, zero or more times;
7. After Patient data in chemotherapy orders becomes stale or disparate and the first subsequent resolve stale or disparate Patient data in chemotherapy orders occur:
   - All the events of secondary interest are allowed to occur zero or more times;
   - None of the events of secondary interest are allowed to occur.

FSA and DNL for Property Chemo.C.14a
Event alphabet:
- **A**: resolve stale or disparate Patient data in chemotherapy orders
- **START**: Patient data in chemotherapy orders becomes stale or disparate
- **END**: Pharmacy verifies chemotherapy orders
- **C**: Clinic RN verifies chemotherapy orders, Practice RN OR second Clinic RN verifies chemotherapy orders

**SCOPE:**
1. A restricted interval in the event sequence can have both a starting delimiter, Patient data in chemotherapy orders becomes stale or disparate, and an ending delimiter, Pharmacy verifies chemotherapy orders.
2. The behavior is required to hold from an occurrence of Patient data in chemotherapy orders becomes stale or disparate, if it ever occurs, through to the first subsequent occurrence of Pharmacy verifies chemotherapy orders, if it ever occurs.
3. If there are multiple occurrences of Patient data in chemotherapy orders becomes stale or disparate without an occurrence of Pharmacy verifies chemotherapy orders in between them, only the last of those occurrences of Patient data in chemotherapy orders becomes stale or disparate potentially starts a restricted interval; each of those occurrences of Patient data in chemotherapy orders becomes stale or disparate resets the beginning of this restricted interval.
4. Patient data in chemotherapy orders becomes stale or disparate is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if Patient data in chemotherapy orders becomes stale or disparate does occur, Pharmacy verifies chemotherapy orders is not required to occur subsequently and if it does not occur subsequently, then the behavior is not required to hold for the remainder of the event sequence.
5. There might be many restricted intervals in the event sequence. If Patient data in chemotherapy orders becomes stale or disparate occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of Patient data in chemotherapy orders becomes stale or disparate, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a potential new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

**BEHAVIOR:**
1. The event of primary interest in this behavior is resolve stale or disparate Patient data in chemotherapy orders.
2. The events of secondary interest in this behavior are Clinic RN verifies chemotherapy orders and Practice RN OR second Clinic RN verifies chemotherapy orders.
3. resolve stale or disparate Patient data in chemotherapy orders is required to occur at least once.

FSA and DNL for Property Chemo.C.14b
**Event alphabet:**
- A: resolve stale or disparate Patient data in chemotherapy orders
- START: Patient data in chemotherapy orders becomes stale or disparate
- END: Clinic RN verifies chemotherapy orders
- C: Pharmacy verifies chemotherapy orders, Practice RN OR second Clinic RN verifies chemotherapy orders

**SCOPE:**
1. A restricted interval in the event sequence can have both a starting delimiter, Patient data in chemotherapy orders becomes stale or disparate, and an ending delimiter, Clinic RN verifies chemotherapy orders.
2. The behavior is required to hold from an occurrence of Patient data in chemotherapy orders becomes stale or disparate, if it ever occurs, through to the first subsequent occurrence of Clinic RN verifies chemotherapy orders, if it ever occurs.
3. If there are multiple occurrences of Patient data in chemotherapy orders becomes stale or disparate without an occurrence of Clinic RN verifies chemotherapy orders in between them, only the last of those occurrences of Patient data in chemotherapy orders becomes stale or disparate potentially starts a restricted interval; each of those occurrences of Patient data in chemotherapy orders becomes stale or disparate resets the beginning of this restricted interval.
4. Patient data in chemotherapy orders becomes stale or disparate is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if Patient data in chemotherapy orders becomes stale or disparate does occur, Clinic RN verifies chemotherapy orders is not required to occur subsequently and if it does not occur subsequently, then the behavior is not required to hold for the remainder of the event sequence.
5. There might be many restricted intervals in the event sequence. If Patient data in chemotherapy orders becomes stale or disparate occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of Patient data in chemotherapy orders becomes stale or disparate, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a potential new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

**BEHAVIOR:**
1. The event of primary interest in this behavior is resolve stale or disparate Patient data in chemotherapy orders.
2. The events of secondary interest in this behavior are Pharmacy verifies chemotherapy orders and Practice RN OR second Clinic RN verifies chemotherapy orders.
3. resolve stale or disparate Patient data in chemotherapy orders is required to occur at least once.

FSA and DNL for Property Chemo.C.14c
Event alphabet:
- A: treatment plan and chemotherapy orders become inconsistent
- B: resolve inconsistency between treatment plan and chemotherapy orders
- START: Fellow MD creates treatment plan and/or chemotherapy orders
- END: Attending MD approves treatment plan

SCOPE:
1. A restricted interval in the event sequence can have both a starting delimiter, Fellow MD creates treatment plan and/or chemotherapy orders, and an ending delimiter, Attending MD approves treatment plan.
2. The behavior is required to hold from an occurrence of Fellow MD creates treatment plan and/or chemotherapy orders, if it ever occurs, through to the first subsequent occurrence of Attending MD approves treatment plan, if it ever occurs.
3. If there are multiple occurrences of Fellow MD creates treatment plan and/or chemotherapy orders without an occurrence of Attending MD approves treatment plan in between them, only the last of those occurrences of Fellow MD creates treatment plan and/or chemotherapy orders potentially starts the restricted interval; each of those occurrences of Fellow MD creates treatment plan and/or chemotherapy orders resets the beginning of the restricted interval.
4. Fellow MD creates treatment plan and/or chemotherapy orders is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if Fellow MD creates treatment plan and/or chemotherapy orders does occur, Attending MD approves treatment plan is not required to occur subsequently and if it does not occur subsequently, then the behavior is not required to hold for the remainder of the event sequence.
5. There can be at most one restricted interval in the event sequence.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

BEHAVIOR:
1. The events of primary interest in this behavior are treatment plan and chemotherapy orders become inconsistent and resolve inconsistency between treatment plan and chemotherapy orders.
2. There are no events of secondary interest in this behavior.
3. If treatment plan and chemotherapy orders become inconsistent occurs, resolve inconsistency between treatment plan and chemotherapy orders is required to occur subsequently.
4. Before the first treatment plan and chemotherapy orders become inconsistent occurs, resolve inconsistency between treatment plan and chemotherapy orders is not allowed to occur.
5. treatment plan and chemotherapy orders become inconsistent is not required to occur.
6. After treatment plan and chemotherapy orders become inconsistent occurs, but before the first subsequent resolve inconsistency between treatment plan and chemotherapy orders occurs, treatment plan and chemotherapy orders become inconsistent is allowed to occur again, zero or more times.
7. After treatment plan and chemotherapy orders become inconsistent and the first subsequent resolve inconsistency between treatment plan and chemotherapy orders occur:
   - resolve inconsistency between treatment plan and chemotherapy orders is not allowed to occur again until after another treatment plan and chemotherapy orders become inconsistent occurs. treatment plan and chemotherapy orders become inconsistent is allowed to occur again and, if it does, then the situation is the same as when the first treatment plan and chemotherapy orders become inconsistent occurred, meaning that the restrictions described in parts 3, 6, and 7 would again apply.

FSA and DNL for Property Chemo.C.6a
Event alphabet:
- **A**: treatment plan and chemotherapy orders become inconsistent
- **B**: resolve inconsistency between treatment plan and chemotherapy orders
- **END**: Practice RN OR second Clinic RN approves treatment plan
- **C**: Clinic RN approves treatment plan, Pharmacy approves treatment plan

**SCOPE:**
1. There can be at most one restricted interval in the event sequence and it has an ending delimiter, Practice RN OR second Clinic RN approves treatment plan.
2. The behavior is required to hold from the start of the event sequence through to the first occurrence of Practice RN OR second Clinic RN approves treatment plan, if it ever occurs.
3. Practice RN OR second Clinic RN approves treatment plan is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence.
4. There are no restrictions imposed on the occurrences of any events after the first occurrence of Practice RN OR second Clinic RN approves treatment plan, if it ever occurs.

**BEHAVIOR:**
1. The events of primary interest in this behavior are treatment plan and chemotherapy orders become inconsistent and resolve inconsistency between treatment plan and chemotherapy orders.
2. The events of secondary interest in this behavior are Clinic RN approves treatment plan and Pharmacy approves treatment plan.
3. If treatment plan and chemotherapy orders become inconsistent occurs, resolve inconsistency between treatment plan and chemotherapy orders is required to occur subsequently.
4. Before the first treatment plan and chemotherapy orders become inconsistent occurs:
   - resolve inconsistency between treatment plan and chemotherapy orders is not allowed to occur;
   - All the events of secondary interest are allowed to occur zero or more times.
5. treatment plan and chemotherapy orders become inconsistent is not required to occur.
6. After treatment plan and chemotherapy orders become inconsistent occurs, but before the first subsequent resolve inconsistency between treatment plan and chemotherapy orders occurs:
   - treatment plan and chemotherapy orders become inconsistent is allowed to occur again, zero or more times;
   - None of the events of secondary interest are allowed to occur.
7. After treatment plan and chemotherapy orders become inconsistent and the first subsequent resolve inconsistency between treatment plan and chemotherapy orders occurs:
   - All the events of secondary interest are allowed to occur zero or more times;
   - resolve inconsistency between treatment plan and chemotherapy orders is not allowed to occur again until after another treatment plan and chemotherapy orders become inconsistent occurs; treatment plan and chemotherapy orders become inconsistent is allowed to occur again and, if it does, then the situation is the same as when the first treatment plan and chemotherapy orders become inconsistent occurred, meaning that the restrictions described in parts 3, 6, and 7 would again apply.

FSA and DNL for Property Chemo.C.6b
**Event alphabet:**
- **A**: resolve inconsistency between treatment plan and chemotherapy orders
- **START**: treatment plan and chemotherapy orders become inconsistent
- **END**: Pharmacy approves treatment plan
- **C**: Clinic RN approves treatment plan, Practice RN OR second Clinic RN approves treatment plan

**SCOPE:**
1. A restricted interval in the event sequence can have both a starting delimiter, treatment plan and chemotherapy orders become inconsistent, and an ending delimiter, Pharmacy approves treatment plan. The behavior is required to hold from an occurrence of treatment plan and chemotherapy orders become inconsistent, if it ever occurs, through to the first subsequent occurrence of Pharmacy approves treatment plan, if it ever occurs.
2. If there are multiple occurrences of treatment plan and chemotherapy orders become inconsistent without an occurrence of Pharmacy approves treatment plan in between them, only the last of those occurrences of treatment plan and chemotherapy orders become inconsistent potentially starts a restricted interval; each of those occurrences of treatment plan and chemotherapy orders become inconsistent resets the beginning of this restricted interval.
3. If there are multiple occurrences of treatment plan and chemotherapy orders become inconsistent without an occurrence of Pharmacy approves treatment plan in between them, only the last of those occurrences of treatment plan and chemotherapy orders become inconsistent potentially starts a restricted interval; each of those occurrences of treatment plan and chemotherapy orders become inconsistent resets the beginning of this restricted interval.
4. treatment plan and chemotherapy orders become inconsistent is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if treatment plan and chemotherapy orders become inconsistent does occur, Pharmacy approves treatment plan is not required to occur subsequently and if it does not occur subsequently, then the behavior is not required to hold for the remainder of the event sequence.
5. There might be many restricted intervals in the event sequence. If treatment plan and chemotherapy orders become inconsistent occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of treatment plan and chemotherapy orders become inconsistent, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a potential new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

**BEHAVIOR:**
1. The event of primary interest in this behavior is resolve inconsistency between treatment plan and chemotherapy orders.
2. The events of secondary interest in this behavior are Clinic RN approves treatment plan and Practice RN OR second Clinic RN approves treatment plan.
3. resolve inconsistency between treatment plan and chemotherapy orders is required to occur at least once.

**FSA and DNL for Property Chemo.C.6c**
**Event alphabet:**

- **A**: resolve inconsistency between treatment plan and chemotherapy orders
- **START**: treatment plan and chemotherapy orders become inconsistent
- **END**: Clinic RN approves treatment plan
- **C**: Pharmacy approves treatment plan, Practice RN OR second Clinic RN approves treatment plan

**SCOPE:**

1. A restricted interval in the event sequence can have both a starting delimiter, **treatment plan and chemotherapy orders become inconsistent**, and an ending delimiter, **Clinic RN approves treatment plan**.
2. The behavior is required to hold from an occurrence of **treatment plan and chemotherapy orders become inconsistent**, if it ever occurs, through to the first subsequent occurrence of **Clinic RN approves treatment plan**, if it ever occurs.
3. If there are multiple occurrences of **treatment plan and chemotherapy orders become inconsistent** without an occurrence of **Clinic RN approves treatment plan** in between them, only the last of those occurrences of **treatment plan and chemotherapy orders become inconsistent** potentially starts a restricted interval; each of those occurrences of **treatment plan and chemotherapy orders become inconsistent** resets the beginning of this restricted interval.
4. **treatment plan and chemotherapy orders become inconsistent** is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if **treatment plan and chemotherapy orders become inconsistent** does occur, **Clinic RN approves treatment plan** is not required to occur subsequently and if it does not occur subsequently, then the behavior is not required to hold for the remainder of the event sequence.
5. There might be many restricted intervals in the event sequence. If **treatment plan and chemotherapy orders become inconsistent** occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of **treatment plan and chemotherapy orders become inconsistent**, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a potential new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

**BEHAVIOR:**

1. The event of primary interest in this behavior is **resolve inconsistency between treatment plan and chemotherapy orders**.
2. The events of secondary interest in this behavior are **Pharmacy approves treatment plan** and **Practice RN OR second Clinic RN approves treatment plan**.
3. **resolve inconsistency between treatment plan and chemotherapy orders** is required to occur at least once.

FSA and DNL for Property Chemo.C.6d
Event alphabet:

- A: put inappropriate chemotherapy drugs in the treatment plan
- B: put only appropriate chemotherapy drugs in the treatment plan
- START: Fellow MD creates treatment plan and/or chemotherapy orders
- END: Attending MD approves treatment plan

SCOPE:
1. A restricted interval in the event sequence can have both a starting delimiter, Fellow MD creates treatment plan and/or chemotherapy orders, and an ending delimiter, Attending MD approves treatment plan.
2. The behavior is required to hold from an occurrence of Fellow MD creates treatment plan and/or chemotherapy orders, if it ever occurs, through to the first subsequent occurrence of Attending MD approves treatment plan, if it ever occurs.
3. If there are multiple occurrences of Fellow MD creates treatment plan and/or chemotherapy orders without an occurrence of Attending MD approves treatment plan in between them, only the last of those occurrences of Fellow MD creates treatment plan and/or chemotherapy orders potentially starts the restricted interval; each of those occurrences of Fellow MD creates treatment plan and/or chemotherapy orders resets the beginning of the restricted interval.
4. Fellow MD creates treatment plan and/or chemotherapy orders is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if Fellow MD creates treatment plan and/or chemotherapy orders does occur, Attending MD approves treatment plan is not required to occur subsequently and if it does not occur subsequently, then the behavior is not required to hold for the remainder of the event sequence.
5. There can be at most one restricted interval in the event sequence.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

BEHAVIOR:
1. The events of primary interest in this behavior are put inappropriate chemotherapy drugs in the treatment plan and put only appropriate chemotherapy drugs in the treatment plan.
2. There are no events of secondary interest in this behavior.
3. If put inappropriate chemotherapy drugs in the treatment plan occurs, put only appropriate chemotherapy drugs in the treatment plan is required to occur subsequently.
4. Before the first put inappropriate chemotherapy drugs in the treatment plan occurs, put only appropriate chemotherapy drugs in the treatment plan is allowed to occur zero or more times.
5. put inappropriate chemotherapy drugs in the treatment plan is not required to occur.
6. After put inappropriate chemotherapy drugs in the treatment plan occurs, but before the first subsequent put only appropriate chemotherapy drugs in the treatment plan occurs, put inappropriate chemotherapy drugs in the treatment plan is allowed to occur again, zero or more times.
7. After put inappropriate chemotherapy drugs in the treatment plan and the first subsequent put only appropriate chemotherapy drugs in the treatment plan occur:
   - put only appropriate chemotherapy drugs in the treatment plan is allowed to occur again, zero or more times, before another put inappropriate chemotherapy drugs in the treatment plan occurs; put inappropriate chemotherapy drugs in the treatment plan is allowed to occur again and, if it does, then the situation is the same as when the first put inappropriate chemotherapy drugs in the treatment plan occurred, meaning that the restrictions described in parts 3, 6, and 7 would again apply.

FSA and DNL for Property Chemo.C.7a
**Event alphabet:**
- **A**: put inappropriate chemotherapy drugs in the treatment plan
- **B**: put only appropriate chemotherapy drugs in the treatment plan
- **C**: Clinic RN approves treatment plan, Pharmacy approves treatment plan
- **END**: Practice RN OR second Clinic RN approves treatment plan

---

**SCOPE:**
1. There can be at most one restricted interval in the event sequence and it has an ending delimiter, Practice RN OR second Clinic RN approves treatment plan.
2. The behavior is required to hold from the start of the event sequence through to the first occurrence of Practice RN OR second Clinic RN approves treatment plan, if it ever occurs.
3. Practice RN OR second Clinic RN approves treatment plan is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence.
4. There are no restrictions imposed on the occurrences of any events after the first occurrence of Practice RN OR second Clinic RN approves treatment plan, if it ever occurs.

**BEHAVIOR:**
1. The events of primary interest in this behavior are put inappropriate chemotherapy drugs in the treatment plan and put only appropriate chemotherapy drugs in the treatment plan.
2. The events of secondary interest in this behavior are Clinic RN approves treatment plan and Pharmacy approves treatment plan.
3. If put inappropriate chemotherapy drugs in the treatment plan occurs, put only appropriate chemotherapy drugs in the treatment plan is required to occur subsequently.
4. Before the first put inappropriate chemotherapy drugs in the treatment plan occurs:
   - put only appropriate chemotherapy drugs in the treatment plan is allowed to occur zero or more times;
   - All the events of secondary interest are allowed to occur zero or more times.
5. Put inappropriate chemotherapy drugs in the treatment plan is not required to occur.
6. After put inappropriate chemotherapy drugs in the treatment plan occurs, but before the first subsequent put only appropriate chemotherapy drugs in the treatment plan occurs:
   - put inappropriate chemotherapy drugs in the treatment plan is allowed to occur again, zero or more times;
   - None of the events of secondary interest are allowed to occur.
7. After put inappropriate chemotherapy drugs in the treatment plan and the first subsequent put only appropriate chemotherapy drugs in the treatment plan occur:
   - All the events of secondary interest are allowed to occur zero or more times;
   - put only appropriate chemotherapy drugs in the treatment plan is allowed to occur again, zero or more times, before another put inappropriate chemotherapy drugs in the treatment plan occurs; put inappropriate chemotherapy drugs in the treatment plan is allowed to occur again and, if it does, then the situation is the same as when the first put inappropriate chemotherapy drugs in the treatment plan occurred, meaning that the restrictions described in parts 3, 6, and 7 would again apply.

---

**FSA and DNL for Property Chemo.C.7b**
**Event alphabet:**
- A: put only appropriate chemotherapy drugs in the treatment plan
- START: put inappropriate chemotherapy drugs in the treatment plan
- END: Pharmacy approves treatment plan
- C: Clinic RN approves treatment plan, Practice RN OR second Clinic RN approves treatment plan

**SCOPE:**
1. A restricted interval in the event sequence can have both a starting delimiter, put inappropriate chemotherapy drugs in the treatment plan, and an ending delimiter, Pharmacy approves treatment plan.
2. The behavior is required to hold from an occurrence of put inappropriate chemotherapy drugs in the treatment plan, if it ever occurs, through to the first subsequent occurrence of Pharmacy approves treatment plan, if it ever occurs.
3. If there are multiple occurrences of put inappropriate chemotherapy drugs in the treatment plan without an occurrence of Pharmacy approves treatment plan in between them, only the last of those occurrences of put inappropriate chemotherapy drugs in the treatment plan potentially starts a restricted interval; each of those occurrences of put inappropriate chemotherapy drugs in the treatment plan resets the beginning of this restricted interval.
4. put inappropriate chemotherapy drugs in the treatment plan is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if put inappropriate chemotherapy drugs in the treatment plan does occur, Pharmacy approves treatment plan is not required to occur subsequently and if it does not occur subsequently, then the behavior is not required to hold for the remainder of the event sequence.
5. There might be many restricted intervals in the event sequence. If put inappropriate chemotherapy drugs in the treatment plan occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of put inappropriate chemotherapy drugs in the treatment plan, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a potential new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

**BEHAVIOR:**
1. The event of primary interest in this behavior is put only appropriate chemotherapy drugs in the treatment plan.
2. The events of secondary interest in this behavior are Clinic RN approves treatment plan and Practice RN OR second Clinic RN approves treatment plan.
3. put only appropriate chemotherapy drugs in the treatment plan is required to occur at least once.

FSA and DNL for Property Chemo.C.7c
Event alphabet:
- A: put only appropriate chemotherapy drugs in the treatment plan
- START: put inappropriate chemotherapy drugs in the treatment plan
- END: Clinic RN approves treatment plan
- C: Pharmacy approves treatment plan, Practice RN OR second Clinic RN approves treatment plan

SCOPE:
1. A restricted interval in the event sequence can have both a starting delimiter, put inappropriate chemotherapy drugs in the treatment plan, and an ending delimiter, Clinic RN approves treatment plan.
2. The behavior is required to hold from an occurrence of put inappropriate chemotherapy drugs in the treatment plan, if it ever occurs, through to the first subsequent occurrence of Clinic RN approves treatment plan, if it ever occurs.
3. If there are multiple occurrences of put inappropriate chemotherapy drugs in the treatment plan without an occurrence of Clinic RN approves treatment plan in between them, only the last of those occurrences of put inappropriate chemotherapy drugs in the treatment plan potentially starts a restricted interval; each of those occurrences of put inappropriate chemotherapy drugs in the treatment plan resets the beginning of this restricted interval.
4. put inappropriate chemotherapy drugs in the treatment plan is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if put inappropriate chemotherapy drugs in the treatment plan does occur, Clinic RN approves treatment plan is not required to occur subsequently and if it does not occur subsequently, then the behavior is not required to hold for the remainder of the event sequence.
5. There might be many restricted intervals in the event sequence. If put inappropriate chemotherapy drugs in the treatment plan occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of put inappropriate chemotherapy drugs in the treatment plan, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a potential new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

BEHAVIOR:
1. The event of primary interest in this behavior is put only appropriate chemotherapy drugs in the treatment plan.
2. The events of secondary interest in this behavior are Pharmacy approves treatment plan and Practice RN OR second Clinic RN approves treatment plan.
3. put only appropriate chemotherapy drugs in the treatment plan is required to occur at least once.
Event alphabet:
• A: put inappropriate chemotherapy drug doses in treatment plan
• B: put only appropriate chemotherapy drug doses in treatment plan
• START: Fellow MD creates treatment plan and/or chemotherapy orders
• END: Attending MD approves treatment plan

SCOPE:
1. A restricted interval in the event sequence can have both a starting delimiter, Fellow MD creates treatment plan and/or chemotherapy orders, and an ending delimiter, Attending MD approves treatment plan.
2. The behavior is required to hold from an occurrence of Fellow MD creates treatment plan and/or chemotherapy orders, if it ever occurs, through to the first subsequent occurrence of Attending MD approves treatment plan, if it ever occurs.
3. If there are multiple occurrences of Fellow MD creates treatment plan and/or chemotherapy orders without an occurrence of Attending MD approves treatment plan in between them, only the last of those occurrences of Fellow MD creates treatment plan and/or chemotherapy orders potentially starts the restricted interval; each of those occurrences of Fellow MD creates treatment plan and/or chemotherapy orders resets the beginning of the restricted interval.
4. Fellow MD creates treatment plan and/or chemotherapy orders is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if Fellow MD creates treatment plan and/or chemotherapy orders does occur, Attending MD approves treatment plan is not required to occur subsequently and if it does not occur subsequently, then the behavior is not required to hold for the remainder of the event sequence.
5. There can be at most one restricted interval in the event sequence.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

BEHAVIOR:
1. The events of primary interest in this behavior are put inappropriate chemotherapy drug doses in treatment plan and put only appropriate chemotherapy drug doses in treatment plan.
2. There are no events of secondary interest in this behavior.
3. If put inappropriate chemotherapy drug doses in treatment plan occurs, put only appropriate chemotherapy drug doses in treatment plan is required to occur subsequently.
4. Before the first put inappropriate chemotherapy drug doses in treatment plan occurs, put only appropriate chemotherapy drug doses in treatment plan is allowed to occur zero or more times.
5. put inappropriate chemotherapy drug doses in treatment plan is not required to occur.
6. After put inappropriate chemotherapy drug doses in treatment plan occurs, but before the first subsequent put only appropriate chemotherapy drug doses in treatment plan occurs, put inappropriate chemotherapy drug doses in treatment plan is allowed to occur again, zero or more times.
7. After put inappropriate chemotherapy drug doses in treatment plan and the first subsequent put only appropriate chemotherapy drug doses in treatment plan occur:
   • put only appropriate chemotherapy drug doses in treatment plan is allowed to occur again, zero or more times, before another put inappropriate chemotherapy drug doses in treatment plan occurs; put inappropriate chemotherapy drug doses in treatment plan is allowed to occur again and, if it does, then the situation is the same as when the first put inappropriate chemotherapy drug doses in treatment plan occurred, meaning that the restrictions described in parts 3, 6, and 7 would again apply.

FSA and DNL for Property Chemo.C.8a
SCOPE:
1. A restricted interval in the event sequence can have both a starting delimiter, put inappropriate chemotherapy drug doses in treatment plan, and an ending delimiter, Pharmacy approves treatment plan.
2. The behavior is required to hold from an occurrence of put inappropriate chemotherapy drug doses in treatment plan, if it ever occurs, through to the first subsequent occurrence of Pharmacy approves treatment plan, if it ever occurs.
3. If there are multiple occurrences of put inappropriate chemotherapy drug doses in treatment plan without an occurrence of Pharmacy approves treatment plan in between them, only the last of those occurrences of put inappropriate chemotherapy drug doses in treatment plan potentially starts a restricted interval; each of those occurrences of put inappropriate chemotherapy drug doses in treatment plan resets the beginning of this restricted interval.
4. put inappropriate chemotherapy drug doses in treatment plan is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if put inappropriate chemotherapy drug doses in treatment plan occurs Pharmacy approves treatment plan is not required to occur subsequently and if it does not occur subsequently, then the behavior is not required to hold for the remainder of the event sequence.
5. There might be many restricted intervals in the event sequence. If put inappropriate chemotherapy drug doses in treatment plan occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of put inappropriate chemotherapy drug doses in treatment plan, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a potential new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

BEHAVIOR:
1. The event of primary interest in this behavior is put only appropriate chemotherapy drugs doses in treatment plan.
2. The events of secondary interest in this behavior are Clinic RN approves treatment plan and Practice RN OR second Clinic RN approves treatment plan.
3. put only appropriate chemotherapy drugs doses in treatment plan is required to occur at least once.
Event alphabet:

- **A**: put only appropriate chemotherapy drugs doses in treatment plan
- **START**: put inappropriate chemotherapy drug doses in treatment plan
- **END**: Clinic RN approves treatment plan
- **C**: Pharmacy approves treatment plan, Practice RN OR second Clinic RN approves treatment plan

**SCOPE:**

1. A restricted interval in the event sequence can have both a starting delimiter, put inappropriate chemotherapy drug doses in treatment plan, and an ending delimiter, Clinic RN approves treatment plan.
2. The behavior is required to hold from an occurrence of put inappropriate chemotherapy drug doses in treatment plan, if it ever occurs, through to the first subsequent occurrence of Clinic RN approves treatment plan, if it ever occurs.
3. If there are multiple occurrences of put inappropriate chemotherapy drug doses in treatment plan in between them, only the last of those occurrences of put inappropriate chemotherapy drug doses in treatment plan potentially starts a restricted interval; each of those occurrences of put inappropriate chemotherapy drug doses in treatment plan resets the beginning of this restricted interval.
4. put inappropriate chemotherapy drug doses in treatment plan is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if put inappropriate chemotherapy drug doses in treatment plan does occur, Clinic RN approves treatment plan is not required to occur subsequently and if it does not occur subsequently, then the behavior is not required to hold for the remainder of the event sequence.
5. There might be many restricted intervals in the event sequence. If put inappropriate chemotherapy drug doses in treatment plan occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of put inappropriate chemotherapy drug doses in treatment plan, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a potential new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

**BEHAVIOR:**

1. The event of primary interest in this behavior is put only appropriate chemotherapy drugs doses in treatment plan.
2. The events of secondary interest in this behavior are Pharmacy approves treatment plan and Practice RN OR second Clinic RN approves treatment plan.
3. put only appropriate chemotherapy drugs doses in treatment plan is required to occur at least once.
Event alphabet:
- A: put inappropriate chemotherapy drug doses in treatment plan
- B: put only appropriate chemotherapy drugs doses in treatment plan
- END: Practice RN OR second Clinic RN approves treatment plan
- C: Clinic RN approves treatment plan, Pharmacy approves treatment plan

SCOPE:
1. There can be at most one restricted interval in the event sequence and it has an ending delimiter, Practice RN OR second Clinic RN approves treatment plan.
2. The behavior is required to hold from the start of the event sequence through to the first occurrence of Practice RN OR second Clinic RN approves treatment plan, if it ever occurs.
3. Practice RN OR second Clinic RN approves treatment plan is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence.
4. There are no restrictions imposed on the occurrences of any events after the first occurrence of Practice RN OR second Clinic RN approves treatment plan, if it ever occurs.

BEHAVIOR:
1. The events of primary interest in this behavior are put inappropriate chemotherapy drug doses in treatment plan and put only appropriate chemotherapy drugs doses in treatment plan.
2. The events of secondary interest in this behavior are Clinic RN approves treatment plan and Pharmacy approves treatment plan.
3. If put inappropriate chemotherapy drug doses in treatment plan occurs, put only appropriate chemotherapy drugs doses in treatment plan is required to occur subsequently.
4. Before the first put inappropriate chemotherapy drug doses in treatment plan occurs:
   - put only appropriate chemotherapy drugs doses in treatment plan is allowed to occur zero or more times;
   - All the events of secondary interest are allowed to occur zero or more times.
5. put inappropriate chemotherapy drug doses in treatment plan is not required to occur.
6. After put inappropriate chemotherapy drug doses in treatment plan occurs, but before the first subsequent put only appropriate chemotherapy drugs doses in treatment plan occurs:
   - put inappropriate chemotherapy drug doses in treatment plan is allowed to occur again, zero or more times;
   - None of the events of secondary interest are allowed to occur.
7. After put inappropriate chemotherapy drug doses in treatment plan and the first subsequent put only appropriate chemotherapy drugs doses in treatment plan occur:
   - All the events of secondary interest are allowed to occur zero or more times;
   - put only appropriate chemotherapy drugs doses in treatment plan is allowed to occur again, zero or more times, before another put inappropriate chemotherapy drug doses in treatment plan occurs; put inappropriate chemotherapy drug doses in treatment plan is allowed to occur again and, if it does, then the situation is the same as when the first put inappropriate chemotherapy drug doses in treatment plan occurred, meaning that the restrictions described in parts 3, 6, and 7 would again apply.

FSA and DNL for Property Chemo.C.8d
Event alphabet:
- A: Patient data in treatment plan becomes stale or disparate
- B: resolve stale or disparate Patient data in treatment plan
- END: Practice RN OR second Clinic RN approves treatment plan
- C: Clinic RN approves treatment plan, Pharmacy approves treatment plan

**SCOPE:**
1. There can be at most one restricted interval in the event sequence and it has an ending delimiter, **Practice RN OR second Clinic RN approves treatment plan**.
2. The behavior is required to hold from the start of the event sequence through to the first occurrence of **Practice RN OR second Clinic RN approves treatment plan**, if it ever occurs.
3. **Practice RN OR second Clinic RN approves treatment plan** is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence.
4. There are no restrictions imposed on the occurrences of any events after the first occurrence of **Practice RN OR second Clinic RN approves treatment plan**, if it ever occurs.

**BEHAVIOR:**
1. The events of primary interest in this behavior are **Patient data in treatment plan becomes stale or disparate** and **resolve stale or disparate** **Patient data in treatment plan**.
2. The events of secondary interest in this behavior are **Clinic RN approves treatment plan** and **Pharmacy approves treatment plan**.
3. If **Patient data in treatment plan becomes stale or disparate** occurs, **resolve stale or disparate** **Patient data in treatment plan** is required to occur subsequently.
4. Before the first **Patient data in treatment plan becomes stale or disparate** occurs:
   - **resolve stale or disparate** **Patient data in treatment plan** is not allowed to occur;
   - All the events of secondary interest are allowed to occur zero or more times.
5. **Patient data in treatment plan becomes stale or disparate** is not required to occur.
6. After **Patient data in treatment plan becomes stale or disparate** occurs, but before the first subsequent **resolve stale or disparate** **Patient data in treatment plan** occurs:
   - **Patient data in treatment plan becomes stale or disparate** is allowed to occur again, zero or more times;
   - None of the events of secondary interest are allowed to occur.
7. After **Patient data in treatment plan becomes stale or disparate** and the first subsequent **resolve stale or disparate** **Patient data in treatment plan** occur:
   - All the events of secondary interest are allowed to occur zero or more times;
   - **resolve stale or disparate** **Patient data in treatment plan** is not allowed to occur again until after another **Patient data in treatment plan becomes stale or disparate** occurs. **Patient data in treatment plan becomes stale or disparate** is allowed to occur again and, if it does, then the situation is the same as when the first **Patient data in treatment plan becomes stale or disparate** occurred, meaning that the restrictions described in parts 3, 6, and 7 would again apply.

**FSA and DNL for Property Chemo.C.9a**
### Event alphabet:
- **A**: resolve stale or disparate Patient data in treatment plan
- **START**: Patient data in treatment plan becomes stale or disparate
- **END**: Pharmacy approves treatment plan
- **C**: Clinic RN approves treatment plan, Practice RN OR second Clinic RN approves treatment plan

### SCOPE:
1. A restricted interval in the event sequence can have both a starting delimiter, **Patient data in treatment plan becomes stale or disparate**, and an ending delimiter, **Pharmacy approves treatment plan**.
2. The behavior is required to hold from an occurrence of **Patient data in treatment plan becomes stale or disparate**, if it ever occurs, through to the first subsequent occurrence of **Pharmacy approves treatment plan**, if it ever occurs.
3. If there are multiple occurrences of **Patient data in treatment plan becomes stale or disparate** without an occurrence of **Pharmacy approves treatment plan** in between them, only the last of those occurrences of **Patient data in treatment plan becomes stale or disparate** potentially starts a restricted interval; each of those occurrences of **Patient data in treatment plan becomes stale or disparate** resets the beginning of this restricted interval.
4. **Patient data in treatment plan becomes stale or disparate** is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if **Patient data in treatment plan becomes stale or disparate** does occur, **Pharmacy approves treatment plan** is not required to occur subsequently and if it does not occur subsequently, then the behavior is not required to hold for the remainder of the event sequence.
5. There might be many restricted intervals in the event sequence. If **Patient data in treatment plan becomes stale or disparate** occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of **Patient data in treatment plan becomes stale or disparate**, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a potential new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

### BEHAVIOR:
1. The event of primary interest in this behavior is **resolve stale or disparate Patient data in treatment plan**.
2. The events of secondary interest in this behavior are **Clinic RN approves treatment plan** and **Practice RN OR second Clinic RN approves treatment plan**.
3. **resolve stale or disparate Patient data in treatment plan** is required to occur at least once.

---

**FSA and DNL for Property Chemo.C.9b**
**Event alphabet:**
- **A**: resolve stale or disparate Patient data in treatment plan
- **START**: Patient data in treatment plan becomes stale or disparate
- **END**: Clinic RN approves treatment plan
- **C**: Pharmacy approves treatment plan, Practice RN OR second Clinic RN approves treatment plan

**SCOPE:**
1. A restricted interval in the event sequence can have both a starting delimiter, Patient data in treatment plan becomes stale or disparate, and an ending delimiter, Clinic RN approves treatment plan.
2. The behavior is required to hold from an occurrence of Patient data in treatment plan becomes stale or disparate, if it ever occurs, through to the first subsequent occurrence of Clinic RN approves treatment plan, if it ever occurs.
3. If there are multiple occurrences of Patient data in treatment plan becomes stale or disparate without an occurrence of Clinic RN approves treatment plan in between them, only the last of those occurrences of Patient data in treatment plan becomes stale or disparate potentially starts a restricted interval; each of those occurrences of Patient data in treatment plan becomes stale or disparate resets the beginning of this restricted interval.
4. Patient data in treatment plan becomes stale or disparate is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if Patient data in treatment plan becomes stale or disparate does occur, Clinic RN approves treatment plan is not required to occur subsequently and if it does not occur subsequently, then the behavior is not required to hold for the remainder of the event sequence.
5. There might be many restricted intervals in the event sequence. If Patient data in treatment plan becomes stale or disparate occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of Patient data in treatment plan becomes stale or disparate, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a potential new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

**BEHAVIOR:**
1. The event of primary interest in this behavior is resolve stale or disparate Patient data in treatment plan.
2. The events of secondary interest in this behavior are Pharmacy approves treatment plan and Practice RN OR second Clinic RN approves treatment plan.
3. resolve stale or disparate Patient data in treatment plan is required to occur at least once.

FSA and DNL for Property Chemo.C.9c
Event alphabet:

- A: resolve problematic Patient data in the treatment plan
- START: Patient data in treatment plan becomes problematic
- END: create treatment plan

**SCOPE:**
1. A restricted interval in the event sequence can have both a starting delimiter, Patient data in treatment plan becomes problematic, and an ending delimiter, create treatment plan.
2. The behavior is required to hold from an occurrence of Patient data in treatment plan becomes problematic, if it ever occurs, through to the first subsequent occurrence of create treatment plan, if it ever occurs.
3. If there are multiple occurrences of Patient data in treatment plan becomes problematic without an occurrence of create treatment plan in between them, only the last of those occurrences of Patient data in treatment plan becomes problematic potentially starts the restricted interval; each of those occurrences of Patient data in treatment plan becomes problematic resets the beginning of the restricted interval.
4. Patient data in treatment plan becomes problematic is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if Patient data in treatment plan becomes problematic resets the beginning of the restricted interval, Patient data in treatment plan becomes problematic is not required to occur subsequently and if it does not occur subsequently, then the behavior is not required to hold for the remainder of the event sequence.
5. There can be at most one restricted interval in the event sequence.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

**BEHAVIOR:**
1. The event of primary interest in this behavior is resolve problematic Patient data in the treatment plan.
2. There are no events of secondary interest in this behavior.
3. resolve problematic Patient data in the treatment plan is required to occur at least once.

FSA and DNL for Property Chemo.C.10
**Event alphabet:**
- A: Attending MD approves treatment plan
- **START:** Fellow MD creates treatment plan
- **END:** administer chemotherapy

**SCOPE:**
1. A restricted interval in the event sequence can have both a starting delimiter, Fellow MD creates treatment plan, and an ending delimiter, administer chemotherapy.
2. The behavior is required to hold from an occurrence of Fellow MD creates treatment plan, if it ever occurs, through to the first subsequent occurrence of administer chemotherapy, if it ever occurs.
3. If there are multiple occurrences of Fellow MD creates treatment plan without an occurrence of administer chemotherapy in between them, only the last of those occurrences of Fellow MD creates treatment plan potentially starts the restricted interval; each of those occurrences of Fellow MD creates treatment plan resets the beginning of the restricted interval.
4. Fellow MD creates treatment plan is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if Fellow MD creates treatment plan does occur, administer chemotherapy is not required to occur subsequently and if it does not occur subsequently, then the behavior is not required to hold for the remainder of the event sequence.
5. There can be at most one restricted interval in the event sequence.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

**BEHAVIOR:**
1. The event of primary interest in this behavior is Attending MD approves treatment plan.
2. There are no events of secondary interest in this behavior.
3. Attending MD approves treatment plan is required to occur at least once.

**FSA and DNL for Property Chemo.C.1a**
**Event alphabet:**
- A: Practice RN OR second Clinic RN approves treatment plan
- B: administer chemotherapy
- C: Attending MD approves treatment plan, Clinic RN approves treatment plan, Fellow MD creates treatment plan, Pharmacy approves treatment plan

**SCOPE:**
1. From the start of any event sequence through to the end of that event sequence, the behavior must hold.

**BEHAVIOR:**
1. The events of primary interest in this behavior are Practice RN OR second Clinic RN approves treatment plan and administer chemotherapy.
2. The events of secondary interest in this behavior are Attending MD approves treatment plan, Clinic RN approves treatment plan, Fellow MD creates treatment plan, and Pharmacy approves treatment plan.
3. administer chemotherapy is not allowed to occur until after Practice RN OR second Clinic RN approves treatment plan occurs.
4. Before the first Practice RN OR second Clinic RN approves treatment plan occurs, all the events of secondary interest are allowed to occur zero or more times.
5. Practice RN OR second Clinic RN approves treatment plan is not required to occur.
6. Even if Practice RN OR second Clinic RN approves treatment plan does occur, administer chemotherapy is not required to occur after Practice RN OR second Clinic RN approves treatment plan occurs.
7. After Practice RN OR second Clinic RN approves treatment plan occurs, but before the first subsequent administer chemotherapy occurs:
   - Practice RN OR second Clinic RN approves treatment plan is allowed to occur again, zero or more times;
   - All the events of secondary interest are allowed to occur zero or more times.
8. After Practice RN OR second Clinic RN approves treatment plan and the first subsequent administer chemotherapy occur:
   - Both Practice RN OR second Clinic RN approves treatment plan and administer chemotherapy are allowed to occur again, zero or more times, and there are no restrictions on the occurrences of any future events.

FSA and DNL for Property Chemo.C.1b
**Event alphabet:**
- A: Pharmacy approves treatment plan
- B: administer chemotherapy
- C: Attending MD approves treatment plan, Clinic RN approves treatment plan, Fellow MD creates treatment plan, Practice RN OR second Clinic RN approves treatment plan

**SCOPE:**
1. From the start of any event sequence through to the end of that event sequence, the behavior must hold.

**BEHAVIOR:**
1. The events of primary interest in this behavior are *Pharmacy approves treatment plan* and *administer chemotherapy*.
2. The events of secondary interest in this behavior are Attending MD approves treatment plan, Clinic RN approves treatment plan, Fellow MD creates treatment plan, and Practice RN OR second Clinic RN approves treatment plan.
3. *administer chemotherapy* is not allowed to occur until after *Pharmacy approves treatment plan* occurs.
4. Before the first *Pharmacy approves treatment plan* occurs, all the events of secondary interest are allowed to occur zero or more times.
5. *Pharmacy approves treatment plan* is not required to occur.
6. Even if *Pharmacy approves treatment plan* does occur, *administer chemotherapy* is not required to occur after *Pharmacy approves treatment plan* occurs.
7. After *Pharmacy approves treatment plan* occurs, but before the first subsequent *administer chemotherapy* occurs:
   - *Pharmacy approves treatment plan* is allowed to occur again, zero or more times;
   - All the events of secondary interest are allowed to occur zero or more times.
8. After *Pharmacy approves treatment plan* and the first subsequent *administer chemotherapy* occur:
   - *All the events of secondary interest are allowed to occur zero or more times*;
   - Both *Pharmacy approves treatment plan* and *administer chemotherapy* are allowed to occur again, zero or more times, and there are no restrictions on the occurrences of any future events.

**FSA and DNL for Property Chemo.C.1c**
Event alphabet:
- A: Clinic RN approves treatment plan
- B: administer chemotherapy
- C: Attending MD approves treatment plan, Fellow MD creates treatment plan, Pharmacy approves treatment plan, Practice RN OR second Clinic RN approves treatment plan

**SCOPE:**
1. From the start of any event sequence through to the end of that event sequence, the behavior must hold.

**BEHAVIOR:**
1. The events of primary interest in this behavior are Clinic RN approves treatment plan and administer chemotherapy.
2. The events of secondary interest in this behavior are Attending MD approves treatment plan, Fellow MD creates treatment plan, Pharmacy approves treatment plan, and Practice RN OR second Clinic RN approves treatment plan.
3. administer chemotherapy is not allowed to occur until after Clinic RN approves treatment plan occurs.
4. Before the first Clinic RN approves treatment plan occurs, all the events of secondary interest are allowed to occur zero or more times.
5. Clinic RN approves treatment plan is not required to occur.
6. Even if Clinic RN approves treatment plan does occur, administer chemotherapy is not required to occur after Clinic RN approves treatment plan occurs.
7. after Clinic RN approves treatment plan occurs, but before the first subsequent administer chemotherapy occurs:
   - Clinic RN approves treatment plan is allowed to occur again, zero or more times;
   - All the events of secondary interest are allowed to occur zero or more times.
8. After Clinic RN approves treatment plan and the first subsequent administer chemotherapy occur:
   - Both Clinic RN approves treatment plan and administer chemotherapy are allowed to occur again, zero or more times, and there are no restrictions on the occurrences of any future events.

FSA and DNL for Property Chemo.C.1d
Event alphabet:
- A: Attending MD approves treatment plan
- START: Fellow MD makes a critical change to the treatment plan or chemotherapy orders
- END: administer chemotherapy

**SCOPE:**
1. A restricted interval in the event sequence can have both a starting delimiter, Fellow MD makes a critical change to the treatment plan or chemotherapy orders, and an ending delimiter, administer chemotherapy.
2. The behavior is required to hold from an occurrence of Fellow MD makes a critical change to the treatment plan or chemotherapy orders, if it ever occurs, through to the first subsequent occurrence of administer chemotherapy, if it ever occurs.
3. If there are multiple occurrences of Fellow MD makes a critical change to the treatment plan or chemotherapy orders without an occurrence of administer chemotherapy in between them, only the last of those occurrences of Fellow MD makes a critical change to the treatment plan or chemotherapy orders potentially starts the restricted interval; each of those occurrences of Fellow MD makes a critical change to the treatment plan or chemotherapy orders resets the beginning of the restricted interval.
4. Fellow MD makes a critical change to the treatment plan or chemotherapy orders is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if Fellow MD makes a critical change to the treatment plan or chemotherapy orders does occur, administer chemotherapy is not required to occur subsequently and if it does not occur subsequently, then the behavior is not required to hold for the remainder of the event sequence.
5. There can be at most one restricted interval in the event sequence.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

**BEHAVIOR:**
1. The event of primary interest in this behavior is Attending MD approves treatment plan.
2. There are no events of secondary interest in this behavior.
3. Attending MD approves treatment plan is required to occur at least once.

FSA and DNL for Property Chemo.C.2a
Event alphabet:

- A: Pharmacy approves treatment plan
- START: make a critical change to the treatment plan
- END: administer chemotherapy
- C: Attending MD approves treatment plan, Clinic RN approves treatment plan

**SCOPE:**

1. A restricted interval in the event sequence can have both a starting delimiter, make a critical change to the treatment plan, and an ending delimiter, administer chemotherapy.
2. The behavior is required to hold from an occurrence of make a critical change to the treatment plan, if it ever occurs, through to the first subsequent occurrence of administer chemotherapy, if it ever occurs.
3. If there are multiple occurrences of make a critical change to the treatment plan without an occurrence of administer chemotherapy in between them, only the last of those occurrences of make a critical change to the treatment plan potentially starts the restricted interval; each of those occurrences of make a critical change to the treatment plan resets the beginning of the restricted interval.
4. make a critical change to the treatment plan is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if make a critical change to the treatment plan does occur, administer chemotherapy is not required to occur subsequently and if it does not occur subsequently, then the behavior is not required to hold for the remainder of the event sequence.
5. There can be at most one restricted interval in the event sequence.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

**BEHAVIOR:**

1. The event of primary interest in this behavior is Pharmacy approves treatment plan.
2. The events of secondary interest in this behavior are Attending MD approves treatment plan and Clinic RN approves treatment plan.
3. Pharmacy approves treatment plan is required to occur at least once.

FSA and DNL for Property Chemo.C.2b
Event alphabet:
- A: Clinic RN approves treatment plan
- START, make a critical change to the treatment plan
- END, administer chemotherapy
- C: Attending MD approves treatment plan, Pharmacy approves treatment plan

**SCOPE:**
1. A restricted interval in the event sequence can have both a starting delimiter, make a critical change to the treatment plan, and an ending delimiter, administer chemotherapy.
2. The behavior is required to hold from an occurrence of make a critical change to the treatment plan, if it ever occurs, through to the first subsequent occurrence of administer chemotherapy, if it ever occurs.
3. If there are multiple occurrences of make a critical change to the treatment plan without an occurrence of administer chemotherapy in between them, only the last of those occurrences of make a critical change to the treatment plan potentially starts the restricted interval; each of those occurrences of make a critical change to the treatment plan resets the beginning of the restricted interval.
4. make a critical change to the treatment plan is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if make a critical change to the treatment plan does occur, administer chemotherapy is not required to occur subsequently and if it does not occur subsequently, then the behavior is not required to hold for the remainder of the event sequence.
5. There can be at most one restricted interval in the event sequence.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

**BEHAVIOR:**
1. The event of primary interest in this behavior is Clinic RN approves treatment plan.
2. The events of secondary interest in this behavior are Attending MD approves treatment plan and Pharmacy approves treatment plan.
3. Clinic RN approves treatment plan is required to occur at least once.

FSA and DNL for Property Chemo.C.2c
Event alphabet:
- A: Practice RN OR second Clinic RN verifies chemotherapy orders
- B: administer chemotherapy
- C: Clinic RN verifies chemotherapy orders, Pharmacy verifies chemotherapy orders

<table>
<thead>
<tr>
<th>A</th>
<th>C</th>
<th>A</th>
<th>C</th>
</tr>
</thead>
</table>

**SCOPE:**
From the start of any event sequence through to the end of that event sequence, the behavior must hold.

**BEHAVIOR:**
1. The events of primary interest in this behavior are Practice RN OR second Clinic RN verifies chemotherapy orders and administer chemotherapy.
2. The events of secondary interest in this behavior are Clinic RN verifies chemotherapy orders and Pharmacy verifies chemotherapy orders.
3. administer chemotherapy is not allowed to occur until after Practice RN OR second Clinic RN verifies chemotherapy orders occurs.
4. Before the first Practice RN OR second Clinic RN verifies chemotherapy orders occurs, all the events of secondary interest are allowed to occur zero or more times.
5. Practice RN OR second Clinic RN verifies chemotherapy orders is not required to occur.
6. Even if Practice RN OR second Clinic RN verifies chemotherapy orders does occur, administer chemotherapy is not required to occur after Practice RN OR second Clinic RN verifies chemotherapy orders occurs.
7. After Practice RN OR second Clinic RN verifies chemotherapy orders occurs, but before the first subsequent administer chemotherapy occurs:
   - Practice RN OR second Clinic RN verifies chemotherapy orders is allowed to occur again, zero or more times;
   - All the events of secondary interest are allowed to occur zero or more times.
8. After Practice RN OR second Clinic RN verifies chemotherapy orders and the first subsequent administer chemotherapy occur:
   - All the events of secondary interest are allowed to occur zero or more times;
   - Both Practice RN OR second Clinic RN verifies chemotherapy orders and administer chemotherapy are allowed to occur again, zero or more times, and there are no restrictions on the occurrences of any future events.

FSA and DNL for Property Chemo.C.3a
Event alphabet:
- A: Pharmacy verifies chemotherapy orders
- B: administer chemotherapy
- C: Clinic RN verifies chemotherapy orders, Practice RN OR second Clinic RN verifies chemotherapy orders

SCOPE:
1. From the start of any event sequence through to the end of that event sequence, the behavior must hold.

BEHAVIOR:
1. The events of primary interest in this behavior are Pharmacy verifies chemotherapy orders and administer chemotherapy.
2. The events of secondary interest in this behavior are Clinic RN verifies chemotherapy orders and Practice RN OR second Clinic RN verifies chemotherapy orders.
3. administer chemotherapy is not allowed to occur until after Pharmacy verifies chemotherapy orders occurs.
4. Before the first Pharmacy verifies chemotherapy orders occurs, all the events of secondary interest are allowed to occur zero or more times.
5. Pharmacy verifies chemotherapy orders is not required to occur.
6. Even if Pharmacy verifies chemotherapy orders does occur, administer chemotherapy is not required to occur after Pharmacy verifies chemotherapy orders occurs.
7. After Pharmacy verifies chemotherapy orders occurs, but before the first subsequent administer chemotherapy occurs:
   - Pharmacy verifies chemotherapy orders is allowed to occur again, zero or more times;
   - All the events of secondary interest are allowed to occur zero or more times.
8. After Pharmacy verifies chemotherapy orders and the first subsequent administer chemotherapy occur:
   - All the events of secondary interest are allowed to occur zero or more times;
   - administer chemotherapy is not allowed to occur again until after another Pharmacy verifies chemotherapy orders occurs; Pharmacy verifies chemotherapy orders is allowed to occur again and, if it does, then the situation is the same as when the first Pharmacy verifies chemotherapy orders occurred, meaning that the restrictions described in parts 6, 7, and 8 would again apply.

FSA and DNL for Property Chemo.C.3b
**Event alphabet:**
- A: Clinic RN verifies chemotherapy orders
- B: administer chemotherapy
- C: Pharmacy verifies chemotherapy orders, Practice RN OR second Clinic RN verifies chemotherapy orders

**SCOPE:**
1. From the start of any event sequence through to the end of that event sequence, the behavior must hold.

**BEHAVIOR:**
1. The events of primary interest in this behavior are Clinic RN verifies chemotherapy orders and administer chemotherapy.
2. The events of secondary interest in this behavior are Pharmacy verifies chemotherapy orders and Practice RN OR second Clinic RN verifies chemotherapy orders.
3. administer chemotherapy is not allowed to occur until after Clinic RN verifies chemotherapy orders occurs.
4. Before the first Clinic RN verifies chemotherapy orders occurs, all the events of secondary interest are allowed to occur zero or more times.
5. Clinic RN verifies chemotherapy orders is not required to occur.
6. Even if Clinic RN verifies chemotherapy orders does occur, administer chemotherapy is not required to occur after Clinic RN verifies chemotherapy orders occurs.
7. After Clinic RN verifies chemotherapy orders occurs, but before the first subsequent administer chemotherapy occurs:
   - Clinic RN verifies chemotherapy orders is allowed to occur again, zero or more times;
   - All the events of secondary interest are allowed to occur zero or more times.
8. After Clinic RN verifies chemotherapy orders and the first subsequent administer chemotherapy occur:
   - All the events of secondary interest are allowed to occur zero or more times;
   - administer chemotherapy is not allowed to occur again until after another Clinic RN verifies chemotherapy orders occurs; Clinic RN verifies chemotherapy orders is allowed to occur again and, if it does, then the situation is the same as when the first Clinic RN verifies chemotherapy orders occurred, meaning that the restrictions described in parts 6, 7, and 8 would again apply.

FSA and DNL for Property Chemo.C.3c
Event alphabet:
- A: Attending MD OR Fellow MD approves critical change to chemotherapy orders
- START: make critical change to chemotherapy orders
- END: Pharmacy verifies chemotherapy orders
- C: Clinic RN verifies chemotherapy orders

**SCOPE:**
1. A restricted interval in the event sequence can have both a starting delimiter, make critical change to chemotherapy orders, and an ending delimiter, Pharmacy verifies chemotherapy orders.
2. The behavior is required to hold from an occurrence of make critical change to chemotherapy orders, if it ever occurs, through to the first subsequent occurrence of Pharmacy verifies chemotherapy orders, if it ever occurs.
3. If there are multiple occurrences of make critical change to chemotherapy orders without an occurrence of Pharmacy verifies chemotherapy orders in between them, only the last of those occurrences of make critical change to chemotherapy orders potentially starts a restricted interval; each of those occurrences of make critical change to chemotherapy orders resets the beginning of this restricted interval.
4. make critical change to chemotherapy orders is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if make critical change to chemotherapy orders does occur, Pharmacy verifies chemotherapy orders is not required to occur subsequently and if it does not occur subsequently, then the behavior is not required to hold for the remainder of the event sequence.
5. There might be many restricted intervals in the event sequence. If make critical change to chemotherapy orders occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of make critical change to chemotherapy orders, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a potential new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

**BEHAVIOR:**
1. The event of primary interest in this behavior is Attending MD OR Fellow MD approves critical change to chemotherapy orders.
2. The event of secondary interest in this behavior is Clinic RN verifies chemotherapy orders.
3. Attending MD OR Fellow MD approves critical change to chemotherapy orders is required to occur at least once.

FSA and DNL for Property Chemo.C.4a
**Event alphabet:**
- \( A \): Attending MD OR Fellow MD approves critical change to chemotherapy orders
- \( START \): make critical change to chemotherapy orders
- \( END \): Clinic RN verifies chemotherapy orders
- \( C \): Pharmacy verifies chemotherapy orders

**SCOPE:**
1. A restricted interval in the event sequence can have both a starting delimiter, \textit{make critical change to chemotherapy orders}, and an ending delimiter, \textit{Clinic RN verifies chemotherapy orders}.
2. The behavior is required to hold from an occurrence of \textit{make critical change to chemotherapy orders}, if it ever occurs, through to the first subsequent occurrence of \textit{Clinic RN verifies chemotherapy orders}, if it ever occurs.
3. If there are multiple occurrences of \textit{make critical change to chemotherapy orders} without an occurrence of \textit{Clinic RN verifies chemotherapy orders} in between them, only the last of those occurrences of \textit{make critical change to chemotherapy orders} potentially starts a restricted interval; each of those occurrences of \textit{make critical change to chemotherapy orders} resets the beginning of this restricted interval.
4. \textit{make critical change to chemotherapy orders} is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if \textit{make critical change to chemotherapy orders} does occur, \textit{Clinic RN verifies chemotherapy orders} is not required to occur subsequently and if it does not occur subsequently, then the behavior is not required to hold for the remainder of the event sequence.
5. There might be many restricted intervals in the event sequence. If \textit{make critical change to chemotherapy orders} occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of \textit{make critical change to chemotherapy orders}, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a potential new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

**BEHAVIOR:**
1. The event of primary interest in this behavior is \textit{Attending MD OR Fellow MD approves critical change to chemotherapy orders}.
2. The event of secondary interest in this behavior is \textit{Pharmacy verifies chemotherapy orders}.
3. \textit{Attending MD OR Fellow MD approves critical change to chemotherapy orders} is required to occur at least once.

**FSA and DNL for Property Chemo.C.4b**
Event alphabet:
- A: Pharmacy verifies chemotherapy orders
- START: make critical change to chemotherapy orders
- END: administer chemotherapy
- C: Clinic RN verifies chemotherapy orders

**SCOPE:**
1. A restricted interval in the event sequence can have both a starting delimiter, make critical change to chemotherapy orders, and an ending delimiter, administer chemotherapy.
2. The behavior is required to hold from an occurrence of make critical change to chemotherapy orders, if it ever occurs, through to the first subsequent occurrence of administer chemotherapy, if it ever occurs.
3. If there are multiple occurrences of make critical change to chemotherapy orders without an occurrence of administer chemotherapy in between them, only the last of those occurrences of make critical change to chemotherapy orders potentially starts a restricted interval; each of those occurrences of make critical change to chemotherapy orders resets the beginning of this restricted interval.
4. make critical change to chemotherapy orders is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if make critical change to chemotherapy orders does occur, administer chemotherapy is not required to occur subsequently and if it does not occur subsequently, then the behavior is not required to hold for the remainder of the event sequence.
5. There might be many restricted intervals in the event sequence. If make critical change to chemotherapy orders occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of make critical change to chemotherapy orders, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a potential new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

**BEHAVIOR:**
1. The event of primary interest in this behavior is Pharmacy verifies chemotherapy orders.
2. The event of secondary interest in this behavior is Clinic RN verifies chemotherapy orders.
3. Pharmacy verifies chemotherapy orders is required to occur at least once.

FSA and DNL for Property Chemo.C.5a
Event alphabet:

- A: Clinic RN verifies chemotherapy orders
- START: make critical change to chemotherapy orders
- END: administer chemotherapy
- C: Pharmacy verifies chemotherapy orders

SCOPE:
1. A restricted interval in the event sequence can have both a starting delimiter, make critical change to chemotherapy orders, and an ending delimiter, administer chemotherapy.
2. The behavior is required to hold from an occurrence of make critical change to chemotherapy orders, if it ever occurs, through to the first subsequent occurrence of administer chemotherapy, if it ever occurs.
3. If there are multiple occurrences of make critical change to chemotherapy orders without an occurrence of administer chemotherapy in between them, only the last of those occurrences of make critical change to chemotherapy orders potentially starts a restricted interval; each of those occurrences of make critical change to chemotherapy orders resets the beginning of this restricted interval.
4. make critical change to chemotherapy orders is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if make critical change to chemotherapy orders does occur, administer chemotherapy is not required to occur subsequently and if it does not occur subsequently, then the behavior is not required to hold for the remainder of the event sequence.
5. There might be many restricted intervals in the event sequence. If make critical change to chemotherapy orders occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of make critical change to chemotherapy orders, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a potential new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

BEHAVIOR:
1. The event of primary interest in this behavior is Clinic RN verifies chemotherapy orders.
2. The event of secondary interest in this behavior is Pharmacy verifies chemotherapy orders.
3. Clinic RN verifies chemotherapy orders is required to occur at least once.

FSA and DNL for Property Chemo.C.5b
**Event alphabet:**
- A: correctly identify Patient
- B: administer chemotherapy
- START: Clinic RN gets responsibility for Patient
- END: Clinic RN ends responsibility for Patient
- C: incorrectly identify Patient, make sure that Patient has appropriate I.V. access, make sure that Patient is well enough to receive treatment

**SCOPE:**
1. A restricted interval in the event sequence can have both a starting delimiter, Clinic RN gets responsibility for Patient, and an ending delimiter, Clinic RN ends responsibility for Patient.
2. The behavior is required to hold from an occurrence of Clinic RN gets responsibility for Patient, if it ever occurs, through to the first subsequent occurrence of Clinic RN ends responsibility for Patient, if it ever occurs.
3. If there are multiple occurrences of Clinic RN gets responsibility for Patient without an occurrence of Clinic RN ends responsibility for Patient in between them, only the first of those occurrences of Clinic RN gets responsibility for Patient starts a restricted interval; later occurrences of Clinic RN gets responsibility for Patient within this restricted interval do not have an effect.
4. Clinic RN gets responsibility for Patient is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if Clinic RN gets responsibility for Patient does occur, Clinic RN ends responsibility for Patient is not required to occur subsequently. Even if Clinic RN ends responsibility for Patient does not occur subsequently, the behavior is still required to hold, until the end of the event sequence.
5. There might be many restricted intervals in the event sequence. If Clinic RN gets responsibility for Patient occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of Clinic RN gets responsibility for Patient, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

**BEHAVIOR:**
1. The events of primary interest in this behavior are correctly identify Patient and administer chemotherapy.
2. The events of secondary interest in this behavior are incorrectly identify Patient, make sure that Patient has appropriate I.V. access, and make sure that Patient is well enough to receive treatment.
3. administer chemotherapy is not allowed to occur until after correctly identify Patient occurs.
4. Before the first correctly identify Patient occurs, all the events of secondary interest are allowed to occur zero or more times.
5. correctly identify Patient is required to occur, whether or not administer chemotherapy eventually occurs.
6. administer chemotherapy is not required to occur after correctly identify Patient occurs.
7. After correctly identify Patient occurs, but before the first subsequent administer chemotherapy occurs:
   - correctly identify Patient is not allowed to occur again.
   - All the events of secondary interest are allowed to occur zero or more times.
8. After correctly identify Patient and the first subsequent administer chemotherapy occur:
   - All the events of secondary interest are allowed to occur zero or more times.
   - administer chemotherapy is not allowed to occur again until after another correctly identify Patient occurs; correctly identify Patient is allowed to occur again and, if it does, then the situation is the same as when the first correctly identify Patient occurred, meaning that the restrictions described in parts 6, 7, and 8 would again apply.

FSA and DNL for Property Chemo.D.1a
**Event alphabet:**
- A: Clinic RN gets responsibility for Patient
- B: Clinic RN ends responsibility for Patient
- C: administer chemotherapy, correctly identify Patient, incorrectly identify Patient, make sure that Patient has appropriate I.V. access, make sure that Patient is well enough to receive treatment

**SCOPE:**
1. From the start of any event sequence through to the end of that event sequence, the behavior must hold.

**BEHAVIOR:**
1. The events of primary interest in this behavior are Clinic RN gets responsibility for Patient and Clinic RN ends responsibility for Patient.
2. The events of secondary interest in this behavior are administer chemotherapy, correctly identify Patient, incorrectly identify Patient, make sure that Patient has appropriate I.V. access, and make sure that Patient is well enough to receive treatment.
3. If Clinic RN gets responsibility for Patient occurs, Clinic RN ends responsibility for Patient is required to occur subsequently.
4. Before the first Clinic RN gets responsibility for Patient occurs:
   - Clinic RN ends responsibility for Patient is not allowed to occur;
   - All the events of secondary interest are allowed to occur zero or more times.
5. Clinic RN gets responsibility for Patient is not required to occur.
6. After Clinic RN gets responsibility for Patient occurs, but before the first subsequent Clinic RN ends responsibility for Patient occurs:
   - Clinic RN gets responsibility for Patient is not allowed to occur again;
   - All the events of secondary interest are allowed to occur zero or more times.
7. After Clinic RN gets responsibility for Patient and the first subsequent Clinic RN ends responsibility for Patient occur:
   - All the events of secondary interest are allowed to occur zero or more times;
   - Clinic RN gets responsibility for Patient is not allowed to occur again until after another Clinic RN gets responsibility for Patient occurs. Clinic RN gets responsibility for Patient is allowed to occur again and, if it does, then the situation is the same as when the first Clinic RN gets responsibility for Patient occurred, meaning that the restrictions described in parts 3, 6, and 7 would again apply.

**FSA and DNL for Property Chemo.D.1a1**
**Event alphabet:**

- A: make sure that Patient is well enough to receive treatment
- B: administer chemotherapy
- C: find that Patient is not well enough to receive treatment

**SCOPE:**

1. From the start of any event sequence through to the end of that event sequence, the behavior must hold.

**BEHAVIOR:**

1. The events of primary interest in this behavior are make sure that Patient is well enough to receive treatment and administer chemotherapy.
2. The event of secondary interest in this behavior is find that Patient is not well enough to receive treatment.
3. Administer chemotherapy is not allowed to occur until after make sure that Patient is well enough to receive treatment occurs.
4. Before the first make sure that Patient is well enough to receive treatment occurs, find that Patient is not well enough to receive treatment is allowed to occur zero or more times.
5. Make sure that Patient is well enough to receive treatment is not required to occur.
6. Even if make sure that Patient is well enough to receive treatment does occur, administer chemotherapy is not required to occur after make sure that Patient is well enough to receive treatment occurs.
7. After make sure that Patient is well enough to receive treatment occurs, but before the first subsequent administer chemotherapy occurs:
   - Make sure that Patient is well enough to receive treatment is allowed to occur again, zero or more times;
   - Find that Patient is not well enough to receive treatment is allowed to occur zero or more times.
8. After make sure that Patient is well enough to receive treatment and the first subsequent administer chemotherapy occur:
   - Find that Patient is not well enough to receive treatment is allowed to occur zero or more times;
   - Administer chemotherapy is not allowed to occur again until after another make sure that Patient is well enough to receive treatment occurs; make sure that Patient is well enough to receive treatment is allowed to occur again and, if it does, then the situation is the same as when the first make sure that Patient is well enough to receive treatment occurred, meaning that the restrictions described in parts 6, 7, and 8 would again apply.

FSA and DNL for Property Chemo.D.2a1
Event alphabet:
- A: make sure that Patient is well enough to receive treatment
- START: find that Patient is not well enough to receive treatment
- END: administer chemotherapy

**SCOPE:**
1. A restricted interval in the event sequence can have both a starting delimiter, find that Patient is not well enough to receive treatment, and an ending delimiter, administer chemotherapy.
2. The behavior is required to hold from an occurrence of find that Patient is not well enough to receive treatment, if it ever occurs, through to the first subsequent occurrence of administer chemotherapy, if it ever occurs.
3. If there are multiple occurrences of find that Patient is not well enough to receive treatment without an occurrence of administer chemotherapy in between them, only the last of those occurrences of find that Patient is not well enough to receive treatment potentially starts a restricted interval; each of those occurrences of find that Patient is not well enough to receive treatment resets the beginning of this restricted interval.
4. find that Patient is not well enough to receive treatment is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if find that Patient is not well enough to receive treatment does occur, administer chemotherapy is not required to occur subsequently and if it does not occur subsequently, then the behavior is not required to hold for the remainder of the event sequence.
5. There might be many restricted intervals in the event sequence. If find that Patient is not well enough to receive treatment occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of find that Patient is not well enough to receive treatment, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a potential new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

**BEHAVIOR:**
1. The event of primary interest in this behavior is make sure that Patient is well enough to receive treatment.
2. There are no events of secondary interest in this behavior.
3. make sure that Patient is well enough to receive treatment is required to occur at least once.
### Event alphabet:
- **A**: make sure that Patient has appropriate I.V. access
- **START**: find that Patient does not have appropriate I.V. access
- **END**: administer chemotherapy

### SCOPE:
1. A restricted interval in the event sequence can have both a starting delimiter, **find that Patient does not have appropriate I.V. access**, and an ending delimiter, **administer chemotherapy**.
2. The behavior is required to hold from an occurrence of **find that Patient does not have appropriate I.V. access**, if it ever occurs, through to the first subsequent occurrence of **administer chemotherapy**, if it ever occurs.
3. If there are multiple occurrences of **find that Patient does not have appropriate I.V. access** without an occurrence of **administer chemotherapy** in between them, only the last of those occurrences of **find that Patient does not have appropriate I.V. access** potentially starts a restricted interval; each of those occurrences of **find that Patient does not have appropriate I.V. access** resets the beginning of this restricted interval.
4. **find that Patient does not have appropriate I.V. access** is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if **find that Patient does not have appropriate I.V. access** does occur, **administer chemotherapy** is not required to occur subsequently and if it does not occur subsequently, then the behavior is not required to hold for the remainder of the event sequence.
5. There might be many restricted intervals in the event sequence. If **find that Patient does not have appropriate I.V. access** occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of **find that Patient does not have appropriate I.V. access**, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a potential new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

### Behavior:
1. The event of primary interest in this behavior is **make sure that Patient has appropriate I.V. access**.
2. There are no events of secondary interest in this behavior.
3. **make sure that Patient has appropriate I.V. access** is required to occur at least once.
Event alphabet:
- A: make sure that Patient has appropriate I.V. access
- B: administer chemotherapy
- C: find that Patient does not have appropriate I.V. access

SCOPE:
1. From the start of any event sequence through to the end of that event sequence, the behavior must hold.

BEHAVIOR:
1. The events of primary interest in this behavior are make sure that Patient has appropriate I.V. access and administer chemotherapy.
2. The event of secondary interest in this behavior is find that Patient does not have appropriate I.V. access.
3. administer chemotherapy is not allowed to occur until after make sure that Patient has appropriate I.V. access occurs.
4. Before the first make sure that Patient has appropriate I.V. access occurs, find that Patient does not have appropriate I.V. access is allowed to occur zero or more times.
5. make sure that Patient has appropriate I.V. access is not required to occur.
6. Even if make sure that Patient has appropriate I.V. access does occur, administer chemotherapy is not required to occur after make sure that Patient has appropriate I.V. access occurs.
7. After make sure that Patient has appropriate I.V. access occurs, but before the first subsequent administer chemotherapy occurs:
   - make sure that Patient has appropriate I.V. access is allowed to occur again, zero or more times;
   - find that Patient does not have appropriate I.V. access is allowed to occur zero or more times.
8. After make sure that Patient has appropriate I.V. access and the first subsequent administer chemotherapy occur:
   - find that Patient does not have appropriate I.V. access is allowed to occur zero or more times;
   - administer chemotherapy is not allowed to occur again until after another make sure that Patient has appropriate I.V. access occurs; make sure that Patient has appropriate I.V. access is allowed to occur again and, if it does, then the situation is the same as when the first make sure that Patient has appropriate I.V. access occurred, meaning that the restrictions described in parts 6, 7, and 8 would again apply.

FSA and DNL for Property Chemo.D.3a2
Event alphabet:
- A: correctly identify Patient
- B: make sure that Patient is well enough to receive treatment
- START: Patient arrives for administration of chemotherapy
- END: administer chemotherapy
- C: make sure that Patient has appropriate I.V. access

SCOPE:
1. A restricted interval in the event sequence can have both a starting delimiter, Patient arrives for administration of chemotherapy, and an ending delimiter, administer chemotherapy.
2. The behavior is required to hold from an occurrence of Patient arrives for administration of chemotherapy, if it ever occurs, through to the first subsequent occurrence of administer chemotherapy, if it ever occurs.
3. If there are multiple occurrences of Patient arrives for administration of chemotherapy without an occurrence of administer chemotherapy in between them, only the last of those occurrences of Patient arrives for administration of chemotherapy starts a restricted interval; each of those occurrences of Patient arrives for administration of chemotherapy resets the beginning of this restricted interval.
4. Patient arrives for administration of chemotherapy is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if Patient arrives for administration of chemotherapy does occur, administer chemotherapy is not required to occur subsequently. Even if administer chemotherapy does not occur subsequently, the behavior is still required to hold, until the end of the event sequence.
5. There might be many restricted intervals in the event sequence. If Patient arrives for administration of chemotherapy occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of Patient arrives for administration of chemotherapy, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

BEHAVIOR:
1. The events of primary interest in this behavior are correctly identify Patient and make sure that Patient is well enough to receive treatment.
2. The event of secondary interest in this behavior is make sure that Patient has appropriate I.V. access.
3. make sure that Patient is well enough to receive treatment is not allowed to occur until after correctly identify Patient occurs.
4. Before the first correctly identify Patient occurs, make sure that Patient has appropriate I.V. access is allowed to occur zero or more times.
5. correctly identify Patient is required to occur, whether or not make sure that Patient is well enough to receive treatment eventually occurs.
6. make sure that Patient is well enough to receive treatment is not required to occur after correctly identify Patient occurs.
7. After correctly identify Patient occurs, but before the first subsequent make sure that Patient is well enough to receive treatment occurs:
   - correctly identify Patient is allowed to occur again, zero or more times;
   - make sure that Patient has appropriate I.V. access is not allowed to occur;
8. After correctly identify Patient and the first subsequent make sure that Patient is well enough to receive treatment occur:
   - make sure that Patient has appropriate I.V. access is allowed to occur zero or more times;
   - Both correctly identify Patient and make sure that Patient is well enough to receive treatment are allowed to occur again, zero or more times, and there are no restrictions on the occurrences of any future events.

FSA and DNL for Property Chemo.D.4a (D.1a → D.2a)
Event alphabet:

- A: make sure that Patient is well enough to receive treatment
- B: make sure that Patient has appropriate I.V. access
- START: Patient arrives for administration of chemotherapy
- END: administer chemotherapy
- C: correctly identify Patient

SCOPE:
1. A restricted interval in the event sequence can have both a starting delimiter, Patient arrives for administration of chemotherapy, and an ending delimiter, administer chemotherapy.
2. The behavior is required to hold from an occurrence of Patient arrives for administration of chemotherapy, if it ever occurs, through to the first subsequent occurrence of administer chemotherapy, if it ever occurs.
3. If there are multiple occurrences of Patient arrives for administration of chemotherapy without an occurrence of administer chemotherapy in between them, only the last of those occurrences of Patient arrives for administration of chemotherapy starts a restricted interval, each of the other occurrences of Patient arrives for administration of chemotherapy resets the beginning of this restricted interval.
4. Patient arrives for administration of chemotherapy is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if Patient arrives for administration of chemotherapy does occur, administer chemotherapy is not required to occur subsequently. Even if administer chemotherapy does not occur subsequently, the behavior is still required to hold, until the end of the event sequence.
5. There might be many restricted intervals in the event sequence. If Patient arrives for administration of chemotherapy occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of Patient arrives for administration of chemotherapy, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

BEHAVIOR:
1. The events of primary interest in this behavior are make sure that Patient is well enough to receive treatment and make sure that Patient has appropriate I.V. access.
2. The event of secondary interest in this behavior is correctly identify Patient.
3. make sure that Patient has appropriate I.V. access is not allowed to occur until after make sure that Patient is well enough to receive treatment occurs.
4. Before the first make sure that Patient is well enough to receive treatment occurs, correctly identify Patient is allowed to occur zero or more times.
5. make sure that Patient is well enough to receive treatment is not required to occur.
6. Even if make sure that Patient is well enough to receive treatment does occur, make sure that Patient has appropriate I.V. access is not required to occur after make sure that Patient is well enough to receive treatment occurs.
7. After make sure that Patient is well enough to receive treatment occurs, but before the first subsequent make sure that Patient has appropriate I.V. access occurs:
   - correctly identify Patient is allowed to occur again, zero or more times;
   - make sure that Patient has appropriate I.V. access is not allowed to occur.
8. After make sure that Patient is well enough to receive treatment and the first subsequent make sure that Patient has appropriate I.V. access occurs:
   - correctly identify Patient is allowed to occur zero or more times;
   - Both make sure that Patient is well enough to receive treatment and make sure that Patient has appropriate I.V. access are allowed to occur again, zero or more times, and there are no restrictions on the occurrences of any future events.

FSA and DNL for Property Chemo.D.4a (D.2a → D.3a)

422
Event alphabet:
- A: correctly identify Patient
- B: administer pre-chemotherapy supportive care medications
- END: Clinic RN ends responsibility for Patient
- C: incorrectly identify Patient, make sure Patient has appropriate I.V. access, make sure Patient is well enough to receive treatment

**SCOPE:**
1. A restricted interval in the event sequence can have both a starting delimiter, Clinic RN gets responsibility for Patient, and an ending delimiter, Clinic RN ends responsibility for Patient.
2. The behavior is required to hold from an occurrence of Clinic RN gets responsibility for Patient, if it ever occurs, through to the first subsequent occurrence of Clinic RN ends responsibility for Patient, if it ever occurs.
3. If there are multiple occurrences of Clinic RN gets responsibility for Patient without an occurrence of Clinic RN ends responsibility for Patient in between them, only the first of those occurrences of Clinic RN gets responsibility for Patient starts a restricted interval; later occurrences of Clinic RN gets responsibility for Patient within this restricted interval do not have an effect.
4. Clinic RN gets responsibility for Patient is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if Clinic RN gets responsibility for Patient does occur, Clinic RN ends responsibility for Patient is not required to occur subsequently. Even if Clinic RN ends responsibility for Patient does not occur subsequently, the behavior is still required to hold, until the end of the event sequence.
5. There might be many restricted intervals in the event sequence. If Clinic RN gets responsibility for Patient occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of Clinic RN gets responsibility for Patient, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

**BEHAVIOR:**
1. The events of primary interest in this behavior are correctly identify Patient and administer pre-chemotherapy supportive care medications.
2. The events of secondary interest in this behavior are incorrectly identify Patient, make sure Patient has appropriate I.V. access, and make sure Patient is well enough to receive treatment.
3. administer pre-chemotherapy supportive care medications is not allowed to occur until after correctly identify Patient occurs.
4. Before the first correctly identify Patient occurs, all the events of secondary interest are allowed to occur zero or more times.
5. correctly identify Patient is required to occur, whether or not administer pre-chemotherapy supportive care medications eventually occurs.
6. administer pre-chemotherapy supportive care medications is not required to occur after correctly identify Patient occurs.
7. After correctly identify Patient occurs, but before the first subsequent administer pre-chemotherapy supportive care medications occurs:
   - correctly identify Patient is allowed to occur again, zero or more times;
   - All the events of secondary interest are allowed to occur zero or more times.
8. After correctly identify Patient and the first subsequent administer pre-chemotherapy supportive care medications occur:
   - All the events of secondary interest are allowed to occur zero or more times;
   - administer pre-chemotherapy supportive care medications is not allowed to occur again until after another correctly identify Patient occurs; correctly identify Patient is allowed to occur again and, if it does, then the situation is the same as when the first correctly identify Patient occurred, meaning that the restrictions described in parts 6, 7, and 8 would again apply.

FSA and DNL for Property Chemo.D.1b
Event alphabet:
- A: make sure that Patient is well enough to receive treatment
- B: administer pre-chemotherapy supportive care medications
- C: find that Patient is not well enough to receive treatment

**SCOPE:**
1. From the start of any event sequence through to the end of that event sequence, the behavior must hold.

**BEHAVIOR:**
1. The events of primary interest in this behavior are make sure that Patient is well enough to receive treatment and administer pre-chemotherapy supportive care medications.
2. The event of secondary interest in this behavior is find that Patient is not well enough to receive treatment.
3. administer pre-chemotherapy supportive care medications is not allowed to occur until after make sure that Patient is well enough to receive treatment occurs.
4. Before the first make sure that Patient is well enough to receive treatment occurs, find that Patient is not well enough to receive treatment is allowed to occur zero or more times.
5. make sure that Patient is well enough to receive treatment is not required to occur.
6. Even if make sure that Patient is well enough to receive treatment occurs, administer pre-chemotherapy supportive care medications is not allowed to occur after make sure that Patient is well enough to receive treatment occurs.
7. After make sure that Patient is well enough to receive treatment occurs, but before the first subsequent administer pre-chemotherapy supportive care medications occurs:
   - make sure that Patient is well enough to receive treatment is allowed to occur again, zero or more times;
   - find that Patient is not well enough to receive treatment is allowed to occur zero or more times.
8. After make sure that Patient is well enough to receive treatment and the first subsequent administer pre-chemotherapy supportive care medications occur:
   - find that Patient is not well enough to receive treatment is allowed to occur zero or more times;
   - administer pre-chemotherapy supportive care medications is not allowed to occur again until after another make sure that Patient is well enough to receive treatment occurs, make sure that Patient is well enough to receive treatment is allowed to occur again and, if it does, then the situation is the same as when the first make sure that Patient is well enough to receive treatment occurred, meaning that the restrictions described in parts 6, 7, and 8 would again apply.

FSA and DNL for Property Chemo.D.2b1
**SCOPE:**
1. A restricted interval in the event sequence can have both a starting delimiter, **find that Patient is not well enough to receive treatment**, and an ending delimiter, **administer pre-chemotherapy supportive care medications**.
2. The behavior is required to hold from an occurrence of **find that Patient is not well enough to receive treatment**, if it ever occurs, through to the first subsequent occurrence of **administer pre-chemotherapy supportive care medications**, if it ever occurs.
3. If there are multiple occurrences of **find that Patient is not well enough to receive treatment** without an occurrence of **administer pre-chemotherapy supportive care medications** in between them, only the last of those occurrences of **find that Patient is not well enough to receive treatment** potentially starts a restricted interval; each of those occurrences of **find that Patient is not well enough to receive treatment** resets the beginning of this restricted interval.
4. **find that Patient is not well enough to receive treatment** is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if **find that Patient is not well enough to receive treatment** does occur, **administer pre-chemotherapy supportive care medications** is not required to occur subsequently and if it does not occur subsequently, then the behavior is not required to hold for the remainder of the event sequence.
5. There might be many restricted intervals in the event sequence. If **find that Patient is not well enough to receive treatment** occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of **find that Patient is not well enough to receive treatment**, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a potential new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

**BEHAVIOR:**
1. The event of primary interest in this behavior is **make sure that Patient is well enough to receive treatment**.
2. There are no events of secondary interest in this behavior.
3. **make sure that Patient is well enough to receive treatment** is required to occur at least once.

FSA and DNL for Property Chemo.D.2b2
**Event alphabet:**
- **A**: make sure that Patient has appropriate I.V. access
- **START**: find that Patient does not have appropriate I.V. access
- **END**: administer pre-chemotherapy supportive care medications

**SCOPE:**
1. A restricted interval in the event sequence can have both a starting delimiter, **find that Patient does not have appropriate I.V. access**, and an ending delimiter, **administer pre-chemotherapy supportive care medications**.
2. The behavior is required to hold from an occurrence of **find that Patient does not have appropriate I.V. access**, if it ever occurs, through to the first subsequent occurrence of **administer pre-chemotherapy supportive care medications**, if it ever occurs.
3. If there are multiple occurrences of **find that Patient does not have appropriate I.V. access** without an occurrence of **administer pre-chemotherapy supportive care medications** in between, only the last of those occurrences of **find that Patient does not have appropriate I.V. access** potentially starts a restricted interval; each of those occurrences of **find that Patient does not have appropriate I.V. access** resets the beginning of this restricted interval.
4. **find that Patient does not have appropriate I.V. access** is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if **find that Patient does not have appropriate I.V. access** does occur, **administer pre-chemotherapy supportive care medications** is not required to occur subsequently and if it does not occur subsequently, then the behavior is not required to hold for the remainder of the event sequence.
5. There might be many restricted intervals in the event sequence. If **find that Patient does not have appropriate I.V. access** occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of **find that Patient does not have appropriate I.V. access**, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a potential new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

**BEHAVIOR:**
1. The event of primary interest in this behavior is **make sure that Patient has appropriate I.V. access**.
2. There are no events of secondary interest in this behavior.
3. **make sure that Patient has appropriate I.V. access** is required to occur at least once.

FSA and DNL for Property Chemo.D.3b1
Event alphabet:
- A: make sure that Patient has appropriate I.V. access
- B: administer pre-chemotherapy supportive care medications
- C: find that Patient does not have appropriate I.V. access

SCOPE:
1. From the start of any event sequence through to the end of that event sequence, the behavior must hold.

BEHAVIOR:
1. The events of primary interest in this behavior are make sure that Patient has appropriate I.V. access and administer pre-chemotherapy supportive care medications.
2. The event of secondary interest in this behavior is find that Patient does not have appropriate I.V. access.
3. administer pre-chemotherapy supportive care medications is not allowed to occur until after make sure that Patient has appropriate I.V. access occurs.
4. Before the first make sure that Patient has appropriate I.V. access occurs, find that Patient does not have appropriate I.V. access is allowed to occur zero or more times.
5. make sure that Patient has appropriate I.V. access is not required to occur.
6. Even if make sure that Patient has appropriate I.V. access does occur, administer pre-chemotherapy supportive care medications is not required to occur after make sure that Patient has appropriate I.V. access occurs.
7. After make sure that Patient has appropriate I.V. access occurs, but before the first subsequent administer pre-chemotherapy supportive care medications occurs:
   - make sure that Patient has appropriate I.V. access is allowed to occur again, zero or more times;
   - find that Patient does not have appropriate I.V. access is allowed to occur zero or more times.
8. After make sure that Patient has appropriate I.V. access and the first subsequent administer pre-chemotherapy supportive care medications occur:
   - find that Patient does not have appropriate I.V. access is allowed to occur zero or more times;
   - administer pre-chemotherapy supportive care medications is not allowed to occur again until after another make sure that Patient has appropriate I.V. access occurs; make sure that Patient has appropriate I.V. access is allowed to occur again and, if it does, then the situation is the same as when the first make sure that Patient has appropriate I.V. access occurred, meaning that the restrictions described in parts 6, 7, and 8 would again apply.

FSA and DNL for Property Chemo.D.3b2
Event alphabet:
- A: correctly identify Patient
- B: make sure that Patient is well enough to receive treatment
- START: Patient arrives for administration of chemotherapy
- END: administer pre-chemotherapy supportive care medications
- C: make sure that Patient has appropriate I.V. access

SCOPE:
1. A restricted interval in the event sequence can have both a starting delimiter, Patient arrives for administration of chemotherapy, and an ending delimiter, administer pre-chemotherapy supportive care medications.
2. The behavior is required to hold from an occurrence of Patient arrives for administration of chemotherapy, if it ever occurs, through to the first subsequent occurrence of administer pre-chemotherapy supportive care medications, if it ever occurs.
3. If there are multiple occurrences of Patient arrives for administration of chemotherapy without an occurrence of administer pre-chemotherapy supportive care medications in between them, only the last of those occurrences of Patient arrives for administration of chemotherapy starts a restricted interval; each of those occurrences of Patient arrives for administration of chemotherapy resets the beginning of this restricted interval.
4. Patient arrives for administration of chemotherapy is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if Patient arrives for administration of chemotherapy does occur, administer pre-chemotherapy supportive care medications is not required to occur subsequently. Even if administer pre-chemotherapy supportive care medications does not occur subsequently, the behavior is still required to hold, until the end of the event sequence.
5. There might be many restricted intervals in the event sequence. If Patient arrives for administration of chemotherapy occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of Patient arrives for administration of chemotherapy, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

BEHAVIOR:
1. The events of primary interest in this behavior are correctly identify Patient and make sure that Patient is well enough to receive treatment.
2. The event of secondary interest in this behavior is make sure that Patient has appropriate I.V. access.
3. make sure that Patient is well enough to receive treatment is not allowed to occur until after correctly identify Patient occurs.
4. Before the first correctly identify Patient occurs, make sure that Patient has appropriate I.V. access is allowed to occur zero or more times.
5. correctly identify Patient is required to occur, whether or not make sure that Patient is well enough to receive treatment eventually occurs.
6. make sure that Patient is well enough to receive treatment is not required to occur after correctly identify Patient occurs.
7. After correctly identify Patient occurs, but before the first subsequent make sure that Patient is well enough to receive treatment occurs:
   - correctly identify Patient is allowed to occur again, zero or more times;
   - make sure that Patient has appropriate I.V. access is not allowed to occur.
8. After correctly identify Patient and the first subsequent make sure that Patient is well enough to receive treatment occur:
   - make sure that Patient has appropriate I.V. access is allowed to occur zero or more times;
   - Both correctly identify Patient and make sure that Patient is well enough to receive treatment are allowed to occur again, zero or more times, and there are no restrictions on the occurrences of any future events.

FSA and DNL for Property Chemo.D.4b (D.1b → D.2b)
EVENT ALPHABET:

- **A**: make sure that Patient is well enough to receive treatment
- **B**: make sure that Patient has appropriate I.V. access
- **START**: Patient arrives for administration of chemotherapy
- **END**: administer pre-chemotherapy supportive care medications
- **C**: correctly identify Patient

SCOPE:

1. A restricted interval in the event sequence can have both a starting delimiter, **Patient arrives for administration of chemotherapy**, and an ending delimiter, **administer pre-chemotherapy supportive care medications**.
2. The behavior is required to hold from an occurrence of **Patient arrives for administration of chemotherapy**, if it ever occurs, through to the first subsequent occurrence of **administer pre-chemotherapy supportive care medications**, if it ever occurs.
3. If there are multiple occurrences of **Patient arrives for administration of chemotherapy** without an occurrence of **administer pre-chemotherapy supportive care medications** in between them, only the last of those occurrences of **Patient arrives for administration of chemotherapy** starts a restricted interval; each of those occurrences of **Patient arrives for administration of chemotherapy** resets the beginning of this restricted interval.
4. **Patient arrives for administration of chemotherapy** is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if **Patient arrives for administration of chemotherapy** does occur, **administer pre-chemotherapy supportive care medications** is not required to occur subsequently. Even if **administer pre-chemotherapy supportive care medications** does not occur subsequently, the behavior is still required to hold, until the end of the event sequence.
5. There might be many restricted intervals in the event sequence. If **Patient arrives for administration of chemotherapy** occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of **Patient arrives for administration of chemotherapy**, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

BEHAVIOR:

1. The events of primary interest in this behavior are **make sure that Patient is well enough to receive treatment** and **make sure that Patient has appropriate I.V. access**.
2. The event of secondary interest in this behavior is **correctly identify Patient**.
3. **make sure that Patient has appropriate I.V. access** is not allowed to occur until after **make sure that Patient is well enough to receive treatment** occurs.
4. Before the first **make sure that Patient is well enough to receive treatment** occurs, **correctly identify Patient** is allowed to occur zero or more times.
5. **make sure that Patient is well enough to receive treatment** is not required to occur.
6. Even if **make sure that Patient is well enough to receive treatment** does occur, **make sure that Patient has appropriate I.V. access** is not required to occur after **make sure that Patient is well enough to receive treatment** occurs.
7. After **make sure that Patient is well enough to receive treatment** occurs, but before the first subsequent **make sure that Patient has appropriate I.V. access** occurs:
   - **make sure that Patient is well enough to receive treatment** is allowed to occur again, zero or more times;
   - **correctly identify Patient** is not allowed to occur.
8. **make sure that Patient has appropriate I.V. access** occurs:
   - **correctly identify Patient** is allowed to occur zero or more times;
   - Both **make sure that Patient is well enough to receive treatment** and **make sure that Patient has appropriate I.V. access** are allowed to occur again, zero or more times, and there are no restrictions on the occurrences of any future events.

FSA and DNL for Property Chemo.D.4b (D.2b → D.3b)
**Event alphabet:**
- A: make sure chemotherapy drugs match the treatment plan and chemotherapy orders
- B: administer chemotherapy

**SCOPE:**
1. From the start of any event sequence through to the end of that event sequence, the behavior must hold.

**BEHAVIOR:**
1. The events of primary interest in this behavior are make sure chemotherapy drugs match the treatment plan and chemotherapy orders and administer chemotherapy.
2. There are no events of secondary interest in this behavior.
3. administer chemotherapy is not allowed to occur until after make sure chemotherapy drugs match the treatment plan and chemotherapy orders occurs.
4. make sure chemotherapy drugs match the treatment plan and chemotherapy orders is not required to occur.
5. Even if make sure chemotherapy drugs match the treatment plan and chemotherapy orders does occur, administer chemotherapy is not required to occur after make sure chemotherapy drugs match the treatment plan and chemotherapy orders occurs.
6. After make sure chemotherapy drugs match the treatment plan and chemotherapy orders occurs, but before the first subsequent administer chemotherapy occurs, make sure chemotherapy drugs match the treatment plan and chemotherapy orders is allowed to occur again, zero or more times.
7. After make sure chemotherapy drugs match the treatment plan and chemotherapy orders and the first subsequent administer chemotherapy occur:
   - administer chemotherapy is not allowed to occur again until after another make sure chemotherapy drugs match the treatment plan and chemotherapy orders occurs. make sure chemotherapy drugs match the treatment plan and chemotherapy orders occurs is allowed to occur again and, if it does, then the situation is the same as when the first make sure chemotherapy drugs match the treatment plan and chemotherapy orders occurred, meaning that the restrictions described in parts 5, 6, and 7 would again apply.

**FSA and DNL for Property Chemo.D.10a**
Event alphabet:
- A: resolve that mismatch between chemotherapy drug i and the treatment plan or chemotherapy orders
- START: chemotherapy drug i differs from the treatment plan or chemotherapy orders
- END: administer chemotherapy

**SCOPE:**
1. A restricted interval in the event sequence can have both a starting delimiter, chemotherapy drug i differs from the treatment plan or chemotherapy orders, and an ending delimiter, administer chemotherapy.
2. The behavior is required to hold from an occurrence of chemotherapy drug i differs from the treatment plan or chemotherapy orders, if it ever occurs, through to the first subsequent occurrence of administer chemotherapy, if it ever occurs.
3. If there are multiple occurrences of chemotherapy drug i differs from the treatment plan or chemotherapy orders without an occurrence of administer chemotherapy in between them, only the last of those occurrences of chemotherapy drug i differs from the treatment plan or chemotherapy orders potentially starts a restricted interval; each of those occurrences of chemotherapy drug i differs from the treatment plan or chemotherapy orders resets the beginning of this restricted interval.
4. chemotherapy drug i differs from the treatment plan or chemotherapy orders is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if chemotherapy drug i differs from the treatment plan or chemotherapy orders does occur, administer chemotherapy is not required to occur subsequently and if it does not occur subsequently, then the behavior is not required to hold for the remainder of the event sequence.
5. There might be many restricted intervals in the event sequence. If chemotherapy drug i differs from the treatment plan or chemotherapy orders occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of chemotherapy drug i differs from the treatment plan or chemotherapy orders, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a potential new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

**BEHAVIOR:**
1. The event of primary interest in this behavior is resolve that mismatch between chemotherapy drug i and the treatment plan or chemotherapy orders.
2. There are no events of secondary interest in this behavior.
3. resolve that mismatch between chemotherapy drug i and the treatment plan or chemotherapy orders is required to occur at least once.

FSA and DNL for Property Chemo.D.10b
Event alphabet:
- A: make sure chemotherapy drugs are in doses that are consistent with Patient data
- B: administer chemotherapy

SCOPE:
1. From the start of any event sequence through to the end of that event sequence, the behavior must hold.

BEHAVIOR:
1. The events of primary interest in this behavior are make sure chemotherapy drugs are in doses that are consistent with Patient data and administer chemotherapy.
2. There are no events of secondary interest in this behavior.
3. administer chemotherapy is not allowed to occur until after make sure chemotherapy drugs are in doses that are consistent with Patient data occurs.
4. make sure chemotherapy drugs are in doses that are consistent with Patient data is not required to occur.
5. Even if make sure chemotherapy drugs are in doses that are consistent with Patient data does occur, administer chemotherapy is not required to occur after make sure chemotherapy drugs are in doses that are consistent with Patient data occurs.
6. After make sure chemotherapy drugs are in doses that are consistent with Patient data occurs, but before the first subsequent administer chemotherapy occurs, make sure chemotherapy drugs are in doses that are consistent with Patient data is allowed to occur again, zero or more times.
7. After make sure chemotherapy drugs are in doses that are consistent with Patient data and the first subsequent administer chemotherapy occur:
   - administer chemotherapy is not allowed to occur again until after another make sure chemotherapy drugs are in doses that are consistent with Patient data occurs; make sure chemotherapy drugs are in doses that are consistent with Patient data is allowed to occur again and, if it does, then the situation is the same as when the first make sure chemotherapy drugs are in doses that are consistent with Patient data occurred, meaning that the restrictions described in parts 5, 6, and 7 would again apply.

FSA and DNL for Property Chemo.D.11a
Event alphabet:
- A: resolve inconsistency between chemotherapy drug i and Patient data
- START: chemotherapy drug dose i becomes inconsistent with Patient data
- END: administer chemotherapy

**SCOPE:**
1. A restricted interval in the event sequence can have both a starting delimiter, chemotherapy drug dose i becomes inconsistent with Patient data, and an ending delimiter, administer chemotherapy.
2. The behavior is required to hold from an occurrence of chemotherapy drug dose i becomes inconsistent with Patient data, if it ever occurs, through to the first subsequent occurrence of administer chemotherapy, if it ever occurs.
3. If there are multiple occurrences of chemotherapy drug dose i becomes inconsistent with Patient data without an occurrence of administer chemotherapy in between them, only the last of those occurrences of chemotherapy drug dose i becomes inconsistent with Patient data potentially starts a restricted interval; each of those occurrences of chemotherapy drug dose i becomes inconsistent with Patient data resets the beginning of this restricted interval.
4. chemotherapy drug dose i becomes inconsistent with Patient data is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if chemotherapy drug dose i becomes inconsistent with Patient data does occur, administer chemotherapy is not required to occur subsequently and if it does not occur subsequently, then the behavior is not required to hold for the remainder of the event sequence.
5. There might be many restricted intervals in the event sequence. If chemotherapy drug dose i becomes inconsistent with Patient data occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of chemotherapy drug dose i becomes inconsistent with Patient data, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a potential new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

**BEHAVIOR:**
1. The event of primary interest in this behavior is resolve inconsistency between chemotherapy drug i and Patient data.
2. There are no events of secondary interest in this behavior.
3. resolve inconsistency between chemotherapy drug i and Patient data is required to occur at least once.

FSA and DNL for Property Chemo.D.11b

433
Event alphabet:
- A: correctly prepare a new dose of chemotherapy drug i
- B: administer chemotherapy
- C: incorrectly prepare a dose of chemotherapy drug i

SCOPE:
1. From the start of any event sequence through to the end of that event sequence, the behavior must hold.

BEHAVIOR:
1. The events of primary interest in this behavior are correctly prepare a new dose of chemotherapy drug i and administer chemotherapy.
2. The event of secondary interest in this behavior is incorrectly prepare a dose of chemotherapy drug i.
3. administer chemotherapy is not allowed to occur until after correctly prepare a new dose of chemotherapy drug i occurs.
4. Before the first correctly prepare a new dose of chemotherapy drug i occurs, incorrectly prepare a dose of chemotherapy drug i is allowed to occur zero or more times.
5. correctly prepare a new dose of chemotherapy drug i is not required to occur.
6. Even if correctly prepare a new dose of chemotherapy drug i does occur, administer chemotherapy is not required to occur after correctly prepare a new dose of chemotherapy drug i occurs.
7. After correctly prepare a new dose of chemotherapy drug i occurs, but before the first subsequent administer chemotherapy occurs:
   - correctly prepare a new dose of chemotherapy drug i is not allowed to occur again;
   - incorrectly prepare a dose of chemotherapy drug i is allowed to occur zero or more times.
8. After correctly prepare a new dose of chemotherapy drug i and the first subsequent administer chemotherapy occur:
   - incorrectly prepare a dose of chemotherapy drug i is allowed to occur zero or more times;
   - administer chemotherapy is not allowed to occur again until after another correctly prepare a new dose of chemotherapy drug i occurs; correctly prepare a new dose of chemotherapy drug i is allowed to occur again and, if it does, then the situation is the same as when the first correctly prepare a new dose of chemotherapy drug i occurred, meaning that the restrictions described in parts 6, 7, and 8 would again apply.

FSA and DNL for Property Chemo.D.12a
**Event alphabet:**

- A: correctly prepare a new dose of chemotherapy drug $i$
- START: incorrectly prepare a dose of chemotherapy drug $i$
- END: administer chemotherapy

**SCOPE:**

1. A restricted interval in the event sequence can have both a starting delimiter, incorrectly prepare a dose of chemotherapy drug $i$, and an ending delimiter, administer chemotherapy.
2. The behavior is required to hold from an occurrence of incorrectly prepare a dose of chemotherapy drug $i$, if it ever occurs, through to the first subsequent occurrence of administer chemotherapy, if it ever occurs.
3. If there are multiple occurrences of incorrectly prepare a dose of chemotherapy drug $i$ without an occurrence of administer chemotherapy in between them, only the last of those occurrences of incorrectly prepare a dose of chemotherapy drug $i$ potentially starts a restricted interval; each of those occurrences of incorrectly prepare a dose of chemotherapy drug $i$ resets the beginning of this restricted interval.
4. incorrectly prepare a dose of chemotherapy drug $i$ is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if incorrectly prepare a dose of chemotherapy drug $i$ does occur, administer chemotherapy is not required to occur subsequently and if it does not occur subsequently, then the behavior is not required to hold for the remainder of the event sequence.
5. There might be many restricted intervals in the event sequence. If incorrectly prepare a dose of chemotherapy drug $i$ occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of incorrectly prepare a dose of chemotherapy drug $i$, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a potential new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

**BEHAVIOR:**

1. The event of primary interest in this behavior is correctly prepare a new dose of chemotherapy drug $i$.
2. There are no events of secondary interest in this behavior.
3. correctly prepare a new dose of chemotherapy drug $i$ is required to occur exactly once.

FSA and DNL for Property Chemo.D.12b
Event alphabet:
- A: assign chemotherapy drugs to Patient
- B: administer chemotherapy
- C: remove assignment of chemotherapy drugs from Patient

**SCOPE:**
1. From the start of any event sequence through to the end of that event sequence, the behavior must hold.

**BEHAVIOR:**
1. The events of primary interest in this behavior are assign chemotherapy drugs to Patient and administer chemotherapy.
2. The event of secondary interest in this behavior is remove assignment of chemotherapy drugs from Patient.
3. administer chemotherapy is not allowed to occur until after assign chemotherapy drugs to Patient occurs.
4. Before the first assign chemotherapy drugs to Patient occurs, remove assignment of chemotherapy drugs from Patient is allowed to occur zero or more times.
5. assign chemotherapy drugs to Patient is not required to occur.
6. Even if assign chemotherapy drugs to Patient does occur, administer chemotherapy is not required to occur after assign chemotherapy drugs to Patient occurs.
7. After assign chemotherapy drugs to Patient occurs, but before the first subsequent administer chemotherapy occurs:
   - assign chemotherapy drugs to Patient is not allowed to occur again;
   - remove assignment of chemotherapy drugs from Patient is allowed to occur zero or more times.
8. After assign chemotherapy drugs to Patient and the first subsequent administer chemotherapy occur:
   - remove assignment of chemotherapy drugs from Patient is allowed to occur zero or more times;
   - administer chemotherapy is not allowed to occur again until after another assign chemotherapy drugs to Patient occurs; assign chemotherapy drugs to Patient is allowed to occur again and, if it does, then the situation is the same as when the first assign chemotherapy drugs to Patient occurred, meaning that the restrictions described in parts 6, 7, and 8 would again apply.

FSA and DNL for Property Chemo.D.13a
Event alphabet:
- **A**: assign chemotherapy drugs to Patient
- **START**: remove assignment of chemotherapy drugs from Patient
- **END**: administer chemotherapy

**SCOPE:**
1. A restricted interval in the event sequence can have both a starting delimiter, **remove assignment of chemotherapy drugs from Patient**, and an ending delimiter, **administer chemotherapy**.
2. The behavior is required to hold from an occurrence of **remove assignment of chemotherapy drugs from Patient**, if it ever occurs, through to the first subsequent occurrence of **administer chemotherapy**, if it ever occurs.
3. If there are multiple occurrences of **remove assignment of chemotherapy drugs from Patient** without an occurrence of **administer chemotherapy** in between them, only the last of those occurrences of **remove assignment of chemotherapy drugs from Patient** potentially starts a restricted interval; each of those occurrences of **remove assignment of chemotherapy drugs from Patient** resets the beginning of this restricted interval.
4. **remove assignment of chemotherapy drugs from Patient** is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if **remove assignment of chemotherapy drugs from Patient** does occur, **administer chemotherapy** is not required to occur subsequently and if it does not occur subsequently, then the behavior is not required to hold for the remainder of the event sequence.
5. There might be many restricted intervals in the event sequence. If **remove assignment of chemotherapy drugs from Patient** occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of **remove assignment of chemotherapy drugs from Patient**, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a potential new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

**BEHAVIOR:**
1. The event of primary interest in this behavior is **assign chemotherapy drugs to Patient**.
2. There are no events of secondary interest in this behavior.
3. **assign chemotherapy drugs to Patient** is required to occur exactly once.

FSA and DNL for Property Chemo.D.13b
Event alphabet:
- A: create a new dose of chemotherapy drug \( i \)
- B: administer chemotherapy
- C: chemotherapy drug \( i \) becomes physically unsuitable for administration

**SCOPE:**
1. From the start of any event sequence through to the end of that event sequence, the behavior must hold.

**BEHAVIOR:**
1. The events of primary interest in this behavior are create a new dose of chemotherapy drug \( i \) and administer chemotherapy.
2. The event of secondary interest in this behavior is chemotherapy drug \( i \) becomes physically unsuitable for administration.
3. administer chemotherapy is not allowed to occur until after create a new dose of chemotherapy drug \( i \) occurs.
4. Before the first create a new dose of chemotherapy drug \( i \) occurs, chemotherapy drug \( i \) becomes physically unsuitable for administration is allowed to occur zero or more times.
5. create a new dose of chemotherapy drug \( i \) is not required to occur.
6. Even if create a new dose of chemotherapy drug \( i \) does occur, administer chemotherapy is not required to occur after create a new dose of chemotherapy drug \( i \) occurs.
7. After create a new dose of chemotherapy drug \( i \) occurs, but before the first subsequent administer chemotherapy occurs:
   - create a new dose of chemotherapy drug \( i \) is not allowed to occur again;
   - chemotherapy drug \( i \) becomes physically unsuitable for administration is allowed to occur zero or more times.
8. After create a new dose of chemotherapy drug \( i \) and the first subsequent administer chemotherapy occur:
   - chemotherapy drug \( i \) becomes physically unsuitable for administration is allowed to occur zero or more times;
   - administer chemotherapy is not allowed to occur again until after another create a new dose of chemotherapy drug \( i \) occurs;
   - create a new dose of chemotherapy drug \( i \) is allowed to occur again and, if it does, then the situation is the same as when the first create a new dose of chemotherapy drug \( i \) occurred, meaning that the restrictions described in parts 6, 7, and 8 would again apply.

FSA and DNL for Property Chemo.D.8a
Event alphabet:

- A: create a new dose of chemotherapy drug i
- START: chemotherapy drug i becomes physically unsuitable for administration
- END: administer chemotherapy

**Scope:**
1. A restricted interval in the event sequence can have both a starting delimiter, chemotherapy drug i becomes physically unsuitable for administration, and an ending delimiter, administer chemotherapy.
2. The behavior is required to hold from an occurrence of chemotherapy drug i becomes physically unsuitable for administration, if it ever occurs, through to the first subsequent occurrence of administer chemotherapy, if it ever occurs.
3. If there are multiple occurrences of chemotherapy drug i becomes physically unsuitable for administration without an occurrence of administer chemotherapy in between them, only the last of those occurrences of chemotherapy drug i becomes physically unsuitable for administration potentially starts a restricted interval; each of those occurrences of chemotherapy drug i becomes physically unsuitable for administration resets the beginning of this restricted interval.
4. chemotherapy drug i becomes physically unsuitable for administration is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if chemotherapy drug i becomes physically unsuitable for administration does occur, administer chemotherapy is not required to occur subsequently and if it does not occur subsequently, then the behavior is not required to hold for the remainder of the event sequence.
5. There might be many restricted intervals in the event sequence. If chemotherapy drug i becomes physically unsuitable for administration occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of chemotherapy drug i becomes physically unsuitable for administration, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a potential new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

**Behavior:**
1. The event of primary interest in this behavior is create a new dose of chemotherapy drug i.
2. There are no events of secondary interest in this behavior.
3. create a new dose of chemotherapy drug i is required to occur exactly once.

FSA and DNL for Property Chemo.D.8b
Event alphabet:
- A: make sure chemotherapy drugs are consistent with cancer diagnosis
- B: administer chemotherapy

SCOPE:
1. From the start of any event sequence through to the end of that event sequence, the behavior must hold.

BEHAVIOR:
1. The events of primary interest in this behavior are make sure chemotherapy drugs are consistent with cancer diagnosis and administer chemotherapy.
2. There are no events of secondary interest in this behavior.
3. administer chemotherapy is not allowed to occur until after make sure chemotherapy drugs are consistent with cancer diagnosis occurs.
4. make sure chemotherapy drugs are consistent with cancer diagnosis is not required to occur.
5. Even if make sure chemotherapy drugs are consistent with cancer diagnosis does occur, administer chemotherapy is not required to occur after make sure chemotherapy drugs are consistent with cancer diagnosis occurs.
6. After make sure chemotherapy drugs are consistent with cancer diagnosis occurs, but before the first subsequent administer chemotherapy occurs, make sure chemotherapy drugs are consistent with cancer diagnosis is allowed to occur again, zero or more times.
7. After make sure chemotherapy drugs are consistent with cancer diagnosis and the first subsequent administer chemotherapy occur:
   - administer chemotherapy is not allowed to occur again until after another make sure chemotherapy drugs are consistent with cancer diagnosis occurs. make sure chemotherapy drugs are consistent with cancer diagnosis is allowed to occur again and, if it does, then the situation is the same as when the first make sure chemotherapy drugs are consistent with cancer diagnosis occurred, meaning that the restrictions described in parts 5, 6, and 7 would again apply.

FSA and DNL for Property Chemo.D.9a
Event alphabet:
- A: resolve inconsistency between chemotherapy drug i and the cancer diagnosis
- START: chemotherapy drug i becomes inconsistent with the cancer diagnosis
- END: administer chemotherapy

**SCOPE:**
1. A restricted interval in the event sequence can have both a starting delimiter, chemotherapy drug i becomes inconsistent with the cancer diagnosis, and an ending delimiter, administer chemotherapy.
2. The behavior is required to hold from an occurrence of chemotherapy drug i becomes inconsistent with the cancer diagnosis, if it ever occurs, through to the first subsequent occurrence of administer chemotherapy, if it ever occurs.
3. If there are multiple occurrences of chemotherapy drug i becomes inconsistent with the cancer diagnosis without an occurrence of administer chemotherapy in between them, only the last of those occurrences of chemotherapy drug i becomes inconsistent with the cancer diagnosis potentially starts a restricted interval; each of those occurrences of chemotherapy drug i becomes inconsistent with the cancer diagnosis resets the beginning of this restricted interval.
4. chemotherapy drug i becomes inconsistent with the cancer diagnosis is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if chemotherapy drug i becomes inconsistent with the cancer diagnosis does occur, administer chemotherapy is not required to occur subsequently and if it does not occur subsequently, then the behavior is not required to hold for the remainder of the event sequence.
5. There might be many restricted intervals in the event sequence. If chemotherapy drug i becomes inconsistent with the cancer diagnosis occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of chemotherapy drug i becomes inconsistent with the cancer diagnosis, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a potential new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

**BEHAVIOR:**
1. The event of primary interest in this behavior is resolve inconsistency between chemotherapy drug i and the cancer diagnosis.
2. There are no events of secondary interest in this behavior.
3. resolve inconsistency between chemotherapy drug i and the cancer diagnosis is required to occur at least once.

FSA and DNL for Property Chemo.D.9b

441
Event alphabet:

- A: create a new dose of pre-chemotherapy supportive care medication
- B: administer pre-chemotherapy supportive care medications
- C: pre-chemotherapy supportive care medication becomes physically unsuitable for administration

**SCOPE:**
1. From the start of any event sequence through to the end of that event sequence, the behavior must hold.

**BEHAVIOR:**
1. The events of primary interest in this behavior are create a new dose of pre-chemotherapy supportive care medication and administer pre-chemotherapy supportive care medications.
2. The event of secondary interest in this behavior is pre-chemotherapy supportive care medication becomes physically unsuitable for administration.
3. administer pre-chemotherapy supportive care medications is not allowed to occur until after create a new dose of pre-chemotherapy supportive care medication occurs.
4. Before the first create a new dose of pre-chemotherapy supportive care medication occurs, pre-chemotherapy supportive care medication becomes physically unsuitable for administration is allowed to occur zero or more times.
5. create a new dose of pre-chemotherapy supportive care medication is not required to occur.
6. Even if create a new dose of pre-chemotherapy supportive care medication occurs, administer pre-chemotherapy supportive care medications is not required to occur after create a new dose of pre-chemotherapy supportive care medication occurs.
7. After create a new dose of pre-chemotherapy supportive care medication occurs, but before the first subsequent administer pre-chemotherapy supportive care medications occurs:
   - create a new dose of pre-chemotherapy supportive care medication is not allowed to occur again;
   - pre-chemotherapy supportive care medication becomes physically unsuitable for administration is allowed to occur zero or more times.
8. After create a new dose of pre-chemotherapy supportive care medication and the first subsequent administer pre-chemotherapy supportive care medications occur:
   - pre-chemotherapy supportive care medication becomes physically unsuitable for administration is allowed to occur zero or more times;
   - administer pre-chemotherapy supportive care medications is not allowed to occur again until after another create a new dose of pre-chemotherapy supportive care medication occurs; create a new dose of pre-chemotherapy supportive care medication is allowed to occur again and, if it does, then the situation is the same as when the first create a new dose of pre-chemotherapy supportive care medication occurred, meaning that the restrictions described in parts 6, 7, and 8 would again apply.

FSA and DNL for Property Chemo.D.14a
Event alphabet:

- \textit{A}: create a new dose of pre-chemotherapy supportive care medication i
- \textit{START}: pre-chemotherapy supportive care medication i becomes physically unsuitable for administration
- \textit{END}: administer pre-chemotherapy supportive care medications

\textbf{SCOPE:}

1. A restricted interval in the event sequence can have both a starting delimiter, pre-chemotherapy supportive care medication i becomes physically unsuitable for administration, and an ending delimiter, administer pre-chemotherapy supportive care medications.
2. The behavior is required to hold from an occurrence of pre-chemotherapy supportive care medication i becomes physically unsuitable for administration, if it ever occurs, through to the first subsequent occurrence of administer pre-chemotherapy supportive care medications, if it ever occurs.
3. If there are multiple occurrences of pre-chemotherapy supportive care medication i becomes physically unsuitable for administration without an occurrence of administer pre-chemotherapy supportive care medications in between them, only the last of those occurrences of pre-chemotherapy supportive care medication i becomes physically unsuitable for administration potentially starts a restricted interval; each of those occurrences of pre-chemotherapy supportive care medication i becomes physically unsuitable for administration resets the beginning of this restricted interval.
4. pre-chemotherapy supportive care medication i becomes physically unsuitable for administration is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if pre-chemotherapy supportive care medication i becomes physically unsuitable for administration does occur, administer pre-chemotherapy supportive care medications is not required to occur subsequently and if it does not occur subsequently, then the behavior is not required to hold for the remainder of the event sequence.
5. There might be many restricted intervals in the event sequence. If pre-chemotherapy supportive care medication i becomes physically unsuitable for administration occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of pre-chemotherapy supportive care medication i becomes physically unsuitable for administration, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a potential new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

\textbf{BEHAVIOR:}

1. The event of primary interest in this behavior is create a new dose of pre-chemotherapy supportive care medication i.
2. There are no events of secondary interest in this behavior.
3. create a new dose of pre-chemotherapy supportive care medication i is required to occur exactly once.

\textbf{FSA and DNL for Property Chemo.D.14b}
SCOPE:
1. From the start of any event sequence through to the end of that event sequence, the behavior must hold.

BEHAVIOR:
1. The events of primary interest in this behavior are make sure pre-chemotherapy supportive care medications are consistent with chemotherapy drugs and administer pre-chemotherapy supportive care medications.
2. There are no events of secondary interest in this behavior.
3. administer pre-chemotherapy supportive care medications is not allowed to occur until after make sure pre-chemotherapy supportive care medications are consistent with chemotherapy drugs occurs.
4. make sure pre-chemotherapy supportive care medications are consistent with chemotherapy drugs is not required to occur.
5. Even if make sure pre-chemotherapy supportive care medications are consistent with chemotherapy drugs does occur, administer pre-chemotherapy supportive care medications is not required to occur after make sure pre-chemotherapy supportive care medications are consistent with chemotherapy drugs occurs.
6. After make sure pre-chemotherapy supportive care medications are consistent with chemotherapy drugs occurs, but before the first subsequent administer pre-chemotherapy supportive care medications occurs, make sure pre-chemotherapy supportive care medications are consistent with chemotherapy drugs is allowed to occur again, zero or more times.
7. After make sure pre-chemotherapy supportive care medications are consistent with chemotherapy drugs and the first subsequent administer pre-chemotherapy supportive care medications occur:
   • administer pre-chemotherapy supportive care medications is not allowed to occur again until after another make sure pre-chemotherapy supportive care medications are consistent with chemotherapy drugs occurs; make sure pre-chemotherapy supportive care medications are consistent with chemotherapy drugs is allowed to occur again and, if it does, then the situation is the same as when the first make sure pre-chemotherapy supportive care medications are consistent with chemotherapy drugs occurred, meaning that the restrictions described in parts 5, 6, and 7 would again apply.

FSA and DNL for Property Chemo.D.15a
SCOPE:
1. A restricted interval in the event sequence can have both a starting delimiter, pre-chemotherapy supportive care medication i becomes inconsistent with chemotherapy drugs, and an ending delimiter, administer pre-chemotherapy supportive care medications.
2. The behavior is required to hold from an occurrence of pre-chemotherapy supportive care medication i becomes inconsistent with chemotherapy drugs, if it ever occurs, through to the first subsequent occurrence of administer pre-chemotherapy supportive care medications, if it ever occurs.
3. If there are multiple occurrences of pre-chemotherapy supportive care medication i becomes inconsistent with chemotherapy drugs without an occurrence of administer pre-chemotherapy supportive care medications in between them, only the last of those occurrences of pre-chemotherapy supportive care medication i becomes inconsistent with chemotherapy drugs potentially starts a restricted interval; each of those occurrences of pre-chemotherapy supportive care medication i becomes inconsistent with chemotherapy drugs resets the beginning of this restricted interval.
4. pre-chemotherapy supportive care medication i becomes inconsistent with chemotherapy drugs is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if pre-chemotherapy supportive care medication i becomes inconsistent with chemotherapy drugs does occur, administer pre-chemotherapy supportive care medications is not required to occur subsequently and if it does not occur subsequently, then the behavior is not required to hold for the remainder of the event sequence.
5. There might be many restricted intervals in the event sequence. If pre-chemotherapy supportive care medication i becomes inconsistent with chemotherapy drugs occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of pre-chemotherapy supportive care medication i becomes inconsistent with chemotherapy drugs, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a potential new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

BEHAVIOR:
1. The event of primary interest in this behavior is resolve inconsistency between pre-chemotherapy supportive care medication i and chemotherapy drugs.
2. There are no events of secondary interest in this behavior.
3. resolve inconsistency between pre-chemotherapy supportive care medication i and chemotherapy drugs is required to occur at least once.

FSA and DNL for Property Chemo.D.15b
Event alphabet:
- A: make sure pre-chemotherapy supportive care medications are consistent with the chemotherapy orders
- B: administer pre-chemotherapy supportive care medications

**SCOPE:**
1. From the start of any event sequence through to the end of that event sequence, the behavior must hold.

**BEHAVIOR:**
1. The events of primary interest in this behavior are make sure pre-chemotherapy supportive care medications are consistent with the chemotherapy orders and administer pre-chemotherapy supportive care medications.
2. There are no events of secondary interest in this behavior.
3. administer pre-chemotherapy supportive care medications is not allowed to occur until after make sure pre-chemotherapy supportive care medications are consistent with the chemotherapy orders occurs.
4. make sure pre-chemotherapy supportive care medications are consistent with the chemotherapy orders is not required to occur.
5. Even if make sure pre-chemotherapy supportive care medications are consistent with the chemotherapy orders does occur, administer pre-chemotherapy supportive care medications is not required to occur after make sure pre-chemotherapy supportive care medications are consistent with the chemotherapy orders occurs.
6. After make sure pre-chemotherapy supportive care medications are consistent with the chemotherapy orders occurs, but before the first subsequent administer pre-chemotherapy supportive care medications occurs, make sure pre-chemotherapy supportive care medications are consistent with the chemotherapy orders is allowed to occur again, zero or more times.
7. After make sure pre-chemotherapy supportive care medications are consistent with the chemotherapy orders and the first subsequent administer pre-chemotherapy supportive care medications occur:
   - administer pre-chemotherapy supportive care medications is not allowed to occur again until after another make sure pre-chemotherapy supportive care medications are consistent with the chemotherapy orders occurs; make sure pre-chemotherapy supportive care medications are consistent with the chemotherapy orders is allowed to occur again and, if it does, then the situation is the same as when the first make sure pre-chemotherapy supportive care medications are consistent with the chemotherapy orders occurred, meaning that the restrictions described in parts 5, 6, and 7 would again apply.

FSA and DNL for Property Chemo.D.16a
Event alphabet:
- A: resolve that mismatch between pre-chemotherapy supportive care medication i and the chemotherapy orders
- START: pre-chemotherapy supportive care medication i differs from the chemotherapy orders
- END: administer pre-chemotherapy supportive care medications

SCOPE:
1. A restricted interval in the event sequence can have both a starting delimiter, pre-chemotherapy supportive care medication i differs from the chemotherapy orders, and an ending delimiter, administer pre-chemotherapy supportive care medications.
2. The behavior is required to hold from an occurrence of pre-chemotherapy supportive care medication i differs from the chemotherapy orders, if it ever occurs, through to the first subsequent occurrence of administer pre-chemotherapy supportive care medications, if it ever occurs.
3. If there are multiple occurrences of pre-chemotherapy supportive care medication i differs from the chemotherapy orders without an occurrence of administer pre-chemotherapy supportive care medications in between them, only the last of those occurrences of pre-chemotherapy supportive care medication i differs from the chemotherapy orders potentially starts a restricted interval; each of those occurrences of pre-chemotherapy supportive care medication i differs from the chemotherapy orders resets the beginning of this restricted interval.
4. pre-chemotherapy supportive care medication i differs from the chemotherapy orders is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if pre-chemotherapy supportive care medication i differs from the chemotherapy orders does occur, administer pre-chemotherapy supportive care medications is not required to occur subsequently and if it does not occur subsequently, then the behavior is not required to hold for the remainder of the event sequence.
5. There might be many restricted intervals in the event sequence. If pre-chemotherapy supportive care medication i differs from the chemotherapy orders occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of pre-chemotherapy supportive care medication i differs from the chemotherapy orders, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a potential new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

BEHAVIOR:
1. The event of primary interest in this behavior is resolve that mismatch between pre-chemotherapy supportive care medication i and the chemotherapy orders.
2. There are no events of secondary interest in this behavior.
3. resolve that mismatch between pre-chemotherapy supportive care medication i and the chemotherapy orders is required to occur at least once.

FSA and DNL for Property Chemo.D.16b
**Event alphabet:**
- A: make sure pre-chemotherapy supportive care medications are in doses that are consistent with Patient data
- B: administer pre-chemotherapy supportive care medications

**SCOPE:**
1. From the start of any event sequence through to the end of that event sequence, the behavior must hold.

**BEHAVIOR:**
1. The events of primary interest in this behavior are **make sure pre-chemotherapy supportive care medications are in doses that are consistent with Patient data** and **administer pre-chemotherapy supportive care medications**.
2. There are no events of secondary interest in this behavior.
3. **administer pre-chemotherapy supportive care medications** is not allowed to occur until after **make sure pre-chemotherapy supportive care medications are in doses that are consistent with Patient data** occurs.
4. **make sure pre-chemotherapy supportive care medications are in doses that are consistent with Patient data** is not required to occur.
5. Even if **make sure pre-chemotherapy supportive care medications are in doses that are consistent with Patient data** does occur, **administer pre-chemotherapy supportive care medications** is not required to occur after **make sure pre-chemotherapy supportive care medications are in doses that are consistent with Patient data** occurs.
6. After **make sure pre-chemotherapy supportive care medications are in doses that are consistent with Patient data** occurs, but before the first subsequent **administer pre-chemotherapy supportive care medications** occurs, **make sure pre-chemotherapy supportive care medications are in doses that are consistent with Patient data** is allowed to occur again, zero or more times.
7. After **make sure pre-chemotherapy supportive care medications are in doses that are consistent with Patient data** and the first subsequent **administer pre-chemotherapy supportive care medications** occur:
   - **administer pre-chemotherapy supportive care medications** is not allowed to occur again until after another **make sure pre-chemotherapy supportive care medications are in doses that are consistent with Patient data** occurs; **make sure pre-chemotherapy supportive care medications are in doses that are consistent with Patient data** is allowed to occur again and, if it does, then the situation is the same as when the first **make sure pre-chemotherapy supportive care medications are in doses that are consistent with Patient data** occurred, meaning that the restrictions described in parts 5, 6, and 7 would again apply.

FSA and DNL for Property Chemo.D.17a
Event alphabet:
- A: resolve inconsistency between pre-chemotherapy supportive care medication i and Patient data
- START: pre-chemotherapy supportive care medication dose i becomes inconsistent with Patient data
- END: administer pre-chemotherapy supportive care medications

SCOPE:
1. A restricted interval in the event sequence can have both a starting delimiter, \text{pre-chemotherapy supportive care medication dose i becomes inconsistent with Patient data}, and an ending delimiter, \text{administer pre-chemotherapy supportive care medications}, if it ever occurs, through to the first subsequent occurrence of \text{administer pre-chemotherapy supportive care medications}, if it ever occurs.
2. The behavior is required to hold from an occurrence of \text{pre-chemotherapy supportive care medication dose i becomes inconsistent with Patient data}, if it ever occurs, through to the first subsequent occurrence of \text{administer pre-chemotherapy supportive care medications}, if it ever occurs.
3. If there are multiple occurrences of \text{pre-chemotherapy supportive care medication dose i becomes inconsistent with Patient data} without an occurrence of \text{administer pre-chemotherapy supportive care medications} in between them, only the last of those occurrences of \text{pre-chemotherapy supportive care medication dose i becomes inconsistent with Patient data} potentially starts a restricted interval; each of those occurrences of \text{pre-chemotherapy supportive care medication dose i becomes inconsistent with Patient data} resets the beginning of this restricted interval.
4. \text{pre-chemotherapy supportive care medication dose i becomes inconsistent with Patient data} is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if \text{pre-chemotherapy supportive care medication dose i becomes inconsistent with Patient data} does occur, \text{administer pre-chemotherapy supportive care medications} is not required to occur subsequently and if it does not occur subsequently, then the behavior is not required to hold for the remainder of the event sequence.
5. There might be many restricted intervals in the event sequence. If \text{pre-chemotherapy supportive care medication dose i becomes inconsistent with Patient data} occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of \text{pre-chemotherapy supportive care medication dose i becomes inconsistent with Patient data}, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a potential new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

BEHAVIOR:
1. The event of primary interest in this behavior is \text{resolve inconsistency between pre-chemotherapy supportive care medication i and Patient data}.
2. There are no events of secondary interest in this behavior.
3. \text{resolve inconsistency between pre-chemotherapy supportive care medication i and Patient data} is required to occur at least once.

FSA and DNL for Property Chemo.D.17b
Event alphabet:

- A: correctly prepare a new dose of pre-chemotherapy supportive care medication i
- B: administer pre-chemotherapy supportive care medications
- C: incorrectly prepare a dose of pre-chemotherapy supportive care medication i

SCOPE:
1. From the start of any event sequence through to the end of that event sequence, the behavior must hold.

BEHAVIOR:
1. The events of primary interest in this behavior are correctly prepare a new dose of pre-chemotherapy supportive care medication i and administer pre-chemotherapy supportive care medications.
2. The event of secondary interest in this behavior is incorrectly prepare a dose of pre-chemotherapy supportive care medication i.
3. adminster pre-chemotherapy supportive care medications is not allowed to occur until after correctly prepare a new dose of pre-chemotherapy supportive care medication i occurs.
4. Before the first correctly prepare a new dose of pre-chemotherapy supportive care medication i occurs, incorrectly prepare a dose of pre-chemotherapy supportive care medication i is allowed to occur zero or more times.
5. The event of secondary interest in this behavior is not allowed to occur on a second occasion when it is the primary interest event.
6. Even if correctly prepare a new dose of pre-chemotherapy supportive care medication i occurs, administer pre-chemotherapy supportive care medications is not required to occur after correctly prepare a new dose of pre-chemotherapy supportive care medication i occurs.
7. After correctly prepare a new dose of pre-chemotherapy supportive care medication i occurs, but before the first subsequent administer pre-chemotherapy supportive care medications occurs:
   - correctly prepare a new dose of pre-chemotherapy supportive care medication i is not allowed to occur again;
   - incorrectly prepare a dose of pre-chemotherapy supportive care medication i is allowed to occur zero or more times.
8. After correctly prepare a new dose of pre-chemotherapy supportive care medication i and the first subsequent administer pre-chemotherapy supportive care medications occur:
   - incorrectly prepare a dose of pre-chemotherapy supportive care medication i is allowed to occur zero or more times;
   - administer pre-chemotherapy supportive care medications is not allowed to occur again until after another correctly prepare a new dose of pre-chemotherapy supportive care medication i occurs; correctly prepare a new dose of pre-chemotherapy supportive care medication i is allowed to occur again and, if it does, then the situation is the same as when the first correctly prepare a new dose of pre-chemotherapy supportive care medication i occurred, meaning that the restrictions described in parts 6, 7, and 8 would again apply.

FSA and DNL for Property Chemo.D.18a
**Event alphabet:**
- **A**: correctly prepare a new dose of pre-chemotherapy supportive care medication i
- **START**: incorrectly prepare a dose of pre-chemotherapy supportive care medication i
- **END**: administer pre-chemotherapy supportive care medications

**Scope:**
1. A restricted interval in the event sequence can have both a starting delimiter, incorrectly prepare a dose of pre-chemotherapy supportive care medication i, and an ending delimiter, administer pre-chemotherapy supportive care medications.
2. The behavior is required to hold from an occurrence of incorrectly prepare a dose of pre-chemotherapy supportive care medication i, if it ever occurs, through to the first subsequent occurrence of administer pre-chemotherapy supportive care medications, if it ever occurs.
3. If there are multiple occurrences of incorrectly prepare a dose of pre-chemotherapy supportive care medication i without an occurrence of administer pre-chemotherapy supportive care medications in between them, only the last of those occurrences of incorrectly prepare a dose of pre-chemotherapy supportive care medication i potentially starts a restricted interval; each of those occurrences of incorrectly prepare a dose of pre-chemotherapy supportive care medication i resets the beginning of this restricted interval.
4. Incorrectly prepare a dose of pre-chemotherapy supportive care medication i is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if incorrectly prepare a dose of pre-chemotherapy supportive care medication i does occur, administer pre-chemotherapy supportive care medications is not required to occur subsequently and if it does not occur subsequently, then the behavior is not required to hold for the remainder of the event sequence.
5. There might be many restricted intervals in the event sequence. If incorrectly prepare a dose of pre-chemotherapy supportive care medication i occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of incorrectly prepare a dose of pre-chemotherapy supportive care medication i, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a potential new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

**Behavior:**
1. The event of primary interest in this behavior is correctly prepare a new dose of pre-chemotherapy supportive care medication i.
2. There are no events of secondary interest in this behavior.
3. Correctly prepare a new dose of pre-chemotherapy supportive care medication i is required to occur exactly once.

FSA and DNL for Property Chemo.D.18b
Event alphabet:
- A: assign pre-chemotherapy supportive care medications to Patient
- B: administer pre-chemotherapy supportive care medications
- C: remove assignment of pre-chemotherapy supportive care medications from Patient

**SCOPE:**
1. From the start of any event sequence through to the end of that event sequence, the behavior must hold.

**BEHAVIOR:**
1. The events of primary interest in this behavior are assign pre-chemotherapy supportive care medications to Patient and administer pre-chemotherapy supportive care medications.
2. The event of secondary interest in this behavior is remove assignment of pre-chemotherapy supportive care medications from Patient.
3. administer pre-chemotherapy supportive care medications is not allowed to occur until after assign pre-chemotherapy supportive care medications to Patient occurs.
4. Before the first assign pre-chemotherapy supportive care medications to Patient occurs, remove assignment of pre-chemotherapy supportive care medications from Patient is allowed to occur zero or more times.
5. assign pre-chemotherapy supportive care medications to Patient is not required to occur.
6. Even if assign pre-chemotherapy supportive care medications to Patient does occur, administer pre-chemotherapy supportive care medications is not required to occur after assign pre-chemotherapy supportive care medications to Patient occurs.
7. After assign pre-chemotherapy supportive care medications to Patient occurs, but before the first subsequent administer pre-chemotherapy supportive care medications occurs:
   - assign pre-chemotherapy supportive care medications to Patient is allowed to occur again, zero or more times;
   - remove assignment of pre-chemotherapy supportive care medications from Patient is allowed to occur zero or more times.
8. After assign pre-chemotherapy supportive care medications to Patient and the first subsequent administer pre-chemotherapy supportive care medications occur:
   - remove assignment of pre-chemotherapy supportive care medications from Patient is allowed to occur zero or more times;
   - administer pre-chemotherapy supportive care medications is not allowed to occur again until after another assign pre-chemotherapy supportive care medications to Patient occurs; assign pre-chemotherapy supportive care medications to Patient is allowed to occur again and, if it does, then the situation is the same as when the first assign pre-chemotherapy supportive care medications to Patient occurred, meaning that the restrictions described in parts 6, 7, and 8 would again apply.

FSA and DNL for Property Chemo.D.19a
Event alphabet:
- A: assign pre-chemotherapy supportive care medications to Patient
- START: remove assignment of pre-chemotherapy supportive care medications from Patient
- END: administer pre-chemotherapy supportive care medications

**SCOPE:**
1. A restricted interval in the event sequence can have both a starting delimiter, **remove assignment of pre-chemotherapy supportive care medications from Patient**, and an ending delimiter, **administer pre-chemotherapy supportive care medications**.
2. The behavior is required to hold from an occurrence of **remove assignment of pre-chemotherapy supportive care medications from Patient**, if it ever occurs, through to the first subsequent occurrence of **administer pre-chemotherapy supportive care medications**, if it ever occurs.
3. If there are multiple occurrences of **remove assignment of pre-chemotherapy supportive care medications from Patient** without an occurrence of **administer pre-chemotherapy supportive care medications** in between them, only the last of those occurrences of **remove assignment of pre-chemotherapy supportive care medications from Patient** potentially starts a restricted interval; each of those occurrences of **remove assignment of pre-chemotherapy supportive care medications from Patient** resets the beginning of this restricted interval.
4. **remove assignment of pre-chemotherapy supportive care medications from Patient** is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if **remove assignment of pre-chemotherapy supportive care medications from Patient** does occur, **administer pre-chemotherapy supportive care medications** is not required to occur subsequently and if it does not occur subsequently, then the behavior is not required to hold for the remainder of the event sequence.
5. There might be many restricted intervals in the event sequence. If **remove assignment of pre-chemotherapy supportive care medications from Patient** occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of **remove assignment of pre-chemotherapy supportive care medications from Patient**, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a potential new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

**BEHAVIOR:**
1. The event of primary interest in this behavior is **assign pre-chemotherapy supportive care medications to Patient**.
2. There are no events of secondary interest in this behavior.
3. **assign pre-chemotherapy supportive care medications to Patient** is required to occur exactly once.

**FSA and DNL for Property Chemo.D.19b**
**Event alphabet:**
- A: resolve stale or disparate Patient data
- START: Patient data becomes stale or disparate
- END: prepare chemotherapy drugs

**SCOPE:**
1. A restricted interval in the event sequence can have both a starting delimiter, Patient data becomes stale or disparate, and an ending delimiter, prepare chemotherapy drugs.
2. The behavior is required to hold from an occurrence of Patient data becomes stale or disparate, if it ever occurs, through to the first subsequent occurrence of prepare chemotherapy drugs, if it ever occurs.
3. If there are multiple occurrences of Patient data becomes stale or disparate without an occurrence of prepare chemotherapy drugs in between them, only the last of those occurrences of Patient data becomes stale or disparate potentially starts a restricted interval; each of those occurrences of Patient data becomes stale or disparate resets the beginning of this restricted interval.
4. Patient data becomes stale or disparate is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if Patient data becomes stale or disparate does occur, prepare chemotherapy drugs is not required to occur subsequently and if it does not occur subsequently, then the behavior is not required to hold for the remainder of the event sequence.
5. There might be many restricted intervals in the event sequence. If Patient data becomes stale or disparate occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of Patient data becomes stale or disparate, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a potential new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

**BEHAVIOR:**
1. The event of primary interest in this behavior is resolve stale or disparate Patient data.
2. There are no events of secondary interest in this behavior.
3. resolve stale or disparate Patient data is required to occur at least once.

FSA and DNL for Property Chemo.D.5
Event alphabet:
- A: resolve problematic Patient data
- START: Patient data becomes problematic
- END: administer chemotherapy

**SCOPE:**
1. A restricted interval in the event sequence can have both a starting delimiter, Patient data becomes problematic, and an ending delimiter, administer chemotherapy.
2. The behavior is required to hold from an occurrence of Patient data becomes problematic, if it ever occurs, through to the first subsequent occurrence of administer chemotherapy, if it ever occurs.
3. If there are multiple occurrences of Patient data becomes problematic without an occurrence of administer chemotherapy in between them, only the last of those occurrences of Patient data becomes problematic potentially starts a restricted interval; each of those occurrences of Patient data becomes problematic resets the beginning of this restricted interval.
4. Patient data becomes problematic is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if Patient data becomes problematic does occur, administer chemotherapy is not required to occur subsequently and if it does not occur subsequently, then the behavior is not required to hold for the remainder of the event sequence.
5. There might be many restricted intervals in the event sequence. If Patient data becomes problematic occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of Patient data becomes problematic, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a potential new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

**BEHAVIOR:**
1. The event of primary interest in this behavior is resolve problematic Patient data.
2. There are no events of secondary interest in this behavior.
3. resolve problematic Patient data is required to occur at least once.

FSA and DNL for Property Chemo.D.6
Event alphabet:
- A: resolve problematic Patient data
- START: Patient data becomes problematic
- END: administer pre-chemotherapy supportive care medications

SCOPE:
1. A restricted interval in the event sequence can have both a starting delimiter, Patient data becomes problematic, and an ending delimiter, administer pre-chemotherapy supportive care medications.
2. The behavior is required to hold from an occurrence of Patient data becomes problematic, if it ever occurs, through to the first subsequent occurrence of administer pre-chemotherapy supportive care medications, if it ever occurs.
3. If there are multiple occurrences of Patient data becomes problematic without an occurrence of administer pre-chemotherapy supportive care medications in between them, each of those occurrences of Patient data becomes problematic resets the beginning of this restricted interval.
4. Patient data becomes problematic is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if Patient data becomes problematic does occur, administer pre-chemotherapy supportive care medications is not required to occur subsequently and if it does not occur subsequently, then the behavior is not required to hold for the remainder of the event sequence.
5. There might be many restricted intervals in the event sequence. If Patient data becomes problematic occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of Patient data becomes problematic, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a potential new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

BEHAVIOR:
1. The event of primary interest in this behavior is resolve problematic Patient data.
2. There are no events of secondary interest in this behavior.
3. resolve problematic Patient data is required to occur at least once.

FSA and DNL for Property Chemo.D.7
**Event alphabet:**
- A: Patient has adverse reaction to administration of pre-chemotherapy supportive care medications
- B: stop administration of pre-chemotherapy supportive care medications
- C: Attending MD or Attending MD’s delegate decides Patient disposition, stabilize Patient condition

**SCOPE:**
1. From the start of any event sequence through to the end of that event sequence, the behavior must hold.

**BEHAVIOR:**
1. The events of primary interest in this behavior are Patient has adverse reaction to administration of pre-chemotherapy supportive care medications and stop administration of pre-chemotherapy supportive care medications.
2. The events of secondary interest in this behavior are Attending MD or Attending MD’s delegate decides Patient disposition and stabilize Patient condition.
3. If Patient has adverse reaction to administration of pre-chemotherapy supportive care medications occurs, stop administration of pre-chemotherapy supportive care medications is required to occur subsequently.
4. Before the first Patient has adverse reaction to administration of pre-chemotherapy supportive care medications occurs:
   - stop administration of pre-chemotherapy supportive care medications is allowed to occur zero or more times;
   - All the events of secondary interest are allowed to occur zero or more times.
5. Patient has adverse reaction to administration of pre-chemotherapy supportive care medications is not required to occur.
6. After Patient has adverse reaction to administration of pre-chemotherapy supportive care medications occurs, but before the first subsequent stop administration of pre-chemotherapy supportive care medications occurs:
   - Patient has adverse reaction to administration of pre-chemotherapy supportive care medications is not allowed to occur again;
   - None of the events of secondary interest are allowed to occur.
7. After Patient has adverse reaction to administration of pre-chemotherapy supportive care medications and the first subsequent stop administration of pre-chemotherapy supportive care medications occur:
   - All the events of secondary interest are allowed to occur zero or more times;
   - stop administration of pre-chemotherapy supportive care medications is allowed to occur again, zero or more times, before another Patient has adverse reaction to administration of pre-chemotherapy supportive care medications occurs. Patient has adverse reaction to administration of pre-chemotherapy supportive care medications is allowed to occur again and, if it does, then the situation is the same as when the first Patient has adverse reaction to administration of pre-chemotherapy supportive care medications occurred, meaning that the restrictions described in parts 3, 6, and 7 would again apply.

**FSA and DNL for Property Chemo.D.21**

457
**Event alphabet:**
- A: Patient has adverse reaction to administration of pre-chemotherapy supportive care medications
- B: stabilize Patient condition
- C: Attending MD or Attending MD’s delegate decides Patient disposition

**SCOPE:**
1. From the start of any event sequence through to the end of that event sequence, the behavior must hold.

**BEHAVIOR:**
1. The events of primary interest in this behavior are *Patient has adverse reaction to administration of pre-chemotherapy supportive care medications* and *stabilize Patient condition*.
2. The event of secondary interest in this behavior is *Attending MD or Attending MD’s delegate decides Patient disposition*.
3. If *Patient has adverse reaction to administration of pre-chemotherapy supportive care medications* occurs, *stabilize Patient condition* is required to occur subsequently.
4. Before the first *Patient has adverse reaction to administration of pre-chemotherapy supportive care medications* occurs:
   - *stabilize Patient condition* is allowed to occur zero or more times;
   - *Attending MD or Attending MD’s delegate decides Patient disposition* is allowed to occur zero or more times.
5. *Patient has adverse reaction to administration of pre-chemotherapy supportive care medications* is not required to occur.
6. After *Patient has adverse reaction to administration of pre-chemotherapy supportive care medications* occurs, but before the first subsequent *stabilize Patient condition* occurs:
   - *Patient has adverse reaction to administration of pre-chemotherapy supportive care medications* is not allowed to occur again;
   - *Attending MD or Attending MD’s delegate decides Patient disposition* is not allowed to occur.
7. After *Patient has adverse reaction to administration of pre-chemotherapy supportive care medications* and the first subsequent *stabilize Patient condition* occurs:
   - *Attending MD or Attending MD’s delegate decides Patient disposition* is allowed to occur zero or more times;
   - *stabilize Patient condition* is allowed to occur again, zero or more times, before another *Patient has adverse reaction to administration of pre-chemotherapy supportive care medications* occurs; *Patient has adverse reaction to administration of pre-chemotherapy supportive care medications* is allowed to occur again and, if it does, then the situation is the same as when the first *Patient has adverse reaction to administration of pre-chemotherapy supportive care medications* occurred, meaning that the restrictions described in parts 3, 6, and 7 would again apply.

**FSA and DNL for Property Chemo.D.22**
Event alphabet:
1. A: Patient has adverse reaction to administration of pre-chemotherapy supportive care medications
2. B: Attending MD or Attending MD’s delegate decides Patient disposition

**SCOPE:**
1. From the start of any event sequence through to the end of that event sequence, the behavior must hold.

**BEHAVIOR:**
1. The events of primary interest in this behavior are **Patient has adverse reaction to administration of pre-chemotherapy supportive care medications** and **Attending MD or Attending MD's delegate decides Patient disposition**.
2. There are no events of secondary interest in this behavior.
3. If **Patient has adverse reaction to administration of pre-chemotherapy supportive care medications** occurs, **Attending MD or Attending MD’s delegate decides Patient disposition** is required to occur subsequently.
4. Before the first **Patient has adverse reaction to administration of pre-chemotherapy supportive care medications** occurs, **Attending MD or Attending MD’s delegate decides Patient disposition** is allowed to occur zero or more times.
5. **Patient has adverse reaction to administration of pre-chemotherapy supportive care medications** is not required to occur.
6. After **Patient has adverse reaction to administration of pre-chemotherapy supportive care medications** occurs, but before the first subsequent **Attending MD or Attending MD’s delegate decides Patient disposition** occurs, **Patient has adverse reaction to administration of pre-chemotherapy supportive care medications** is not allowed to occur again.
7. After **Patient has adverse reaction to administration of pre-chemotherapy supportive care medications** and the first subsequent **Attending MD or Attending MD’s delegate decides Patient disposition** occurs:
   - **Attending MD or Attending MD’s delegate decides Patient disposition** is allowed to occur again, zero or more times, before another **Patient has adverse reaction to administration of pre-chemotherapy supportive care medications** occurs. **Patient has adverse reaction to administration of pre-chemotherapy supportive care medications** is allowed to occur again and, if it does, then the situation is the same as when the first **Patient has adverse reaction to administration of pre-chemotherapy supportive care medications** occurred, meaning that the restrictions described in parts 3, 6, and 7 would again apply.

FSA and DNL for Property Chemo.D.23
Event alphabet:
- A: stop administration of pre-chemotherapy supportive care medication
- B: stabilize Patient condition
- START: Patient has adverse reaction to administration of pre-chemotherapy supportive care medication
- END: discharge Patient
- C: Attending MD or Attending MD’s delegate decides Patient disposition

SCOPE:
1. A restricted interval in the event sequence can have both a starting delimiter, Patient has adverse reaction to administration of pre-chemotherapy supportive care medication, and an ending delimiter, discharge Patient.
2. The behavior is required to hold from an occurrence of Patient has adverse reaction to administration of pre-chemotherapy supportive care medication, if it ever occurs, through to the first subsequent occurrence of discharge Patient, if it ever occurs.
3. If there are multiple occurrences of Patient has adverse reaction to administration of pre-chemotherapy supportive care medication without an occurrence of discharge Patient in between them, only the last of those occurrences of Patient has adverse reaction to administration of pre-chemotherapy supportive care medication starts a restricted interval; each of those occurrences of Patient has adverse reaction to administration of pre-chemotherapy supportive care medication resets the beginning of this restricted interval.
4. Patient has adverse reaction to administration of pre-chemotherapy supportive care medication is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if Patient has adverse reaction to administration of pre-chemotherapy supportive care medication does occur, discharge Patient is not required to occur subsequently. Even if discharge Patient does not occur subsequently, the behavior is still required to hold, until the end of the event sequence.
5. There might be many restricted intervals in the event sequence. If Patient has adverse reaction to administration of pre-chemotherapy supportive care medication occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of Patient has adverse reaction to administration of pre-chemotherapy supportive care medication, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

BEHAVIOR:
1. The events of primary interest in this behavior are stop administration of pre-chemotherapy supportive care medication and stabilize Patient condition.
2. The event of secondary interest in this behavior is Attending MD or Attending MD’s delegate decides Patient disposition.
3. stabilize Patient condition is not allowed to occur until after stop administration of pre-chemotherapy supportive care medication occurs.
4. Before the first stop administration of pre-chemotherapy supportive care medication occurs, Attending MD or Attending MD’s delegate decides Patient disposition is allowed to occur zero or more times.
5. stop administration of pre-chemotherapy supportive care medication is required to occur, whether or not stabilize Patient condition eventually occurs.
6. stabilize Patient condition is not required to occur after stop administration of pre-chemotherapy supportive care medication occurs.
7. After stop administration of pre-chemotherapy supportive care medication occurs, but before the first subsequent stabilize Patient condition occurs:
   - stop administration of pre-chemotherapy supportive care medication is not allowed to occur again;
   - Attending MD or Attending MD’s delegate decides Patient disposition is not allowed to occur.
8. After stop administration of pre-chemotherapy supportive care medication and the first subsequent stabilize Patient condition occur:
   - Attending MD or Attending MD’s delegate decides Patient disposition is allowed to occur zero or more times;
   - Both stop administration of pre-chemotherapy supportive care medication and stabilize Patient condition are allowed to occur again, zero or more times, and there are no restrictions on the occurrences of any future events.

FSA and DNL for Property Chemo.D.24a (D.21 → D.22)
**Event alphabet:**
- **A**: stabilize Patient condition
- **B**: Attending MD or Attending MD’s delegate decides Patient disposition
- **START**: Patient has adverse reaction to administration of pre-chemotherapy supportive care medication
- **END**: discharge Patient
- **C**: stop administration of pre-chemotherapy supportive care medication

**SCOPE:**
1. A restricted interval in the event sequence can have both a starting delimiter, Patient has adverse reaction to administration of pre-chemotherapy supportive care medication, and an ending delimiter, discharge Patient.
2. The behavior is required to hold from an occurrence of Patient has adverse reaction to administration of pre-chemotherapy supportive care medication, if it ever occurs, through to the first subsequent occurrence of discharge Patient, if it ever occurs.
3. If there are multiple occurrences of Patient has adverse reaction to administration of pre-chemotherapy supportive care medication without an occurrence of discharge Patient in between them, only the last of those occurrences of Patient has adverse reaction to administration of pre-chemotherapy supportive care medication starts a restricted interval; each of those occurrences of Patient has adverse reaction to administration of pre-chemotherapy supportive care medication resets the beginning of this restricted interval.
4. Patient has adverse reaction to administration of pre-chemotherapy supportive care medication is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if Patient has adverse reaction to administration of pre-chemotherapy supportive care medication does occur, discharge Patient is not required to occur subsequently. Even if discharge Patient does not occur subsequently, the behavior is still required to hold, until the end of the event sequence.
5. There might be many restricted intervals in the event sequence. If Patient has adverse reaction to administration of pre-chemotherapy supportive care medication occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of Patient has adverse reaction to administration of pre-chemotherapy supportive care medication, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

**BEHAVIOR:**
1. The events of primary interest in this behavior are stabilize Patient condition and Attending MD or Attending MD’s delegate decides Patient disposition.
2. The event of secondary interest in this behavior is stop administration of pre-chemotherapy supportive care medication.
3. Attending MD or Attending MD’s delegate decides Patient disposition is not allowed to occur until after stabilize Patient condition occurs.
4. Before the first stabilize Patient condition occurs, stop administration of pre-chemotherapy supportive care medication is allowed to occur zero or more times.
5. Stabilize Patient condition is required to occur, whether or not Attending MD or Attending MD’s delegate decides Patient disposition eventually occurs.
6. Attending MD or Attending MD’s delegate decides Patient disposition is not required to occur after stabilize Patient condition occurs.
7. After stabilize Patient condition occurs, but before the first subsequent Attending MD or Attending MD’s delegate decides Patient disposition occurs:
   - Stabilize Patient condition is allowed to occur again, zero or more times;
   - Stop administration of pre-chemotherapy supportive care medication is not allowed to occur.
8. After stabilize Patient condition and the first subsequent Attending MD or Attending MD’s delegate decides Patient disposition occur:
   - Stop administration of pre-chemotherapy supportive care medication is allowed to occur zero or more times;
   - Both stabilize Patient condition and Attending MD or Attending MD’s delegate decides Patient disposition are allowed to occur again, zero or more times, and there are no restrictions on the occurrences of any future events.

FSA and DNL for Property Chemo.D.24a (D.22 → D.23)
**Event alphabet:**

- A: Patient has adverse reaction to administration of pre-chemotherapy supportive care medication
- B: discharge Patient
- C: Attending MD or Attending MD's delegate decides Patient disposition, stabilize Patient condition, stop administration of pre-chemotherapy supportive care medication

**SCOPE:**

1. From the start of any event sequence through to the end of that event sequence, the behavior must hold.

**BEHAVIOR:**

1. The events of primary interest in this behavior are Patient has adverse reaction to administration of pre-chemotherapy supportive care medication and discharge Patient.
2. The events of secondary interest in this behavior are Attending MD or Attending MD's delegate decides Patient disposition, stabilize Patient condition, and stop administration of pre-chemotherapy supportive care medication.
3. If Patient has adverse reaction to administration of pre-chemotherapy supportive care medication occurs, discharge Patient is required to occur subsequently.
4. Before the first Patient has adverse reaction to administration of pre-chemotherapy supportive care medication occurs:
   - discharge Patient is allowed to occur zero or more times;
   - All the events of secondary interest are allowed to occur zero or more times.
5. Patient has adverse reaction to administration of pre-chemotherapy supportive care medication is not required to occur.
6. After Patient has adverse reaction to administration of pre-chemotherapy supportive care medication occurs, but before the first subsequent discharge Patient occurs:
   - Patient has adverse reaction to administration of pre-chemotherapy supportive care medication is allowed to occur again, zero or more times;
   - All the events of secondary interest are allowed to occur zero or more times.
7. After Patient has adverse reaction to administration of pre-chemotherapy supportive care medication and the first subsequent discharge Patient occur:
   - All the events of secondary interest are allowed to occur zero or more times;
   - discharge Patient is allowed to occur again, zero or more times, before another Patient has adverse reaction to administration of pre-chemotherapy supportive care medication occurs; Patient has adverse reaction to administration of pre-chemotherapy supportive care medication is allowed to occur again and, if it does, then the situation is the same as when the first Patient has adverse reaction to administration of pre-chemotherapy supportive care medication occurred; meaning that the restrictions described in parts 3, 6, and 7 would again apply.

**FSA and DNL for Property Chemo.D.24b**
Event alphabet:
* A: administer pre-chemotherapy supportive care medications
* B: administer chemotherapy

SCOPE:
1. From the start of any event sequence through to the end of that event sequence, the behavior must hold.

BEHAVIOR:
1. The events of primary interest in this behavior are administer pre-chemotherapy supportive care medications and administer chemotherapy.
2. There are no events of secondary interest in this behavior.
3. administer chemotherapy is not allowed to occur until after administer pre-chemotherapy supportive care medications occurs.
4. administer pre-chemotherapy supportive care medications is not required to occur.
5. Even if administer pre-chemotherapy supportive care medications does occur, administer chemotherapy is not required to occur after administer pre-chemotherapy supportive care medications occurs.
6. After administer pre-chemotherapy supportive care medications occurs, but before the first subsequent administer chemotherapy occurs, administer pre-chemotherapy supportive care medications is not allowed to occur again.
7. After administer pre-chemotherapy supportive care medications and the first subsequent administer chemotherapy occur:
   - administer chemotherapy is not allowed to occur again until after another administer pre-chemotherapy supportive care medications occurs; administer pre-chemotherapy supportive care medications is allowed to occur again and, if it does, then the situation is the same as when the first administer pre-chemotherapy supportive care medications occurred, meaning that the restrictions described in parts 5, 6, and 7 would again apply.

FSA and DNL for Property Chemo.D.20
**Event alphabet:**
- A: Patient has adverse reaction to administration of chemotherapy
- B: stop administration of chemotherapy
- C: Attending MD or Attending MD’s delegate decides Patient disposition, stabilize Patient condition

**SCOPE:**
1. From the start of any event sequence through to the end of that event sequence, the behavior must hold.

**BEHAVIOR:**
1. The events of primary interest in this behavior are Patient has adverse reaction to administration of chemotherapy and stop administration of chemotherapy.
2. The events of secondary interest in this behavior are Attending MD or Attending MD’s delegate decides Patient disposition and stabilize Patient condition.
3. If Patient has adverse reaction to administration of chemotherapy occurs, stop administration of chemotherapy is required to occur subsequently.
4. Before the first Patient has adverse reaction to administration of chemotherapy occurs:
   - stop administration of chemotherapy is allowed to occur zero or more times;
   - All the events of secondary interest are allowed to occur zero or more times.
5. Patient has adverse reaction to administration of chemotherapy is not required to occur.
6. After Patient has adverse reaction to administration of chemotherapy occurs, but before the first subsequent stop administration of chemotherapy occurs:
   - Patient has adverse reaction to administration of chemotherapy is not allowed to occur again;
   - None of the events of secondary interest are allowed to occur.
7. After Patient has adverse reaction to administration of chemotherapy and the first subsequent stop administration of chemotherapy occur:
   - All the events of secondary interest are allowed to occur zero or more times;
   - stop administration of chemotherapy is allowed to occur again, zero or more times, before another Patient has adverse reaction to administration of chemotherapy occurs. Patient has adverse reaction to administration of chemotherapy is allowed to occur again and, if it does, then the situation is the same as when the first Patient has adverse reaction to administration of chemotherapy occurred, meaning that the restrictions described in parts 3, 6, and 7 would again apply.

FSA and DNL for Property Chemo.E.1
Event alphabet:
- • A: Patient has an adverse reaction to administration of chemotherapy
- • B: Stabilize Patient condition
- • C: Attending MD or Attending MD’s delegate decides Patient disposition

**SCOPE:**
1. From the start of any event sequence through to the end of that event sequence, the behavior must hold.

**BEHAVIOR:**
1. The events of primary interest in this behavior are *Patient has an adverse reaction to administration of chemotherapy* and *stabilize Patient condition*.
2. The event of secondary interest in this behavior is *Attending MD or Attending MD’s delegate decides Patient disposition*.
3. If *Patient has an adverse reaction to administration of chemotherapy* occurs, *stabilize Patient condition* is required to occur subsequently.
4. Before the first *Patient has an adverse reaction to administration of chemotherapy* occurs:
   - *stabilize Patient condition* is allowed to occur zero or more times;
   - *Attending MD or Attending MD’s delegate decides Patient disposition* is allowed to occur zero or more times.
5. *Patient has an adverse reaction to administration of chemotherapy* is not required to occur.
6. After *Patient has an adverse reaction to administration of chemotherapy* occurs, but before the first subsequent *stabilize Patient condition* occurs:
   - *Patient has an adverse reaction to administration of chemotherapy* is not allowed to occur again;
   - *Attending MD or Attending MD’s delegate decides Patient disposition* is not allowed to occur.
7. After *Patient has an adverse reaction to administration of chemotherapy* and the first subsequent *stabilize Patient condition* occurs:
   - *Attending MD or Attending MD’s delegate decides Patient disposition* is allowed to occur zero or more times;
   - *stabilize Patient condition* is allowed to occur again, zero or more times, before another *Patient has an adverse reaction to administration of chemotherapy* occurs; *Patient has an adverse reaction to administration of chemotherapy* is allowed to occur again and, if it does, then the situation is the same as when the first *Patient has an adverse reaction to administration of chemotherapy* occurred, meaning that the restrictions described in parts 3, 6, and 7 would again apply.

FSA and DNL for Property Chemo.E.2
Event alphabet:
- A: Patient has an adverse reaction to administration of chemotherapy
- B: Attending MD or Attending MD’s delegate decides Patient disposition

**SCOPE:**
1. From the start of any event sequence through to the end of that event sequence, the behavior must hold.

**BEHAVIOR:**
1. The events of primary interest in this behavior are Patient has an adverse reaction to administration of chemotherapy and Attending MD or Attending MD’s delegate decides Patient disposition.
2. There are no events of secondary interest in this behavior.
3. If Patient has an adverse reaction to administration of chemotherapy occurs, Attending MD or Attending MD’s delegate decides Patient disposition is required to occur subsequently.
4. Before the first Patient has an adverse reaction to administration of chemotherapy occurs, Attending MD or Attending MD’s delegate decides Patient disposition is allowed to occur zero or more times.
5. Patient has an adverse reaction to administration of chemotherapy is not required to occur.
6. After Patient has an adverse reaction to administration of chemotherapy occurs, but before the first subsequent Attending MD or Attending MD’s delegate decides Patient disposition occurs, Patient has an adverse reaction to administration of chemotherapy is not allowed to occur again.
7. After Patient has an adverse reaction to administration of chemotherapy and the first subsequent Attending MD or Attending MD’s delegate decides Patient disposition occur:
   - Attending MD or Attending MD’s delegate decides Patient disposition is allowed to occur again, zero or more times, before another Patient has an adverse reaction to administration of chemotherapy occurs; Patient has an adverse reaction to administration of chemotherapy occurred, meaning that the restrictions described in parts 3, 6, and 7 would again apply.

FSA and DNL for Property Chemo.E.3
Event alphabet:

- A: stop administration of chemotherapy
- B: stabilize Patient condition
- START: Patient has adverse reaction to the administration of chemotherapy
- END: discharge Patient
- C: Attending MD or Attending MD’s delegate decides Patient disposition

SCOPE:

1. A restricted interval in the event sequence can have both a starting delimiter, Patient has adverse reaction to the administration of chemotherapy, and an ending delimiter, discharge Patient.
2. The behavior is required to hold from an occurrence of Patient has adverse reaction to the administration of chemotherapy, if it ever occurs, through to the first subsequent occurrence of discharge Patient, if it ever occurs.
3. If there are multiple occurrences of Patient has adverse reaction to the administration of chemotherapy without an occurrence of discharge Patient in between them, only the last of those occurrences of Patient has adverse reaction to the administration of chemotherapy starts a restricted interval; each of those occurrences of Patient has adverse reaction to the administration of chemotherapy resets the beginning of this restricted interval.
4. Patient has adverse reaction to the administration of chemotherapy is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if Patient has adverse reaction to the administration of chemotherapy does occur, discharge Patient is not required to occur subsequently. Even if discharge Patient does not occur subsequently, the behavior is still required to hold, until the end of the event sequence.
5. There might be many restricted intervals in the event sequence. If Patient has adverse reaction to the administration of chemotherapy occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of Patient has adverse reaction to the administration of chemotherapy, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

BEHAVIOR:

1. The events of primary interest in this behavior are stop administration of chemotherapy and stabilize Patient condition.
2. The event of secondary interest in this behavior is Attending MD or Attending MD’s delegate decides Patient disposition.
3. stabilize Patient condition is not allowed to occur until after stop administration of chemotherapy occurs.
4. Before the first stop administration of chemotherapy occurs, Attending MD or Attending MD’s delegate decides Patient disposition is allowed to occur zero or more times.
5. stop administration of chemotherapy is required to occur, whether or not stabilize Patient condition eventually occurs.
6. stabilize Patient condition is not required to occur after stop administration of chemotherapy occurs.
7. After stop administration of chemotherapy occurs, but before the first subsequent stabilize Patient condition occurs:
   - stop administration of chemotherapy is not allowed to occur again;
   - Attending MD or Attending MD’s delegate decides Patient disposition is not allowed to occur.
8. After stop administration of chemotherapy and the first subsequent stabilize Patient condition occur:
   - Attending MD or Attending MD’s delegate decides Patient disposition is allowed to occur zero or more times;
   - Both stop administration of chemotherapy and stabilize Patient condition are allowed to occur again, zero or more times, and there are no restrictions on the occurrences of any future events.

FSA and DNL for Property Chemo.E.4 (E.1 → E.2)
Event alphabet:
- A: stabilize Patient condition
- B: Attending MD or Attending MD’s delegate decides Patient disposition
- START: Patient has adverse reaction to the administration of chemotherapy
- END: discharge Patient
- C: stop administration of chemotherapy

SCOPE:
1. A restricted interval in the event sequence can have both a starting delimiter, Patient has adverse reaction to the administration of chemotherapy, and an ending delimiter, discharge Patient.
2. The behavior is required to hold from an occurrence of Patient has adverse reaction to the administration of chemotherapy, if it ever occurs, through to the first subsequent occurrence of discharge Patient, if it ever occurs.
3. If there are multiple occurrences of Patient has adverse reaction to the administration of chemotherapy without an occurrence of discharge Patient in between them, only the last of those occurrences of Patient has adverse reaction to the administration of chemotherapy starts a restricted interval; each of those occurrences of Patient has adverse reaction to the administration of chemotherapy resets the beginning of this restricted interval.
4. Patient has adverse reaction to the administration of chemotherapy is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if Patient has adverse reaction to the administration of chemotherapy does occur, discharge Patient is not required to occur subsequently. Even if discharge Patient does not occur subsequently, the behavior is still required to hold, until the end of the event sequence.
5. There might be many restricted intervals in the event sequence. If Patient has adverse reaction to the administration of chemotherapy occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of Patient has adverse reaction to the administration of chemotherapy, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

BEHAVIOR:
1. The events of primary interest in this behavior are stabilize Patient condition and Attending MD or Attending MD’s delegate decides Patient disposition.
2. The event of secondary interest in this behavior is stop administration of chemotherapy.
3. Attending MD or Attending MD’s delegate decides Patient disposition is not allowed to occur until after stabilize Patient condition occurs.
4. Before the first stabilize Patient condition occurs, stop administration of chemotherapy is allowed to occur zero or more times.
5. stabilize Patient condition is required to occur, whether or not Attending MD or Attending MD’s delegate decides Patient disposition eventually occurs.
6. Attending MD or Attending MD’s delegate decides Patient disposition is not required to occur after stabilize Patient condition occurs.
7. After stabilize Patient condition occurs, but before the first subsequent Attending MD or Attending MD’s delegate decides Patient disposition occurs:
   - stabilize Patient condition is allowed to occur again, zero or more times;
   - stop administration of chemotherapy is not allowed to occur.
8. After stabilize Patient condition and the first subsequent Attending MD or Attending MD’s delegate decides Patient disposition occur:
   - stop administration of chemotherapy is allowed to occur zero or more times;
   - Both stabilize Patient condition and Attending MD or Attending MD’s delegate decides Patient disposition are allowed to occur again, zero or more times, and there are no restrictions on the occurrences of any future events.

FSA and DNL for Property Chemo.E.4 (E.2 → E.3)
Event alphabet:
- A: Patient has adverse reaction to the administration of chemotherapy
- B: discharge Patient
- C: Attending MD or Attending MD’s delegate decides Patient disposition, stabilize Patient condition, stop administration of chemotherapy


**SCOPE:**
1. From the start of any event sequence through to the end of that event sequence, the behavior must hold.

**BEHAVIOR:**
1. The events of primary interest in this behavior are Patient has adverse reaction to the administration of chemotherapy and discharge Patient.
2. The events of secondary interest in this behavior are Attending MD or Attending MD’s delegate decides Patient disposition, stabilize Patient condition, and stop administration of chemotherapy.
3. If Patient has adverse reaction to the administration of chemotherapy occurs, discharge Patient is required to occur subsequently.
4. Before the first Patient has adverse reaction to the administration of chemotherapy occurs:
   - discharge Patient is allowed to occur zero or more times;
   - All the events of secondary interest are allowed to occur zero or more times.
5. Patient has adverse reaction to the administration of chemotherapy is not required to occur.
6. After Patient has adverse reaction to the administration of chemotherapy occurs, but before the first subsequent discharge Patient occurs:
   - Patient has adverse reaction to the administration of chemotherapy is allowed to occur again, zero or more times;
   - All the events of secondary interest are allowed to occur zero or more times.
7. After Patient has adverse reaction to the administration of chemotherapy and the first subsequent discharge Patient occur:
   - All the events of secondary interest are allowed to occur zero or more times;
   - discharge Patient is allowed to occur again, zero or more times, before another Patient has adverse reaction to the administration of chemotherapy occurs. Patient has adverse reaction to the administration of chemotherapy is allowed to occur again and, if it does, then the situation is the same as when the first Patient has adverse reaction to the administration of chemotherapy occurred, meaning that the restrictions described in parts 3, 6, and 7 would again apply.

FSA and DNL for Property Chemo.E.4b

469
Event alphabet:
- A: Patient becomes well enough to be discharged
- B: discharge Patient
- C: Patient becomes too ill to be discharged

SCOPE:
1. From the start of any event sequence through to the end of that event sequence, the behavior must hold.

BEHAVIOR:
1. The events of primary interest in this behavior are Patient becomes well enough to be discharged and discharge Patient.
2. The event of secondary interest in this behavior is Patient becomes too ill to be discharged.
3. discharge Patient is not allowed to occur until after Patient becomes well enough to be discharged occurs.
4. Before the first Patient becomes well enough to be discharged occurs, Patient becomes too ill to be discharged is allowed to occur zero or more times.
5. Patient becomes well enough to be discharged is not required to occur.
6. Even if Patient becomes well enough to be discharged does occur, discharge Patient is not required to occur after Patient becomes well enough to be discharged occurs.
7. After Patient becomes well enough to be discharged occurs, but before the first subsequent discharge Patient occurs:
   - Patient becomes well enough to be discharged is allowed to occur again, zero or more times;
   - Patient becomes too ill to be discharged is allowed to occur zero or more times.
8. After Patient becomes well enough to be discharged and the first subsequent discharge Patient occur:
   - Patient becomes too ill to be discharged is allowed to occur zero or more times;
   - discharge Patient is not allowed to occur again until after another Patient becomes well enough to be discharged occurs; Patient becomes well enough to be discharged is allowed to occur again and, if it does, then the situation is the same as when the first Patient becomes well enough to be discharged occurred, meaning that the restrictions described in parts 6, 7, and 8 would again apply.

FSA and DNL for Property Chemo.F.1a
Event alphabet:
- A: Patient becomes well enough to be discharged
- START: Patient becomes too ill to be discharged
- END: discharge Patient

**SCOPE:**
1. A restricted interval in the event sequence can have both a starting delimiter, Patient becomes too ill to be discharged, and an ending delimiter, discharge Patient.
2. The behavior is required to hold from an occurrence of Patient becomes too ill to be discharged, if it ever occurs, through to the first subsequent occurrence of discharge Patient, if it ever occurs.
3. If there are multiple occurrences of Patient becomes too ill to be discharged without an occurrence of discharge Patient in between them, only the last of those occurrences of Patient becomes too ill to be discharged potentially starts a restricted interval; each of those occurrences of Patient becomes too ill to be discharged resets the beginning of this restricted interval.
4. Patient becomes too ill to be discharged is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if Patient becomes too ill to be discharged does occur, discharge Patient is not required to occur subsequently and if it does not occur subsequently, then the behavior is not required to hold for the remainder of the event sequence.
5. There might be many restricted intervals in the event sequence. If Patient becomes too ill to be discharged occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of Patient becomes too ill to be discharged, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a potential new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

**BEHAVIOR:**
1. The event of primary interest in this behavior is Patient becomes well enough to be discharged.
2. There are no events of secondary interest in this behavior.
3. Patient becomes well enough to be discharged is required to occur at least once.

FSA and DNL for Property Chemo.F.1b
Event alphabet:
- A: observe Patient
- START: determine that Patient needs observation
- END: discharge Patient

SCOPE:
1. A restricted interval in the event sequence can have both a starting delimiter, determine that Patient needs observation, and an ending delimiter, discharge Patient.
2. The behavior is required to hold from an occurrence of determine that Patient needs observation, if it ever occurs, through to the first subsequent occurrence of discharge Patient, if it ever occurs.
3. If there are multiple occurrences of determine that Patient needs observation without an occurrence of discharge Patient in between them, only the last of those occurrences of determine that Patient needs observation starts a restricted interval; each of those occurrences of determine that Patient needs observation resets the beginning of this restricted interval.
4. determine that Patient needs observation is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if determine that Patient needs observation does occur, discharge Patient is not required to occur subsequently. Even if discharge Patient does not occur subsequently, the behavior is still required to hold, until the end of the event sequence.
5. There might be many restricted intervals in the event sequence. If determine that Patient needs observation occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of determine that Patient needs observation, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

BEHAVIOR:
1. The event of primary interest in this behavior is observe Patient.
2. There are no events of secondary interest in this behavior.
3. observe Patient is required to occur at least once.
Event alphabet:
- A: give all necessary prescriptions for post-chemotherapy supportive care medications to Patient
- START: administer chemotherapy
- END: discharge Patient

SCOPE:
1. A restricted interval in the event sequence can have both a starting delimiter, administer chemotherapy, and an ending delimiter, discharge Patient.
2. The behavior is required to hold from an occurrence of administer chemotherapy, if it ever occurs, through to the first subsequent occurrence of discharge Patient, if it ever occurs.
3. If there are multiple occurrences of administer chemotherapy without an occurrence of discharge Patient in between them, only the last of those occurrences of administer chemotherapy potentially starts a restricted interval; each of those occurrences of administer chemotherapy resets the beginning of this restricted interval.
4. administer chemotherapy is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if administer chemotherapy does occur, discharge Patient is not required to occur subsequently and if it does not occur subsequently, then the behavior is not required to hold for the remainder of the event sequence.
5. There might be many restricted intervals in the event sequence. If administer chemotherapy occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of administer chemotherapy, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a potential new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

BEHAVIOR:
1. The event of primary interest in this behavior is give all necessary prescriptions for post-chemotherapy supportive care medications to Patient.
2. There are no events of secondary interest in this behavior.
3. give all necessary prescriptions for post-chemotherapy supportive care medications to Patient is required to occur exactly once.

FSA and DNL for Property Chemo.F.3
Event alphabet:
- A: give all post-chemotherapy instructions to Patient
- START: administer chemotherapy
- END: discharge Patient

SCOPE:
1. A restricted interval in the event sequence can have both a starting delimiter, administer chemotherapy, and an ending delimiter, discharge Patient.
2. The behavior is required to hold from an occurrence of administer chemotherapy, if it ever occurs, through to the first subsequent occurrence of discharge Patient, if it ever occurs.
3. If there are multiple occurrences of administer chemotherapy without an occurrence of discharge Patient in between them, only the last of those occurrences of administer chemotherapy potentially starts a restricted interval; each of those occurrences of administer chemotherapy resets the beginning of this restricted interval.
4. administer chemotherapy is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if administer chemotherapy does occur, discharge Patient is not required to occur subsequently and if it does not occur subsequently, then the behavior is not required to hold for the remainder of the event sequence.
5. There might be many restricted intervals in the event sequence. If administer chemotherapy occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of administer chemotherapy, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a potential new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

BEHAVIOR:
1. The event of primary interest in this behavior is give all post-chemotherapy instructions to Patient.
2. There are no events of secondary interest in this behavior.
3. give all post-chemotherapy instructions to Patient is required to occur at least once.

FSA and DNL for Property Chemo.F.4
Event alphabet:
- A: schedule follow-up appointment
- START: administer chemotherapy
- END: discharge Patient

SCOPE:
1. A restricted interval in the event sequence can have both a starting delimiter, administer chemotherapy, and an ending delimiter, discharge Patient.
2. The behavior is required to hold from an occurrence of administer chemotherapy, if it ever occurs, through to the first subsequent occurrence of discharge Patient, if it ever occurs.
3. If there are multiple occurrences of administer chemotherapy without an occurrence of discharge Patient in between them, only the last of those occurrences of administer chemotherapy potentially starts a restricted interval; each of those occurrences of administer chemotherapy resets the beginning of this restricted interval.
4. administer chemotherapy is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if administer chemotherapy does occur, discharge Patient is not required to occur subsequently and if it does not occur subsequently, then the behavior is not required to hold for the remainder of the event sequence.
5. There might be many restricted intervals in the event sequence. If administer chemotherapy occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of administer chemotherapy, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a potential new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

BEHAVIOR:
1. The event of primary interest in this behavior is schedule follow-up appointment.
2. There are no events of secondary interest in this behavior.
3. schedule follow-up appointment is required to occur at least once.

FSA and DNL for Property Chemo.F.5
**Event alphabet:**
- A: enter all Patient laboratory results and chemotherapy administration data into Patient record
- START: administer chemotherapy
- END: business day ends

**SCOPE:**
1. A restricted interval in the event sequence can have both a starting delimiter, administer chemotherapy, and an ending delimiter, business day ends.
2. The behavior is required to hold from an occurrence of administer chemotherapy, if it ever occurs, through to the first subsequent occurrence of business day ends, if it ever occurs.
3. If there are multiple occurrences of administer chemotherapy without an occurrence of business day ends in between them, only the last of those occurrences of administer chemotherapy starts a restricted interval; each of those occurrences of administer chemotherapy resets the beginning of this restricted interval.
4. administer chemotherapy is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if administer chemotherapy does occur, business day ends is not required to occur subsequently. Even if business day ends does not occur subsequently, the behavior is still required to hold, until the end of the event sequence.
5. There might be many restricted intervals in the event sequence. If administer chemotherapy occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of administer chemotherapy, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

**BEHAVIOR:**
1. The event of primary interest in this behavior is enter all Patient laboratory results and chemotherapy administration data into Patient record.
2. There are no events of secondary interest in this behavior.
3. enter all Patient laboratory results and chemotherapy administration data into Patient record is required to occur exactly once.

**FSA and DNL for Property Chemo.F.6a**
Event alphabet:
- A: administer chemotherapy
- B: business day ends

SCOPE:
1. From the start of any event sequence through to the end of that event sequence, the behavior must hold.

BEHAVIOR:
1. The events of primary interest in this behavior are administer chemotherapy and business day ends.
2. There are no events of secondary interest in this behavior.
3. If administer chemotherapy occurs, business day ends is required to occur subsequently.
4. Before the first administer chemotherapy occurs, business day ends is allowed to occur zero or more times.
5. administer chemotherapy is not required to occur.
6. After administer chemotherapy occurs, but before the first subsequent business day ends occurs, administer chemotherapy is allowed to occur again, zero or more times.
7. After administer chemotherapy and the first subsequent business day ends occur:
   - business day ends is allowed to occur again, zero or more times, before another administer chemotherapy occurs; administer chemotherapy is allowed to occur again and, if it does, then the situation is the same as when the first administer chemotherapy occurred, meaning that the restrictions described in parts 3, 6, and 7 would again apply.

FSA and DNL for Property Chemo.F.6b
C.3 Emergency Department

C.3.1 Description of the Domain

The Emergency Department (ED) case study is part of a larger medical safety project that our research group is participating in. In this case study, the domain expert was an M.D. with ED administration experience. We elicited all of the properties through interviews with him. We, the computer scientists, were the only ones who worked with PROPEL to formally specify the properties. Due to our limited access to this domain expert, he has not yet vetted any of the informal and formal property specifications, so they are considered to be in a draft stage. This domain expert does not have any prior knowledge of property specification formalisms or RE.

Due to resource limitations, we focused on a subset of the ED domain, specifically covering only a cross-section of the activities surrounding patient flow through the ED. In addition, all properties in this case study are assumed to be in the context of one patient, one or more visits to the emergency department by that patient, and the participation of one or more physicians in the management of that patient.

There are 26 informal property specifications in this case study, and they are given in Appendix C.3.2. See Section 5.4.3.2 for an explanation of the notation used in the informal specifications. Appendix C.3.3 gives the glossary for all of the bolded terms used in the informal specifications; see Section 5.4.3.1 for an explanation of the glossary structure. For the 21 informal ED property specifications that can be captured in PROPEL, there are 27 formal ED property specifications, and they are given in Appendix C.3.4. The FSA and DNL specifications are given for each property, with some modifications made to these two views to enable the property specifications to fit on a single page. The FSA specification is shown in two parts: at the top is a mapping from parameter names to specifier-specified events, labeled “Event alphabet:”, and directly below that mapping is an FSA specification of the property. The FSA is assumed to be total and deterministic\(^3\) and its transitions are labeled with the parameters, rather than with the specifier-specified events. The DNL specification is also shown in two parts, the scope and the behavior, but a third part, the preamble, is elided. The preamble is the same for every DNL property specification, and can be found in Section 3.2.4.

\(^3\)For brevity, we do not show the transitions that go to a non-accepting trap state. When no transition is provided that explicitly allows an event to occur, it should be assumed that an occurrence of that event puts the FSA into a non-accepting trap state.
### C.3.2 Organization of the Informal Property Specifications

<table>
<thead>
<tr>
<th>A</th>
<th>PATIENT IDENTIFICATION</th>
</tr>
</thead>
<tbody>
<tr>
<td>7</td>
<td>A.1 A Patient who has been admitted to the Emergency Department must have exactly one ID band secured to their body.</td>
</tr>
<tr>
<td>7</td>
<td>A.2 Before can ID band be secured to a Patient's body, that ID band must be created to uniquely identify that Patient.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>B</th>
<th>TREATMENT PROCEDURES</th>
</tr>
</thead>
<tbody>
<tr>
<td>7</td>
<td>B.1 it must be confirmed that that Patient has exactly one ID band, unless it is a life-or-death situation. Restricted types of management: performing tests for a Patient, administering medication to a Patient, or evaluating a Patient.</td>
</tr>
<tr>
<td>7</td>
<td>B.2 that Patient's stated first and last name and birth date must be obtained, unless that Patient is unable to communicate or it is a life-or-death situation. Restricted types of management: performing tests for a Patient, administering medication to a Patient, or evaluating a Patient.</td>
</tr>
<tr>
<td>7</td>
<td>B.3 it must be made sure that the first and last name and birth date on that Patient's ID band match that Patient's stated first and last name and birth date, unless that Patient is unable to communicate or it is a life-or-death situation. Restricted types of management: performing tests for a Patient, administering medication to a Patient, or evaluating a Patient.</td>
</tr>
<tr>
<td>7</td>
<td>B.4 it must be made sure that the first and last name and unique ID number on that Patient's ID band match the first and last name and unique ID number on the physician order, unless it is a life-or-death situation. Restricted types of management: performing tests for a Patient or administering medication to a Patient.</td>
</tr>
<tr>
<td>7</td>
<td>B.5 these activities described above must be performed in that order.</td>
</tr>
<tr>
<td>7</td>
<td>B.6 Before any management can be performed for a Patient, that Patient must sign an informed consent form, unless it is a life-or-death situation. After a Patient is brought into the Main Emergency Department, the following activities must be performed:</td>
</tr>
<tr>
<td>7</td>
<td>B.7 that Patient must be placed in an Emergency Department bed or a hall bed.</td>
</tr>
<tr>
<td>7</td>
<td>B.8 that Patient's information must be written on the patient status board.</td>
</tr>
<tr>
<td>X</td>
<td>B.9 Before any medication can be administered to a Patient, that Patient's allergies must be confirmed, unless that Patient is unable to communicate or it is a life-or-death situation.</td>
</tr>
<tr>
<td>7</td>
<td>B.10 If a Patient dies within 24 hours of being admitted to the Emergency Department, the coroner must be called.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>C</th>
<th>DOCUMENTATION CONSTRAINTS</th>
</tr>
</thead>
<tbody>
<tr>
<td>7</td>
<td>C.1 Before a physician order can be followed, that physician order must be signed by an MD.</td>
</tr>
<tr>
<td>7</td>
<td>C.2 After each time a Patient is admitted to the Emergency Department, a medical record must be completed for that Patient's visit.</td>
</tr>
<tr>
<td>7</td>
<td>C.3 If a Patient is brought to the Emergency Department by an Emergency Medical Technician, that Emergency Medical Technician must give their report to the Nurse assigned to that Patient or to the External Triage Nurse before that Emergency Medical Technician can leave the Emergency Department.</td>
</tr>
<tr>
<td>7</td>
<td>C.4 Before a Patient can leave the Emergency Department against medical advice, that Patient must sign a waiver.</td>
</tr>
</tbody>
</table>
D. ACTIVITIES THAT MUST BE PERFORMED BEFORE DISCHARGE

Before a Patient can be discharged from the Emergency Department, that Patient must complete full registration, unless that Patient leaves the Emergency Department against medical advice or without notifying a medical professional.

Before a Patient can be discharged from the Emergency Department, the following activities must be performed unless that Patient leaves the Emergency Department against medical advice or without notifying a medical professional:

- that Patient’s vital signs must be checked.

- an MD must evaluate that Patient.

- that Patient must be given discharge instructions.

- these activities described above must be performed in that order.

Before a Patient can be discharged from the Emergency Department, the following activities must be performed unless that Patient leaves the Emergency Department against medical advice or without notifying a medical professional:

To summarize: that Patient’s ID band must be verified by performing the following activities.

- it must be confirmed that that Patient has exactly one ID band.

- that Patient’s stated first and last name and birth date must be obtained, unless that Patient is unable to communicate.

- it must be made sure that the first and last name and birth date on that Patient’s ID band match that Patient’s stated first and last name and birth date, unless that Patient is unable to communicate.

- it must be made sure that the first and last name and unique ID number on that Patient’s ID band match the first and last name and unique ID number on the discharge order.

- these activities described above must be performed in that order.
### C.3.3 Glossary

<table>
<thead>
<tr>
<th>Term</th>
<th>Definition</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>allergies</td>
<td>confirm</td>
<td></td>
</tr>
<tr>
<td>coroner</td>
<td>be called</td>
<td>This occurs if a Patient dies within 24 hours of being admitted to the Emergency Department.</td>
</tr>
<tr>
<td>discharge instructions</td>
<td>give to Patient</td>
<td></td>
</tr>
<tr>
<td>discharge order</td>
<td>make sure match between first and last name and unique ID number on Patient ID band and first and last name and unique ID number on order</td>
<td></td>
</tr>
<tr>
<td>Emergency Department</td>
<td>be admitted to</td>
<td></td>
</tr>
<tr>
<td></td>
<td>be brought to</td>
<td></td>
</tr>
<tr>
<td></td>
<td>discharge from</td>
<td></td>
</tr>
<tr>
<td></td>
<td>leave against medical advice</td>
<td>a.k.a. &quot;AMA&quot; (Against Medical Advice)</td>
</tr>
<tr>
<td></td>
<td>leave without notifying a medical professional</td>
<td>a.k.a. &quot;MIA&quot; (Missing In Action)</td>
</tr>
<tr>
<td>Emergency Department bed</td>
<td>place Patient in</td>
<td></td>
</tr>
<tr>
<td>Emergency Medical Technician</td>
<td>bring Patient to Emergency Department</td>
<td></td>
</tr>
<tr>
<td></td>
<td>give report to a Nurse</td>
<td></td>
</tr>
<tr>
<td></td>
<td>leave Emergency Department</td>
<td></td>
</tr>
<tr>
<td>Emergency Medical Technician report</td>
<td>give to Nurse</td>
<td></td>
</tr>
<tr>
<td>External Triage Nurse</td>
<td>give Emergency Medical Technician report to</td>
<td></td>
</tr>
<tr>
<td>first and last name and birth date</td>
<td>be on Patient ID band</td>
<td></td>
</tr>
<tr>
<td></td>
<td>be stated</td>
<td></td>
</tr>
<tr>
<td></td>
<td>make sure match between Patient ID band and physician order</td>
<td></td>
</tr>
<tr>
<td></td>
<td>make sure match between Patient ID band and stated information</td>
<td></td>
</tr>
<tr>
<td>obtain</td>
<td>Get access to a Patient’s first and last name and birth date by some means that is independent of that Patient’s record in the medical facility.</td>
<td>Possible means of obtaining a Patient’s first and last name and birth date, in order of most to least preferred: 1. Ask Patient to state their first and last name and birth date. 2. Ask Patient’s family (or legal guardian) to state their first and last name and birth date. 3. Go hunting through Patient’s belongings to find first and last name and birth date. (This is generally frowned upon.) 4. Use John/Jane Doe protocol.</td>
</tr>
<tr>
<td>state</td>
<td></td>
<td></td>
</tr>
<tr>
<td>full registration</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Term</td>
<td>Definition</td>
<td>Notes</td>
</tr>
<tr>
<td>---------------------------</td>
<td>----------------------------------------------------------------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>full registration</td>
<td>create to uniquely identify Patient</td>
<td>To uniquely identify a Patient, an ID band must have that Patient’s name (if it is known, otherwise the John/Jane Doe protocol should be used) and a unique ID number. Other information that an ID band can contain: that Patient’s birth date, Attending MD, and admission date and/or time.</td>
</tr>
<tr>
<td>hall bed</td>
<td>place Patient in</td>
<td></td>
</tr>
<tr>
<td>ID band</td>
<td>be secured to Patient’s body</td>
<td></td>
</tr>
<tr>
<td></td>
<td>be verified</td>
<td></td>
</tr>
<tr>
<td></td>
<td>confirm the presence of exactly one on Patient</td>
<td></td>
</tr>
<tr>
<td>informed consent form</td>
<td>sign</td>
<td></td>
</tr>
<tr>
<td>life-or-death situation</td>
<td></td>
<td></td>
</tr>
<tr>
<td>management</td>
<td>be performed for Patient</td>
<td>Includes performing tests for a Patient, doing an evaluation of a Patient, following a physician order, and administering medications to a Patient. Does not include discharging a Patient.</td>
</tr>
<tr>
<td>MD</td>
<td>evaluate Patient</td>
<td></td>
</tr>
<tr>
<td></td>
<td>sign physician order</td>
<td></td>
</tr>
<tr>
<td>medical advice</td>
<td>leave Emergency Department against</td>
<td></td>
</tr>
<tr>
<td>medical record</td>
<td>complete</td>
<td></td>
</tr>
<tr>
<td>medication</td>
<td>administer to Patient</td>
<td></td>
</tr>
<tr>
<td>Nurse</td>
<td>be assigned to Patient</td>
<td>Can include an External Triage Nurse.</td>
</tr>
<tr>
<td></td>
<td>give Emergency Medical Technician report to</td>
<td></td>
</tr>
<tr>
<td>Patient</td>
<td>administer medications to</td>
<td></td>
</tr>
<tr>
<td></td>
<td>be able to communicate</td>
<td>Patient is able to communicate verbally and/or with writing. This term assumes mental competence, but does not assume accuracy of communicated information.</td>
</tr>
<tr>
<td></td>
<td>be admitted to Emergency Department</td>
<td></td>
</tr>
<tr>
<td></td>
<td>be asked to state first and last name and birth date</td>
<td></td>
</tr>
<tr>
<td></td>
<td>be brought to Emergency Department</td>
<td></td>
</tr>
<tr>
<td></td>
<td>be discharged from Emergency Department</td>
<td></td>
</tr>
<tr>
<td></td>
<td>be given discharge instructions</td>
<td></td>
</tr>
<tr>
<td></td>
<td>be in a life-or-death situation</td>
<td></td>
</tr>
<tr>
<td></td>
<td>bring to Emergency Department</td>
<td></td>
</tr>
<tr>
<td>Term</td>
<td>Definition</td>
<td>Notes</td>
</tr>
<tr>
<td>------</td>
<td>------------</td>
<td>-------</td>
</tr>
<tr>
<td>Patient</td>
<td>bring to Emergency Department</td>
<td>complete full registration</td>
</tr>
<tr>
<td>die</td>
<td>If this occurs within 24 hours of being admitted to the Emergency Department, the coroner must be called.</td>
<td></td>
</tr>
<tr>
<td>discharge</td>
<td></td>
<td></td>
</tr>
<tr>
<td>evaluate</td>
<td></td>
<td></td>
</tr>
<tr>
<td>have allergies</td>
<td></td>
<td></td>
</tr>
<tr>
<td>have ID band secured to their body</td>
<td></td>
<td></td>
</tr>
<tr>
<td>have vital signs checked</td>
<td></td>
<td></td>
</tr>
<tr>
<td>leave Emergency Department against medical advice</td>
<td>a.k.a. 'AMA (Against Medical Advice) Patient'</td>
<td></td>
</tr>
<tr>
<td>leave Emergency Department without notifying a medical professional</td>
<td>a.k.a. 'MIA (Missing In Action) Patient'</td>
<td></td>
</tr>
<tr>
<td>obtain stated first and last name and birth date from</td>
<td></td>
<td></td>
</tr>
<tr>
<td>perform test for</td>
<td></td>
<td></td>
</tr>
<tr>
<td>place in bed</td>
<td>Can either be a regular Emergency Department bed or a hall bed.</td>
<td></td>
</tr>
<tr>
<td>sign informed consent form</td>
<td></td>
<td></td>
</tr>
<tr>
<td>sign waiver</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Patient information</td>
<td>This is the Patient's first and last name, location (i.e., bed number and hall number), current status (e.g., &quot;to be seen&quot;, &quot;to be admitted&quot;), and chief complaint.</td>
<td></td>
</tr>
<tr>
<td>write on patient status board</td>
<td></td>
<td></td>
</tr>
<tr>
<td>patient status board</td>
<td></td>
<td></td>
</tr>
<tr>
<td>write Patient information on</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Patient visit</td>
<td>complete medical record for</td>
<td></td>
</tr>
<tr>
<td>Patient's body</td>
<td></td>
<td></td>
</tr>
<tr>
<td>physician order</td>
<td></td>
<td></td>
</tr>
<tr>
<td>follow</td>
<td></td>
<td></td>
</tr>
<tr>
<td>make sure match between first and last name and unique ID number on Patient ID band and first and last name and unique ID number on order</td>
<td></td>
<td></td>
</tr>
<tr>
<td>sign</td>
<td></td>
<td></td>
</tr>
<tr>
<td>test</td>
<td>This is a type of management.</td>
<td></td>
</tr>
<tr>
<td>be performed for Patient</td>
<td></td>
<td></td>
</tr>
<tr>
<td>unique ID number</td>
<td>make sure match between physician order and Patient ID band</td>
<td></td>
</tr>
<tr>
<td>vital signs</td>
<td>check</td>
<td></td>
</tr>
<tr>
<td>waiver</td>
<td></td>
<td></td>
</tr>
<tr>
<td>sign</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
C.3.4 Formal Property Specifications

Event alphabet:
- A: admit Patient to ED
- B: secure ID band to Patient's body

SCOPE:
1. From the start of any event sequence through to the end of that event sequence, the behavior must hold.

BEHAVIOR:
1. The events of primary interest in this behavior are admit Patient to ED and secure ID band to Patient's body.
2. There are no events of secondary interest in this behavior.
3. If admit Patient to ED occurs, secure ID band to Patient's body is required to occur subsequently.
4. Before the first admit Patient to ED occurs, secure ID band to Patient's body is not allowed to occur.
5. admit Patient to ED is not required to occur.
6. After admit Patient to ED occurs, but before the first subsequent secure ID band to Patient's body occurs, admit Patient to ED is not allowed to occur again.
7. After admit Patient to ED and the first subsequent secure ID band to Patient's body occur:
   - secure ID band to Patient's body is not allowed to occur again until after another admit Patient to ED occurs; admit Patient to ED is allowed to occur again and, if it does, then the situation is the same as when the first admit Patient to ED occurred, meaning that the restrictions described in parts 3, 6, and 7 would again apply.

FSA and DNL for Property ED.A.1
Event alphabet:
• A: create unique ID band
• B: secure ID band to Patient’s body
• START: admit Patient to ED
• END: discharge Patient from ED

SCOPE:
1. A restricted interval in the event sequence can have both a starting delimiter, admit Patient to ED, and an ending delimiter, discharge Patient from ED.
2. The behavior is required to hold from an occurrence of admit Patient to ED, if it ever occurs, through to the first subsequent occurrence of discharge Patient from ED, if it ever occurs.
3. If there are multiple occurrences of admit Patient to ED without an occurrence of discharge Patient from ED in between them, only the first of those occurrences of admit Patient to ED starts a restricted interval; later occurrences of admit Patient to ED within this restricted interval do not have an effect.
4. admit Patient to ED is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if admit Patient to ED does occur, discharge Patient from ED is not required to occur subsequently. Even if discharge Patient from ED does not occur subsequently, the behavior is still required to hold, until the end of the event sequence.
5. There might be many restricted intervals in the event sequence. If admit Patient to ED occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of admit Patient to ED, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

BEHAVIOR:
1. The events of primary interest in this behavior are create unique ID band and secure ID band to Patient’s body.
2. There are no events of secondary interest in this behavior.
3. secure ID band to Patient’s body is not allowed to occur until after create unique ID band occurs.
4. create unique ID band is not required to occur.
5. Even if create unique ID band does occur, secure ID band to Patient’s body is not required to occur after create unique ID band occurs.
6. After create unique ID band occurs, but before the first subsequent secure ID band to Patient’s body occurs, create unique ID band is not allowed to occur again.
7. After create unique ID band and the first subsequent secure ID band to Patient’s body occur:
   • Neither create unique ID band nor secure ID band to Patient’s body are allowed to occur again.

FSA and DNL for Property ED.A.2
Event alphabet:

- **A**: confirm presence of exactly one ID band
- **B**: perform management activity for Patient
- **START**: determine that it is not a life-or-death situation
- **END**: determine that it is life-or-death situation
- **C**: obtain Patient’s stated name and birth date, verify ID band and Patient’s stated name and birth date match, verify ID band and physician order match

**SCOPE:**

1. A restricted interval in the event sequence can have both a starting delimiter, `determine that it is not a life-or-death situation`, and an ending delimiter, `determine that it is life-or-death situation`.
2. The behavior is required to hold from an occurrence of `determine that it is not a life-or-death situation`, if it ever occurs, through to the first subsequent occurrence of `determine that it is life-or-death situation`, if it ever occurs.
3. If there are multiple occurrences of `determine that it is not a life-or-death situation` without an occurrence of `determine that it is life-or-death situation` in between them, only the first of those occurrences of `determine that it is not a life-or-death situation` starts a restricted interval; later occurrences of `determine that it is not a life-or-death situation` within this restricted interval do not have an effect.
4. `determine that it is not a life-or-death situation` is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if `determine that it is not a life-or-death situation` does occur, `determine that it is life-or-death situation` is not required to occur subsequently. Even if `determine that it is life-or-death situation` does not occur subsequently, the behavior is still required to hold, until the end of the event sequence.
5. There might be many restricted intervals in the event sequence. If `determine that it is not a life-or-death situation` occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of `determine that it is not a life-or-death situation`, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

**BEHAVIOR:**

1. The events of primary interest in this behavior are `confirm presence of exactly one ID band` and `perform management activity for Patient`.
2. The events of secondary interest in this behavior are `obtain Patient’s stated name and birth date`, `verify ID band and Patient’s stated name and birth date match`, and `verify ID band and physician order match`.
3. `perform management activity for Patient` is not allowed to occur until after `confirm presence of exactly one ID band` occurs.
4. Before the first `confirm presence of exactly one ID band` occurs, all the events of secondary interest are allowed to occur zero or more times.
5. `confirm presence of exactly one ID band` is not required to occur.
6. Even if `confirm presence of exactly one ID band` does occur, `perform management activity for Patient` is not required to occur after `confirm presence of exactly one ID band` occurs.
7. After `confirm presence of exactly one ID band` occurs, but before the first subsequent `perform management activity for Patient` occurs:
   - `confirm presence of exactly one ID band` is allowed to occur again, zero or more times;
   - All the events of secondary interest are allowed to occur zero or more times.
8. After `confirm presence of exactly one ID band` and the first subsequent `perform management activity for Patient` occur:
   - All the events of secondary interest are allowed to occur zero or more times;
   - `perform management activity for Patient` is not allowed to occur again until after another `confirm presence of exactly one ID band` occurs; `confirm presence of exactly one ID band` is allowed to occur again and, if it does, then the situation is the same as when the first `confirm presence of exactly one ID band` occurred, meaning that the restrictions described in parts 6, 7, and 8 would again apply.

FSA and DNL for Property ED.B.1
Event alphabet:
- A: verify ID band and physician order match
- B: perform tests for Patient OR follow physician order OR administer medication to Patient
- START: determine that it is not a life-or-death situation
- END: determine that it is a life-or-death situation

**SCOPE:**
1. A restricted interval in the event sequence can have both a starting delimiter, determine that it is not a life-or-death situation, and an ending delimiter, determine that it is a life-or-death situation.
2. The behavior is required to hold from an occurrence of determine that it is not a life-or-death situation, if it ever occurs, through to the first subsequent occurrence of determine that it is a life-or-death situation, if it ever occurs.
3. If there are multiple occurrences of determine that it is not a life-or-death situation without an occurrence of determine that it is a life-or-death situation in between them, only the first of those occurrences of determine that it is not a life-or-death situation starts a restricted interval; later occurrences of determine that it is not a life-or-death situation within this restricted interval do not have an effect.
4. determine that it is not a life-or-death situation is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if determine that it is not a life-or-death situation does occur, determine that it is a life-or-death situation is not required to occur subsequently. Even if determine that it is a life-or-death situation does not occur subsequently, the behavior is still required to hold, until the end of the event sequence.
5. There might be many restricted intervals in the event sequence. If determine that it is not a life-or-death situation occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of determine that it is not a life-or-death situation, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

**BEHAVIOR:**
1. The events of primary interest in this behavior are verify ID band and physician order match and perform tests for Patient OR follow physician order OR administer medication to Patient.
2. There are no events of secondary interest in this behavior.
3. perform tests for Patient OR follow physician order OR administer medication to Patient is not allowed to occur until after verify ID band and physician order match occurs.
4. verify ID band and physician order match is not required to occur.
5. Even if verify ID band and physician order match does occur, perform tests for Patient OR follow physician order OR administer medication to Patient is not required to occur after verify ID band and physician order match occurs.
6. After verify ID band and physician order match occurs, but before the first subsequent perform tests for Patient OR follow physician order OR administer medication to Patient occurs, verify ID band and physician order match is allowed to occur again, zero or more times.
7. After verify ID band and physician order match and the first subsequent perform tests for Patient OR follow physician order OR administer medication to Patient occurs:
   - perform tests for Patient OR follow physician order OR administer medication to Patient is not allowed to occur again until after another verify ID band and physician order match occurs; verify ID band and physician order match is allowed to occur again and, if it does, then the situation is the same as when the first verify ID band and physician order match occurred, meaning that the restrictions described in parts 5, 6, and 7 would again apply.

FSA and DNL for Property ED.B.4
Event alphabet:
- A: confirm presence of exactly one ID band
- B: obtain Patient’s stated name and birth date
- END: perform management activity i
- C: verify ID band and Patient’s stated name and birth date match, verify ID band and physician order match

**SCOPE:**
1. There can be at most one restricted interval in the event sequence and it has an ending delimiter, perform management activity i.
2. The behavior is required to hold from the start of the event sequence through to the first occurrence of perform management activity i, if it ever occurs.
3. perform management activity i is not required to occur and if it never occurs, then the behavior is required to hold throughout the entire event sequence.
4. There are no restrictions imposed on the occurrences of any events after the first occurrence of perform management activity i, if it ever occurs.

**BEHAVIOR:**
1. The events of primary interest in this behavior are confirm presence of exactly one ID band and obtain Patient’s stated name and birth date.
2. The events of secondary interest in this behavior are verify ID band and Patient’s stated name and birth date match and verify ID band and physician order match.
3. obtain Patient’s stated name and birth date is not allowed to occur until after confirm presence of exactly one ID band occurs.
4. Before the first confirm presence of exactly one ID band occurs, all the events of secondary interest are allowed to occur zero or more times.
5. confirm presence of exactly one ID band is not required to occur.
6. Even if confirm presence of exactly one ID band does occur, obtain Patient’s stated name and birth date is not required to occur after confirm presence of exactly one ID band occurs.
7. After confirm presence of exactly one ID band occurs, but before the first subsequent obtain Patient’s stated name and birth date occurs:
   - confirm presence of exactly one ID band is allowed to occur again, zero or more times;
   - None of the events of secondary interest are allowed to occur.
8. After confirm presence of exactly one ID band and the first subsequent obtain Patient’s stated name and birth date occur:
   - All the events of secondary interest are allowed to occur zero or more times;
   - Both confirm presence of exactly one ID band and obtain Patient’s stated name and birth date are allowed to occur again, zero or more times, and there are no restrictions on the occurrences of any future events.

**FSA and DNL for Property ED.B.5 (B.1 → B.2)**
Event alphabet:

- A: obtain Patient’s stated name and birth date
- B: verify ID band and Patient’s stated name and birth date match
- END: perform management activity i
- C: verify ID band and physician order match

**SCOPE:**

1. There can be at most one restricted interval in the event sequence and it has an ending delimiter, perform management activity i.
2. The behavior is required to hold from the start of the event sequence through to the first occurrence of perform management activity i, if it ever occurs.
3. perform management activity i is not required to occur and if it never occurs, then the behavior is required to hold throughout the entire event sequence.
4. There are no restrictions imposed on the occurrences of any events after the first occurrence of perform management activity i, if it ever occurs.

**BEHAVIOR:**

1. The events of primary interest in this behavior are obtain Patient’s stated name and birth date and verify ID band and Patient’s stated name and birth date match.
2. The event of secondary interest in this behavior is verify ID band and physician order match.
3. verify ID band and Patient’s stated name and birth date match is not allowed to occur until after obtain Patient’s stated name and birth date occurs.
4. Before the first obtain Patient’s stated name and birth date occurs, verify ID band and physician order match is allowed to occur zero or more times.
5. obtain Patient’s stated name and birth date is not required to occur.
6. Even if obtain Patient’s stated name and birth date does occur, verify ID band and Patient’s stated name and birth date match is not required to occur after obtain Patient’s stated name and birth date occurs.
7. After obtain Patient’s stated name and birth date occurs, but before the first subsequent verify ID band and Patient’s stated name and birth date match occurs:
   - obtain Patient’s stated name and birth date is allowed to occur again, zero or more times;
   - verify ID band and physician order match is not allowed to occur.
8. After obtain Patient’s stated name and birth date and the first subsequent verify ID band and Patient’s stated name and birth date match occur:
   - verify ID band and physician order match is allowed to occur zero or more times;
   - Both obtain Patient’s stated name and birth date and verify ID band and Patient’s stated name and birth date match are allowed to occur again, zero or more times, and there are no restrictions on the occurrences of any future events.

**FSA and DNL for Property ED.B.5 (B.2 → B.3)**
**Event alphabet:**
- A: verify ID band and Patient’s stated name and birth date match
- B: verify ID band and physician order match
- END: perform management activity i

**SCOPE:**
1. There can be at most one restricted interval in the event sequence and it has an ending delimiter, perform management activity i.
2. The behavior is required to hold from the start of the event sequence through to the first occurrence of perform management activity i, if it ever occurs.
3. perform management activity i is not required to occur and if it never occurs, then the behavior is required to hold throughout the entire event sequence.
4. There are no restrictions imposed on the occurrences of any events after the first occurrence of perform management activity i, if it ever occurs.

**BEHAVIOR:**
1. The events of primary interest in this behavior are verify ID band and Patient’s stated name and birth date match and verify ID band and physician order match.
2. There are no events of secondary interest in this behavior.
3. verify ID band and physician order match is not allowed to occur until after verify ID band and Patient’s stated name and birth date match occurs.
4. verify ID band and Patient’s stated name and birth date match is not required to occur.
5. Even if verify ID band and Patient’s stated name and birth date match does occur, verify ID band and physician order match is not required to occur after verify ID band and Patient’s stated name and birth date match occurs.
6. After verify ID band and Patient’s stated name and birth date match occurs, but before the first subsequent verify ID band and physician order match occurs, verify ID band and Patient’s stated name and birth date match is allowed to occur again, zero or more times.
7. After verify ID band and Patient’s stated name and birth date match and the first subsequent verify ID band and physician order match occur:
   - Both verify ID band and Patient’s stated name and birth date match and verify ID band and physician order match are allowed to occur again, zero or more times, and there are no restrictions on the occurrences of any future events.

FSA and DNL for Property ED.B.5 (B.3 → B.4)
**Event alphabet:**
- A: confirm presence of signed consent form
- B: perform management activity for Patient
- START: determine that it is not a life-or-death situation
- END: determine that it is a life-or-death situation

**SCOPE:**
1. A restricted interval in the event sequence can have both a starting delimiter, determine that it is not a life-or-death situation, and an ending delimiter, determine that it is a life-or-death situation.
2. The behavior is required to hold from an occurrence of determine that it is not a life-or-death situation, if it ever occurs, through to the first subsequent occurrence of determine that it is a life-or-death situation, if it ever occurs.
3. If there are multiple occurrences of determine that it is not a life-or-death situation in between them, only the first of those occurrences of determine that it is not a life-or-death situation starts a restricted interval; later occurrences of determine that it is not a life-or-death situation within this restricted interval do not have an effect.
4. determine that it is not a life-or-death situation is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if determine that it is not a life-or-death situation does occur, determine that it is a life-or-death situation is not required to occur subsequently. Even if determine that it is a life-or-death situation does not occur subsequently, the behavior is still required to hold, until the end of the event sequence.
5. There might be many restricted intervals in the event sequence. If determine that it is not a life-or-death situation occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of determine that it is not a life-or-death situation, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

**BEHAVIOR:**
1. The events of primary interest in this behavior are confirm presence of signed consent form and perform management activity for Patient.
2. There are no events of secondary interest in this behavior.
3. perform management activity for Patient is not allowed to occur until after confirm presence of signed consent form occurs.
4. confirm presence of signed consent form is not required to occur.
5. Even if confirm presence of signed consent form does occur, perform management activity for Patient is not required to occur after confirm presence of signed consent form occurs.
6. After confirm presence of signed consent form occurs, but before the first subsequent perform management activity for Patient occurs, confirm presence of signed consent form is allowed to occur again, zero or more times.
7. After confirm presence of signed consent form and the first subsequent perform management activity for Patient occur:
   - Both confirm presence of signed consent form and perform management activity for Patient are allowed to occur again, zero or more times, and there are no restrictions on the occurrences of any future events.

FSA and DNL for Property ED.B.6
Event alphabet:
- A: Patient dies
- B: call coroner
- END: 24 hours pass since Patient admittance to ED

SCOPE:
1. There can be at most one restricted interval in the event sequence and it has an ending delimiter, 24 hours pass since Patient admittance to ED.
2. The behavior is required to hold from the start of the event sequence through to the first occurrence of 24 hours pass since Patient admittance to ED, if it ever occurs.
3. 24 hours pass since Patient admittance to ED is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence.
4. There are no restrictions imposed on the occurrences of any events after the first occurrence of 24 hours pass since Patient admittance to ED, if it ever occurs.

BEHAVIOR:
1. The events of primary interest in this behavior are Patient dies and call coroner.
2. There are no events of secondary interest in this behavior.
3. If Patient dies occurs, call coroner is required to occur subsequently.
4. Before the first Patient dies occurs, call coroner is not allowed to occur.
5. Patient dies is not required to occur.
6. After Patient dies occurs, but before the first subsequent call coroner occurs, Patient dies is not allowed to occur again.
7. After Patient dies and the first subsequent call coroner occur:
   - call coroner is allowed to occur again, zero or more times; Patient dies is never allowed to occur again; Further occurrences of call coroner do not impose additional restrictions on the occurrences of any future events.

FSA and DNL for Property ED.B.10a
Event alphabet:
- A: admit Patient to ED
- B: 24 hours pass since Patient admittance to ED

**SCOPE:**
1. From the start of any event sequence through to the end of that event sequence, the behavior must hold.

**BEHAVIOR:**
1. The events of primary interest in this behavior are admit Patient to ED and 24 hours pass since Patient admittance to ED.
2. There are no events of secondary interest in this behavior.
3. If admit Patient to ED occurs, 24 hours pass since Patient admittance to ED is required to occur subsequently.
4. Before the first admit Patient to ED occurs, 24 hours pass since Patient admittance to ED is not allowed to occur.
5. admit Patient to ED is not required to occur.
6. After admit Patient to ED occurs, but before the first subsequent 24 hours pass since Patient admittance to ED occurs, admit Patient to ED is allowed to occur again, zero or more times.
7. After admit Patient to ED and the first subsequent 24 hours pass since Patient admittance to ED occur:
   - 24 hours pass since Patient admittance to ED is not allowed to occur again until after another admit Patient to ED occurs; admit Patient to ED is allowed to occur again and, if it does, then the situation is the same as when the first admit Patient to ED occurred, meaning that the restrictions described in parts 3, 6, and 7 would again apply.

FSA and DNL for Property ED.B.10b
Event alphabet:
- A: bring Patient to Main ED
- B: place Patient in regular ED bed OR in hall bed

**SCOPE:**
1. From the start of any event sequence through to the end of that event sequence, the behavior must hold.

**BEHAVIOR:**
1. The events of primary interest in this behavior are bring Patient to Main ED and place Patient in regular ED bed OR in hall bed.
2. There are no events of secondary interest in this behavior.
3. If bring Patient to Main ED occurs, place Patient in regular ED bed OR in hall bed is required to occur subsequently.
4. Before the first bring Patient to Main ED occurs, place Patient in regular ED bed OR in hall bed is not allowed to occur.
5. bring Patient to Main ED is not required to occur.
6. After bring Patient to Main ED occurs, but before the first subsequent place Patient in regular ED bed OR in hall bed occurs, bring Patient to Main ED is not allowed to occur again.
7. After bring Patient to Main ED and the first subsequent place Patient in regular ED bed OR in hall bed occur:
   - place Patient in regular ED bed OR in hall bed is allowed to occur again, zero or more times, before another bring Patient to Main ED occurs; bring Patient to Main ED is allowed to occur again and, if it does, then the situation is the same as when the first bring Patient to Main ED occurred, meaning that the restrictions described in parts 3, 6, and 7 would again apply.

FSA and DNL for Property ED.B.7
Event alphabet:
- A: bring Patient into Main ED
- B: write Patient information on patient status board

SCOPE:
1. From the start of any event sequence through to the end of that event sequence, the behavior must hold.

BEHAVIOR:
1. The events of primary interest in this behavior are bring Patient into Main ED and write Patient information on patient status board.
2. There are no events of secondary interest in this behavior.
3. If bring Patient into Main ED occurs, write Patient information on patient status board is required to occur subsequently.
4. Before the first bring Patient into Main ED occurs, write Patient information on patient status board is not allowed to occur.
5. bring Patient into Main ED is not required to occur.
6. After bring Patient into Main ED occurs, but before the first subsequent write Patient information on patient status board occurs, bring Patient into Main ED is not allowed to occur again.
7. After bring Patient into Main ED and the first subsequent write Patient information on patient status board occur:
   - write Patient information on patient status board is not allowed to occur again until after another bring Patient into Main ED occurs; bring Patient into Main ED is allowed to occur again and, if it does, then the situation is the same as when the first bring Patient into Main ED occurred, meaning that the restrictions described in parts 3, 6, and 7 would again apply.

FSA and DNL for Property ED.B.8
Event alphabet:
- A: MD signs physician order
- B: follow physician order

SCOPE:
1. From the start of any event sequence through to the end of that event sequence, the behavior must hold.

BEHAVIOR:
1. The events of primary interest in this behavior are **MD signs physician order** and **follow physician order**.
2. There are no events of secondary interest in this behavior.
3. **follow physician order** is not allowed to occur until after **MD signs physician order** occurs.
4. **MD signs physician order** is not required to occur.
5. Even if **MD signs physician order** does occur, **follow physician order** is not required to occur after **MD signs physician order** occurs.
6. After **MD signs physician order** occurs, but before the first subsequent **follow physician order** occurs, **MD signs physician order** is allowed to occur again, zero or more times.
7. After **MD signs physician order** and the first subsequent **follow physician order** occur:
   - **follow physician order** is not allowed to occur again until after another **MD signs physician order** occurs; **MD signs physician order** is allowed to occur again and, if it does, then the situation is the same as when the first **MD signs physician order** occurred, meaning that the restrictions described in parts 5, 6, and 7 would again apply.
Event alphabet:
- A: admit Patient to ED
- B: complete medical record

SCOPE:
1. From the start of any event sequence through to the end of that event sequence, the behavior must hold.

BEHAVIOR:
1. The events of primary interest in this behavior are admit Patient to ED and complete medical record.
2. There are no events of secondary interest in this behavior.
3. If admit Patient to ED occurs, complete medical record is required to occur subsequently.
4. Before the first admit Patient to ED occurs, complete medical record is not allowed to occur.
5. admit Patient to ED is not required to occur.
6. After admit Patient to ED occurs, but before the first subsequent complete medical record occurs, admit Patient to ED is not allowed to occur again.
7. After admit Patient to ED and the first subsequent complete medical record occur:
   - complete medical record is not allowed to occur again until after another admit Patient to ED occurs; admit Patient to ED is allowed to occur again and, if it does, then the situation is the same as when the first admit Patient to ED occurred, meaning that the restrictions described in parts 3, 6, and 7 would again apply.

FSA and DNL for Property ED.C.2
Event alphabet:
- A: EMT gives report to Nurse assigned to Patient OR to External Triage Nurse
- START: EMT brings Patient to ED
- END: EMT leaves ED

**SCOPE:**
1. A restricted interval in the event sequence can have both a starting delimiter, EMT brings Patient to ED, and an ending delimiter, EMT leaves ED.
2. The behavior is required to hold from an occurrence of EMT brings Patient to ED, if it ever occurs, through to the first subsequent occurrence of EMT leaves ED, if it ever occurs.
3. If there are multiple occurrences of EMT brings Patient to ED without an occurrence of EMT leaves ED in between them, only the first of those occurrences of EMT brings Patient to ED starts a restricted interval; later occurrences of EMT brings Patient to ED within this restricted interval do not have an effect.
4. EMT brings Patient to ED is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if EMT brings Patient to ED does occur, EMT leaves ED is not required to occur subsequently. Even if EMT leaves ED does not occur subsequently, the behavior is still required to hold, until the end of the event sequence.
5. There might be many restricted intervals in the event sequence. If EMT brings Patient to ED occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of EMT brings Patient to ED, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

**BEHAVIOR:**
1. The event of primary interest in this behavior is EMT gives report to Nurse assigned to Patient OR to External Triage Nurse.
2. There are no events of secondary interest in this behavior.
3. EMT gives report to Nurse assigned to Patient OR to External Triage Nurse is required to occur exactly once.

FSA and DNL for Property ED.C.3
Event alphabet:

- A: Patient signs waiver
- START, admit Patient to ED
- END, Patient leaves ED against medical advice

**SCOPE:**

1. A restricted interval in the event sequence can have both a starting delimiter, admit Patient to ED, and an ending delimiter, Patient leaves ED against medical advice.
2. The behavior is required to hold from an occurrence of admit Patient to ED, if it ever occurs, through to the first subsequent occurrence of Patient leaves ED against medical advice, if it ever occurs.
3. If there are multiple occurrences of admit Patient to ED without an occurrence of Patient leaves ED against medical advice in between them, only the first of those occurrences of admit Patient to ED potentially starts a restricted interval; later occurrences of admit Patient to ED within this restricted interval do not have an effect.
4. admit Patient to ED is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if admit Patient to ED does occur, Patient leaves ED against medical advice is not required to occur subsequently and if it does not occur subsequently, then the behavior is not required to hold for the remainder of the event sequence.
5. There might be many restricted intervals in the event sequence. If admit Patient to ED occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of admit Patient to ED, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a potential new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

**BEHAVIOR:**

1. The event of primary interest in this behavior is Patient signs waiver.
2. There are no events of secondary interest in this behavior.
3. Patient signs waiver is required to occur at least once.

FSA and DNL for Property ED.C.4
Event alphabet:
- **A**: Patient completes full registration
- **B**: discharge Patient from ED
- **START**: admit Patient to ED
- **END**: Patient leaves Emergency Department against medical advice OR without notifying a medical professional

**SCOPE:**
1. A restricted interval in the event sequence can have both a starting delimiter, admit Patient to ED, and an ending delimiter, Patient leaves Emergency Department against medical advice OR without notifying a medical professional.
2. The behavior is required to hold from an occurrence of admit Patient to ED, if it ever occurs, through to the first subsequent occurrence of Patient leaves Emergency Department against medical advice OR without notifying a medical professional, if it ever occurs.
3. If there are multiple occurrences of admit Patient to ED without an occurrence of Patient leaves Emergency Department against medical advice OR without notifying a medical professional in between them, only the first of those occurrences of admit Patient to ED starts a restricted interval; later occurrences of admit Patient to ED within this restricted interval do not have an effect.
4. admit Patient to ED is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if admit Patient to ED does occur, Patient leaves Emergency Department against medical advice OR without notifying a medical professional is not required to occur subsequently. Even if Patient leaves Emergency Department against medical advice OR without notifying a medical professional does not occur subsequently, the behavior is still required to hold, until the end of the event sequence.
5. There might be many restricted intervals in the event sequence. If admit Patient to ED occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of admit Patient to ED, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

**BEHAVIOR:**
1. The events of primary interest in this behavior are Patient completes full registration and discharge Patient from ED.
2. There are no events of secondary interest in this behavior.
3. discharge Patient from ED is not allowed to occur until after Patient completes full registration occurs.
4. Patient completes full registration is not required to occur.
5. Even if Patient completes full registration does occur, discharge Patient from ED is not required to occur after Patient completes full registration occurs.
6. After Patient completes full registration occurs, but before the first subsequent discharge Patient from ED occurs, Patient completes full registration is not allowed to occur again.
7. After Patient completes full registration and the first subsequent discharge Patient from ED occur:
   - discharge Patient from ED is not allowed to occur again until after another Patient completes full registration occurs. Patient completes full registration is allowed to occur again and, if it does, then the situation is the same as when the first Patient completes full registration occurred, meaning that the restrictions described in parts 5, 6, and 7 would again apply.

FSA and DNL for Property ED.D.1
Event alphabet:
- A: confirm presence of exactly one ID band
- B: obtain Patient’s stated name and birth date
- START: admit Patient to ED
- END: discharge Patient from ED
- C: verify ID band and discharge order match, verify ID band and Patient’s stated name and birth date match

SCOPE:
1. A restricted interval in the event sequence can have both a starting delimiter, admit Patient to ED, and an ending delimiter, discharge Patient from ED.
2. The behavior is required to hold from an occurrence of admit Patient to ED, if it ever occurs, through to the first subsequent occurrence of discharge Patient from ED, if it ever occurs.
3. If there are multiple occurrences of admit Patient to ED without an occurrence of discharge Patient from ED in between them, only the first of those occurrences of admit Patient to ED starts a restricted interval; later occurrences of admit Patient to ED within this restricted interval do not have an effect.
4. admit Patient to ED is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if admit Patient to ED does occur, discharge Patient from ED is not required to occur subsequently. Even if discharge Patient from ED does not occur subsequently, the behavior is still required to hold, until the end of the event sequence.
5. There might be many restricted intervals in the event sequence. If admit Patient to ED occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of admit Patient to ED, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

BEHAVIOR:
1. The events of primary interest in this behavior are confirm presence of exactly one ID band and obtain Patient’s stated name and birth date.
2. The events of secondary interest in this behavior are verify ID band and discharge order match and verify ID band and Patient’s stated name and birth date match.
3. obtain Patient’s stated name and birth date is not allowed to occur until after confirm presence of exactly one ID band occurs.
4. Before the first confirm presence of exactly one ID band occurs, all the events of secondary interest are allowed to occur zero or more times.
5. confirm presence of exactly one ID band is not required to occur.
6. Even if confirm presence of exactly one ID band occurs, obtain Patient’s stated name and birth date is not required to occur after confirm presence of exactly one ID band occurs.
7. After confirm presence of exactly one ID band occurs, but before the first subsequent obtain Patient’s stated name and birth date occurs:
   - confirm presence of exactly one ID band is allowed to occur again, zero or more times;
   - None of the events of secondary interest are allowed to occur.
8. After confirm presence of exactly one ID band and the first subsequent obtain Patient’s stated name and birth date occur:
   - All the events of secondary interest are allowed to occur zero or more times;
   - Both confirm presence of exactly one ID band and obtain Patient’s stated name and birth date are allowed to occur again, zero or more times, and there are no restrictions on the occurrences of any future events.

FSA and DNL for Property ED.D.10 (D.6 → D.7)
**Event alphabet:**
- A: obtain Patient’s stated name and birth date
- B: verify ID band and Patient’s stated name and birth date match
- START: admit Patient to ED
- END: discharge Patient from ED
- C: verify ID band and discharge order match

**SCOPE:**
1. A restricted interval in the event sequence can have both a starting delimiter, admit Patient to ED, and an ending delimiter, discharge Patient from ED.
2. The behavior is required to hold from an occurrence of admit Patient to ED, if it ever occurs, through to the first subsequent occurrence of discharge Patient from ED, if it ever occurs.
3. If there are multiple occurrences of admit Patient to ED without an occurrence of discharge Patient from ED in between them, only the first of those occurrences of admit Patient to ED starts a restricted interval; later occurrences of admit Patient to ED within this restricted interval do not have an effect.
4. admit Patient to ED is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if admit Patient to ED does occur, discharge Patient from ED is not required to occur subsequently. Even if discharge Patient from ED does not occur subsequently, the behavior is still required to hold, until the end of the event sequence.
5. There might be many restricted intervals in the event sequence. If admit Patient to ED occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of admit Patient to ED, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

**BEHAVIOR:**
1. The events of primary interest in this behavior are obtain Patient’s stated name and birth date and verify ID band and Patient’s stated name and birth date match.
2. The event of secondary interest in this behavior is verify ID band and discharge order match.
3. verify ID band and Patient’s stated name and birth date match is not allowed to occur until after obtain Patient’s stated name and birth date occurs.
4. Before the first obtain Patient’s stated name and birth date occurs, verify ID band and discharge order match is allowed to occur zero or more times.
5. obtain Patient’s stated name and birth date is not required to occur.
6. Even if obtain Patient’s stated name and birth date does occur, verify ID band and Patient’s stated name and birth date match is not required to occur after obtain Patient’s stated name and birth date occurs.
7. After obtain Patient’s stated name and birth date occurs, but before the first subsequent verify ID band and Patient’s stated name and birth date match occurs:
   - obtain Patient’s stated name and birth date is allowed to occur again, zero or more times;
   - verify ID band and discharge order match is not allowed to occur.
8. After obtain Patient’s stated name and birth date and the first subsequent verify ID band and Patient’s stated name and birth date match occur:
   - verify ID band and discharge order match is allowed to occur zero or more times;
   - Both obtain Patient’s stated name and birth date and verify ID band and Patient’s stated name and birth date match are allowed to occur again, zero or more times, and there are no restrictions on the occurrences of any future events.

FSA and DNL for Property ED.D.10 (D.7 → D.8)
Event alphabet:
- **A**: verify ID band and Patient’s stated name and birth date match
- **B**: verify ID band and discharge order match
- **START**: admit Patient to ED
- **END**: discharge Patient from ED

**SCOPE:**
1. A restricted interval in the event sequence can have both a starting delimiter, admit Patient to ED, and an ending delimiter, discharge Patient from ED.
2. The behavior is required to hold from an occurrence of admit Patient to ED, if it ever occurs, through to the first subsequent occurrence of discharge Patient from ED, if it ever occurs.
3. If there are multiple occurrences of admit Patient to ED without an occurrence of discharge Patient from ED in between them, only the first of those occurrences of admit Patient to ED starts a restricted interval; later occurrences of admit Patient to ED within this restricted interval do not have an effect.
4. admit Patient to ED is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if admit Patient to ED does occur, discharge Patient from ED is not required to occur subsequently. Even if discharge Patient from ED does not occur subsequently, the behavior is still required to hold, until the end of the event sequence.
5. There might be many restricted intervals in the event sequence. If admit Patient to ED occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of admit Patient to ED, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

**BEHAVIOR:**
1. The events of primary interest in this behavior are verify ID band and Patient’s stated name and birth date match and verify ID band and discharge order match.
2. There are no events of secondary interest in this behavior.
3. verify ID band and discharge order match is not allowed to occur until after verify ID band and Patient’s stated name and birth date match occurs.
4. verify ID band and Patient’s stated name and birth date match is not required to occur.
5. Even if verify ID band and Patient’s stated name and birth date match does occur, verify ID band and discharge order match is not required to occur after verify ID band and Patient’s stated name and birth date match occurs.
6. After verify ID band and Patient’s stated name and birth date match occurs, but before the first subsequent verify ID band and discharge order match occurs, verify ID band and Patient’s stated name and birth date match is allowed to occur again, zero or more times.
7. After verify ID band and Patient’s stated name and birth date match and the first subsequent verify ID band and discharge order match occurs:
   - Both verify ID band and Patient’s stated name and birth date match and verify ID band and discharge order match are allowed to occur again, zero or more times, and there are no restrictions on the occurrences of any future events.

FSA and DNL for Property ED.D.10 (D.8 → D.9)
Event alphabet:
- A: check Patient’s vital signs
- B: discharge Patient from ED
- START: admit Patient to ED
- END: Patient leaves Emergency Department against medical advice OR without notifying a medical professional
- C: MD evaluates Patient

SCOPE:
1. A restricted interval in the event sequence can have both a starting delimiter, admit Patient to ED, and an ending delimiter, Patient leaves Emergency Department against medical advice OR without notifying a medical professional.
2. The behavior is required to hold from an occurrence of admit Patient to ED, if it ever occurs, through to the first subsequent occurrence of Patient leaves Emergency Department against medical advice OR without notifying a medical professional, if it ever occurs.
3. If there are multiple occurrences of admit Patient to ED without an occurrence of Patient leaves Emergency Department against medical advice OR without notifying a medical professional in between them, only the first of those occurrences of admit Patient to ED starts a restricted interval; later occurrences of admit Patient to ED within this restricted interval do not have an effect.
4. admit Patient to ED is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if admit Patient to ED does occur, Patient leaves Emergency Department against medical advice OR without notifying a medical professional is not required to occur subsequently. Even if Patient leaves Emergency Department against medical advice OR without notifying a medical professional does not occur subsequently, the behavior is still required to hold, until the end of the event sequence.
5. There might be many restricted intervals in the event sequence. If admit Patient to ED occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of admit Patient to ED, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

BEHAVIOR:
1. The events of primary interest in this behavior are check Patient’s vital signs and discharge Patient from ED.
2. The event of secondary interest in this behavior is MD evaluates Patient.
3. discharge Patient from ED is not allowed to occur until after check Patient’s vital signs occurs.
4. Before the first check Patient’s vital signs occurs, MD evaluates Patient is allowed to occur zero or more times.
5. check Patient’s vital signs is not required to occur.
6. Even if check Patient’s vital signs does occur, discharge Patient from ED is not required to occur after check Patient’s vital signs occurs.
7. After check Patient’s vital signs occurs, but before the first subsequent discharge Patient from ED occurs:
   - check Patient’s vital signs is allowed to occur again, zero or more times;
   - MD evaluates Patient is allowed to occur zero or more times.
8. After check Patient’s vital signs and the first subsequent discharge Patient from ED occur:
   - MD evaluates Patient is allowed to occur zero or more times;
   - discharge Patient from ED is not allowed to occur again until after another check Patient’s vital signs occurs; check Patient’s vital signs is allowed to occur again and, if it does, then the situation is the same as when the first check Patient’s vital signs occurred, meaning that the restrictions described in parts 6, 7, and 8 would again apply.

FSA and DNL for Property ED.D.2
Event alphabet:
- A: MD evaluates Patient
- B: discharge Patient from ED
- START: admit Patient to ED
- END: Patient leaves Emergency Department against medical advice OR without notifying a medical professional

**SCOPE:**
1. A restricted interval in the event sequence can have both a starting delimiter, admit Patient to ED, and an ending delimiter, Patient leaves Emergency Department against medical advice OR without notifying a medical professional.
2. The behavior is required to hold from an occurrence of admit Patient to ED, if it ever occurs, through to the first subsequent occurrence of Patient leaves Emergency Department against medical advice OR without notifying a medical professional, if it ever occurs.
3. If there are multiple occurrences of admit Patient to ED within a restricted interval, only the first of those occurrences of admit Patient to ED starts a restricted interval; later occurrences of admit Patient to ED within this restricted interval do not have an effect.
4. admiPnt Patient to ED is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence.
5. If there are multiple occurrences of admit Patient to ED without an occurrence of Patient leaves Emergency Department against medical advice OR without notifying a medical professional in between them, only the first of those occurrences of admit Patient to ED starts a restricted interval; later occurrences of admit Patient to ED within this restricted interval do not have an effect.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

**BEHAVIOR:**
1. The events of primary interest in this behavior are MD evaluates Patient and discharge Patient from ED.
2. There are no events of secondary interest in this behavior.
3. discharge Patient from ED is not allowed to occur until after MD evaluates Patient occurs.
4. MD evaluates Patient is not required to occur.
5. Even if MD evaluates Patient does occur, discharge Patient from ED is not required to occur after MD evaluates Patient occurs.
6. After MD evaluates Patient occurs, but before the first subsequent discharge Patient from ED occurs, MD evaluates Patient is allowed to occur again, zero or more times.
7. After MD evaluates Patient and the first subsequent discharge Patient from ED occur:
   - discharge Patient from ED is not allowed to occur again until after another MD evaluates Patient occurs; MD evaluates Patient is allowed to occur again and, if it does, then the situation is the same as when the first MD evaluates Patient occurred, meaning that the restrictions described in parts 5, 6, and 7 would again apply.

FSA and DNL for Property ED.D.3
Event alphabet:
- A: give Patient discharge instructions
- B: discharge Patient from ED
- START: admit Patient to ED
- END: Patient leaves Emergency Department against medical advice OR without notifying a medical professional

SCOPE:
1. A restricted interval in the event sequence can have both a starting delimiter, admit Patient to ED, and an ending delimiter, Patient leaves Emergency Department against medical advice OR without notifying a medical professional.
2. The behavior is required to hold from an occurrence of admit Patient to ED, if it ever occurs, through to the first subsequent occurrence of Patient leaves Emergency Department against medical advice OR without notifying a medical professional, if it ever occurs.
3. If there are multiple occurrences of admit Patient to ED without an occurrence of Patient leaves Emergency Department against medical advice OR without notifying a medical professional in between them, only the first of those occurrences of admit Patient to ED starts a restricted interval; later occurrences of admit Patient to ED within this restricted interval do not have an effect.
4. admit Patient to ED is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if admit Patient to ED does occur, Patient leaves Emergency Department against medical advice OR without notifying a medical professional is not required to occur subsequently. Even if Patient leaves Emergency Department against medical advice OR without notifying a medical professional does not occur subsequently, the behavior is still required to hold, until the end of the event sequence.
5. There might be many restricted intervals in the event sequence. If admit Patient to ED occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of admit Patient to ED, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

BEHAVIOR:
1. The events of primary interest in this behavior are give Patient discharge instructions and discharge Patient from ED.
2. There are no events of secondary interest in this behavior.
3. discharge Patient from ED is not allowed to occur until after give Patient discharge instructions occurs.
4. give Patient discharge instructions is not required to occur.
5. Even if give Patient discharge instructions does occur, discharge Patient from ED is not required to occur after give Patient discharge instructions occurs.
6. After give Patient discharge instructions occurs, but before the first subsequent discharge Patient from ED occurs, give Patient discharge instructions is allowed to occur again, zero or more times.
7. After give Patient discharge instructions and the first subsequent discharge Patient from ED occur:
   - discharge Patient from ED is not allowed to occur again until after another give Patient discharge instructions occurs; give Patient discharge instructions is allowed to occur again and, if it does, then the situation is the same as when the first give Patient discharge instructions occurred, meaning that the restrictions described in parts 5, 6, and 7 would again apply.

FSA and DNL for Property ED.D.4
Event alphabet:
- A: check Patient vital signs
- B: MD evaluates Patient
- START: admit Patient to ED
- END: discharge Patient from ED
- C: give Patient discharge instructions

**SCOPE:**
1. A restricted interval in the event sequence can have both a starting delimiter, admit Patient to ED, and an ending delimiter, discharge Patient from ED.
2. The behavior is required to hold from an occurrence of admit Patient to ED, if it ever occurs, through to the first subsequent occurrence of discharge Patient from ED, if it ever occurs.
3. If there are multiple occurrences of admit Patient to ED without an occurrence of discharge Patient from ED in between them, only the first of those occurrences of admit Patient to ED starts a restricted interval; later occurrences of admit Patient to ED within this restricted interval do not have an effect.
4. admit Patient to ED is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if admit Patient to ED does occur, discharge Patient from ED is not required to occur subsequently. Even if discharge Patient from ED does not occur subsequently, the behavior is still required to hold, until the end of the event sequence.
5. There might be many restricted intervals in the event sequence. If admit Patient to ED occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of admit Patient to ED, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

**BEHAVIOR:**
1. The events of primary interest in this behavior are check Patient vital signs and MD evaluates Patient.
2. The event of secondary interest in this behavior is give Patient discharge instructions.
3. MD evaluates Patient is not allowed to occur until after check Patient vital signs occurs.
4. Before the first check Patient vital signs occurs, give Patient discharge instructions is allowed to occur zero or more times.
5. check Patient vital signs is not required to occur.
6. Even if check Patient vital signs does occur, MD evaluates Patient is not required to occur after check Patient vital signs occurs.
7. After check Patient vital signs occurs, but before the first subsequent MD evaluates Patient occurs:
   - check Patient vital signs is allowed to occur again, zero or more times;
   - give Patient discharge instructions is not allowed to occur.
8. After check Patient vital signs and the first subsequent MD evaluates Patient occur:
   - give Patient discharge instructions is allowed to occur zero or more times;
   - both check Patient vital signs and MD evaluates Patient are allowed to occur again, zero or more times, and there are no restrictions on the occurrences of any future events.

**FSA and DNL for Property ED.D.5 (D.2 → D.3)**
**Event alphabet:**
- A: MD evaluates Patient
- B: give Patient discharge instructions
- START: admit Patient to ED
- END: discharge Patient from ED
- C: check Patient vital signs

**SCOPE:**
1. A restricted interval in the event sequence can have both a starting delimiter, admit Patient to ED, and an ending delimiter, discharge Patient from ED.
2. The behavior is required to hold from an occurrence of admit Patient to ED, if it ever occurs, through to the first subsequent occurrence of discharge Patient from ED, if it ever occurs.
3. If there are multiple occurrences of admit Patient to ED without an occurrence of discharge Patient from ED in between them, only the first of those occurrences of admit Patient to ED starts a restricted interval; later occurrences of admit Patient to ED within this restricted interval do not have an effect.
4. admit Patient to ED is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if admit Patient to ED does occur, discharge Patient from ED is not required to occur subsequently. Even if discharge Patient from ED does not occur subsequently, the behavior is still required to hold, until the end of the event sequence.
5. There might be many restricted intervals in the event sequence. If admit Patient to ED occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of admit Patient to ED, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

**BEHAVIOR:**
1. The events of primary interest in this behavior are MD evaluates Patient and give Patient discharge instructions.
2. The event of secondary interest in this behavior is check Patient vital signs.
3. give Patient discharge instructions is not allowed to occur until after MD evaluates Patient occurs.
4. Before the first MD evaluates Patient occurs, check Patient vital signs is allowed to occur zero or more times.
5. MD evaluates Patient is not required to occur.
6. Even if MD evaluates Patient does occur, give Patient discharge instructions is not required to occur after MD evaluates Patient occurs.
7. After MD evaluates Patient occurs, but before the first subsequent give Patient discharge instructions occurs:
   - MD evaluates Patient is allowed to occur again, zero or more times;
   - check Patient vital signs is allowed to occur zero or more times.
8. After MD evaluates Patient and the first subsequent give Patient discharge instructions occur:
   - check Patient vital signs is allowed to occur zero or more times;
   - Both MD evaluates Patient and give Patient discharge instructions are allowed to occur again, zero or more times, and there are no restrictions on the occurrences of any future events.

FSA and DNL for Property ED.D.5 (D.3 → D.4)
Event alphabet:
- A: confirm presence of exactly one ID band
- B: discharge Patient from ED
- START: admit Patient to ED
- END: Patient leaves Emergency Department against medical advice OR without notifying a medical professional
- C: obtain Patient’s stated name and birth date, verify ID band and discharge order match, verify ID band and Patient’s stated name and birth date match

**SCOPE:**
1. A restricted interval in the event sequence can have both a starting delimiter, admit Patient to ED, and an ending delimiter, Patient leaves Emergency Department against medical advice OR without notifying a medical professional.
2. The behavior is required to hold from an occurrence of admit Patient to ED, if it ever occurs, through to the first subsequent occurrence of Patient leaves Emergency Department against medical advice OR without notifying a medical professional, if it ever occurs.
3. If there are multiple occurrences of admit Patient to ED without an occurrence of Patient leaves Emergency Department against medical advice OR without notifying a medical professional in between them, only the first of those occurrences of admit Patient to ED starts a restricted interval; later occurrences of admit Patient to ED within this restricted interval do not have an effect.
4. admit Patient to ED is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if admit Patient to ED does occur, Patient leaves Emergency Department against medical advice OR without notifying a medical professional is not required to occur subsequently. Even if Patient leaves Emergency Department against medical advice OR without notifying a medical professional does not occur subsequently, the behavior is still required to hold, until the end of the event sequence.
5. There might be many restricted intervals in the event sequence. If admit Patient to ED occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of admit Patient to ED, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

**BEHAVIOR:**
1. The events of primary interest in this behavior are confirm presence of exactly one ID band and discharge Patient from ED.
2. The events of secondary interest in this behavior are obtain Patient’s stated name and birth date, verify ID band and discharge order match, and verify ID band and Patient’s stated name and birth date match.
3. discharge Patient from ED is not allowed to occur until after confirm presence of exactly one ID band occurs.
4. Before the first confirm presence of exactly one ID band occurs, all the events of secondary interest are allowed to occur zero or more times.
5. confirm presence of exactly one ID band is not required to occur.
6. Even if confirm presence of exactly one ID band does occur, discharge Patient from ED is not required to occur after confirm presence of exactly one ID band occurs.
7. After confirm presence of exactly one ID band occurs, but before the first subsequent discharge Patient from ED occurs:
   - confirm presence of exactly one ID band is allowed to occur again, zero or more times;
   - All the events of secondary interest are allowed to occur zero or more times.
8. After confirm presence of exactly one ID band and the first subsequent discharge Patient from ED occur:
   - All the events of secondary interest are allowed to occur zero or more times;
   - discharge Patient from ED is not allowed to occur again until after another confirm presence of exactly one ID band occurs;
   - confirm presence of exactly one ID band is allowed to occur again and, if it does, then the situation is the same as when the first confirm presence of exactly one ID band occurred, meaning that the restrictions described in parts 6, 7, and 8 would again apply.

FSA and DNL for Property ED.D.6
Event alphabet:
- **A**: verify ID band and discharge order match
- **B**: discharge Patient from ED
- **END**: Patient leaves Emergency Department against medical advice OR without notifying a medical professional
- **C**: confirm presence of exactly one ID band, obtain Patient’s stated name and birth date, verify ID band and Patient’s stated name and birth date match

**SCOPE:**
1. A restricted interval in the event sequence can have both a starting delimiter, admit Patient to ED, and an ending delimiter, Patient leaves Emergency Department against medical advice OR without notifying a medical professional.
2. The behavior is required to hold from an occurrence of admit Patient to ED, if it ever occurs, through to the first subsequent occurrence of Patient leaves Emergency Department against medical advice OR without notifying a medical professional, if it ever occurs.
3. If there are multiple occurrences of admit Patient to ED without an occurrence of Patient leaves Emergency Department against medical advice OR without notifying a medical professional in between them, only the first of those occurrences of admit Patient to ED starts a restricted interval; later occurrences of admit Patient to ED within this restricted interval do not have an effect.
4. admit Patient to ED is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence.
5. if admit Patient to ED does occur, Patient leaves Emergency Department against medical advice OR without notifying a medical professional is not required to occur subsequently. Even if Patient leaves Emergency Department against medical advice OR without notifying a medical professional does not occur subsequently, the behavior is still required to hold, until the end of the event sequence.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

**BEHAVIOR:**
1. The events of primary interest in this behavior are verify ID band and discharge order match and discharge Patient from ED.
2. The events of secondary interest in this behavior are confirm presence of exactly one ID band, obtain Patient’s stated name and birth date, and verify ID band and Patient’s stated name and birth date match.
3. discharge Patient from ED is not allowed to occur until after verify ID band and discharge order match occurs.
4. Before the first verify ID band and discharge order match occurs, all the events of secondary interest are allowed to occur zero or more times.
5. verify ID band and discharge order match is not required to occur.
6. Even if verify ID band and discharge order match does occur, discharge Patient from ED is not required to occur after verify ID band and discharge order match occurs.
7. After verify ID band and discharge order match occurs, but before the first subsequent discharge Patient from ED occurs:
   - verify ID band and discharge order match is allowed to occur again, zero or more times;
   - All the events of secondary interest are allowed to occur zero or more times.
8. After verify ID band and discharge order match and the first subsequent discharge Patient from ED occur:
   - All the events of secondary interest are allowed to occur zero or more times;
   - discharge Patient from ED is not allowed to occur again until after another verify ID band and discharge order match occurs;
   - verify ID band and discharge order match is allowed to occur again and, if it does, then the situation is the same as when the first verify ID band and discharge order match occurred, meaning that the restrictions described in parts 6, 7, and 8 would again apply.

FSA and DNL for Property ED.D.9
C.4 Verification of Patient ID

C.4.1 Description of the Domain

The Verification of Patient ID (VPID) case study is part of a larger medical safety project that our research group is participating in. In this case study, the domain expert was a Nursing Ph.D, and we elicited all of the properties through interviews with her. Although we, the computer scientists, were primarily the ones who worked with PROPEL to formally specify the properties, the domain expert used the DNL template property view to specify 1-2 of them herself, with us observing in those cases. In addition, she worked with us to vet all of the informal and formal property specifications against her domain expertise. This domain expert also worked with us on the BT case study (see Appendix C.1) and she did not have any prior knowledge of property specification formalisms or RE.

Due to resource limitations, we focused on a subset of the VPID domain, specifically covering only clinical situations, and all properties in this case study are assumed to be in the context of one patient, one or more ID bands, and one nurse. In addition, because the verification of patient ID is an essential precondition for any medical procedure, the property specifications in this study are partial, in the sense that events describing any medical procedure can be inserted into the VPID property specifications where necessary.

There are 5 informal property specifications in this case study, and they are given in Appendix C.4.2. See Section 5.4.3.2 for an explanation of the notation used in the informal specifications. Appendix C.4.3 gives the glossary for all of the bolded terms used in the informal specifications; see Section 5.4.3.1 for an explanation of the glossary structure. There are 7 formal property specifications in this case study, and they are given in Appendix C.4.4. The FSA and DNL specifications are given for each property, with some modifications made to these two views to enable the property specifications to fit on a single page. The FSA specification is shown in two parts: at the top is a mapping from parameter names to specifier-specified events, labeled “Event alphabet:”, and directly below that mapping is an FSA specification of the property. The FSAs are assumed to be total and deterministic\(^4\) and its transitions are labeled with the parameters, rather than with the specifier-specified events. The DNL specification is also shown in two parts, the scope and the behavior, but a third part, the preamble, is elided. The preamble is the same for every DNL property specification, and can be found in Section 3.2.4.

\(^4\)For brevity, we do not show the transitions that go to a non-accepting trap state. When no transition is provided that explicitly allows an event to occur, it should be assumed that an occurrence of that event puts the FSA into a non-accepting trap state.
C.4.2 Organization of the Informal Property Specifications

Verification of Patient ID Properties

Before each assessment of or intervention for a Patient, the following activities must be performed:

To summarize: that Patient’s ID band must be verified by performing the following activities.

- **MP A.1** ● it must be confirmed that that Patient has exactly one institutional ID band.
- **MP A.2** ● that Patient’s stated first and last name and birth date must be obtained.
- **MP A.3** ● it must be made sure that the first and last name and birth date on that Patient’s ID band match that Patient’s stated first and last name and birth date.
- **MP A.4** ● it must be made sure that the first and last name and medical record number on that Patient’s ID band match the first and last name and medical record number on the physician order.
- **-- A.5.k** ● Optional additional activities, included as necessary for each particular intervention. (See below for examples.)
- **MP A.k+1** ● these activities described above must be performed in that order.

**Example additional activities for particular interventions:**

**Obtaining a specimen:**

- **A.5** ● it must be made sure that the first and last name and medical record number on that Patient’s ID band match the first and last name and medical record number on the specimen container label.

**Administrating an infusion of a unit of blood product or handling a suspected transfusion reaction:**

- **A.5** ● all the information for that unit of blood product must be verified.
  All of this blood product information must be verified by two separate nurses for A.5 to be satisfied. The cases enumerate what information for the unit of blood product must be verified.
- **A.5.1** ● it must be made sure that the first and last name and medical record number on that Patient’s ID band match the first and last name and medical record number on the tag affixed to the unit of blood product.
- **A.5.2** ● it must be made sure that the blood type and blood product unit number on the tag affixed to the unit of blood product match the blood type and blood product unit number on the unit of blood product.
- **A.5.3** ● it must be made sure that the expiration date on the unit of blood product has not been exceeded.
- **A.5.4** ● these verifications described above must occur in that order.
- **A.6** ● a baseline single-unit assessment must be done for that Patient to make sure that that Patient is well enough to receive an infusion.

**Administering chemotherapy**

See section D in the Chemotherapy Properties. Ordering on the activities has not yet been clearly established, so they are not yet included here.

---

Figure C.29. Verification of Patient ID Case Study Informal Specifications
## C.4.3 Glossary

### Verification of Patient ID Property Glossary

<table>
<thead>
<tr>
<th>Term</th>
<th>Definition</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>assessment of Patient</td>
<td></td>
<td></td>
</tr>
<tr>
<td>first and last name and birth date</td>
<td>be on Patient ID band</td>
<td></td>
</tr>
<tr>
<td></td>
<td>be stated</td>
<td></td>
</tr>
<tr>
<td>make sure match between Patient ID band</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>and stated information</td>
<td></td>
</tr>
<tr>
<td>make sure match between Patient ID band</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>and physician order</td>
<td></td>
</tr>
</tbody>
</table>
| obtain                                    | Get access to a Patient’s first and last name and birth date by some means that is independent of that Patient’s record in the medical facility. | Possible means of obtaining a Patient’s first and last name and birth date, in order of most to least preferred:  
1. Ask Patient to state their first and last name and birth date.  
2. Ask Patient’s family (or legal guardian) to state their first and last name and birth date.  
3. Go hunting through Patient’s belongings to find first and last name and birth date. (This is generally frowned upon.)  
| state                                     |                                                                           |                                                                                                                                         |
| ID band                                   | Contains a Patient’s first and last name, birth date, and medical record number. |                                                                                                                                         |
| be institutional                          |                                                                           | ‘Institutional’ is a flexible designation based on context. A Patient cannot have an ID band from another hospital or from a nursing home on, for example. |
| be verified                               | It has been verified that a Patient has exactly one institutional ID band and that that Patient’s stated first and last name and birth date match that Patient’s first and last name and birth date on their ID band(s). |                                                                                                                                         |
| confirm presence of exactly one on Patient|                                                                           |                                                                                                                                         |
| intervention for Patient                  |                                                                           |                                                                                                                                         |
| medical record number                     |                                                                           |                                                                                                                                         |
| make sure match between physician order   |                                                                           |                                                                                                                                         |
|                                           | and Patient ID band                                                      |                                                                                                                                         |
| Patient                                   |                                                                           |                                                                                                                                         |
| be assessed                               |                                                                           |                                                                                                                                         |
| have exactly one ID band                  |                                                                           |                                                                                                                                         |
| obtain stated first and last name and birth date from |                                                                           |                                                                                                                                         |
| physician order                           |                                                                           |                                                                                                                                         |
| make sure match between first and last name and medical record number on Patient ID band and first and last name and medical record number on order |                                                                           |                                                                                                                                         |

**Figure C.30. Verification of Patient ID Case Study Glossary**
C.4.4 Formal Property Specifications

**Event alphabet:**
- A: confirm presence of exactly one ID band
- B: start assessment of OR intervention for Patient

**SCOPE:**
1. From the start of any event sequence through to the end of that event sequence, the behavior must hold.

**BEHAVIOR:**
1. The events of primary interest in this behavior are confirm presence of exactly one ID band and start assessment of OR intervention for Patient.
2. There are no events of secondary interest in this behavior.
3. start assessment of OR intervention for Patient is not allowed to occur until after confirm presence of exactly one ID band occurs.
4. confirm presence of exactly one ID band is not required to occur.
5. Even if confirm presence of exactly one ID band does occur, start assessment of OR intervention for Patient is not required to occur after confirm presence of exactly one ID band occurs.
6. After confirm presence of exactly one ID band occurs, but before the first subsequent start assessment of OR intervention for Patient occurs, confirm presence of exactly one ID band is allowed to occur again, zero or more times.
7. After confirm presence of exactly one ID band and the first subsequent start assessment of OR intervention for Patient occur:
   - start assessment of OR intervention for Patient is not allowed to occur again until after another confirm presence of exactly one ID band occurs; confirm presence of exactly one ID band is allowed to occur again and, if it does, then the situation is the same as when the first confirm presence of exactly one ID band occurred, meaning that the restrictions described in parts 5, 6, and 7 would again apply.

**FSA and DNL for Property VPID.A.1**
Event alphabet:
- A: obtain Patient’s stated name and birth date
- B: start assessment of OR intervention for Patient

SCOPE:
1. From the start of any event sequence through to the end of that event sequence, the behavior must hold.

BEHAVIOR:
1. The events of primary interest in this behavior are **obtain Patient’s stated name and birth date** and **start assessment of OR intervention for Patient**.
2. There are no events of secondary interest in this behavior.
3. **start assessment of OR intervention for Patient** is not allowed to occur until after **obtain Patient’s stated name and birth date** occurs.
4. **obtain Patient’s stated name and birth date** is not required to occur.
5. Even if **obtain Patient’s stated name and birth date** does occur, **start assessment of OR intervention for Patient** is not required to occur after it occurs.
6. After **obtain Patient’s stated name and birth date** occurs, but before the first subsequent **start assessment of OR intervention for Patient** occurs, **obtain Patient’s stated name and birth date** is allowed to occur again, zero or more times.
7. After **obtain Patient’s stated name and birth date** and the first subsequent **start assessment of OR intervention for Patient** occur:
   - **start assessment of OR intervention for Patient** is not allowed to occur again until after another **obtain Patient’s stated name and birth date** occurs; **obtain Patient’s stated name and birth date** is allowed to occur again and, if it does, then the situation is the same as when the first **obtain Patient’s stated name and birth date** occurred, meaning that the restrictions described in parts 5, 6, and 7 would again apply.

FSA and DNL for Property VPID.A.2
Event alphabet:
  • A: verify ID band and Patient’s stated name and birth date match
  • B: start assessment of OR intervention for Patient

SCOPE:
1. From the start of any event sequence through to the end of that event sequence, the behavior must hold.

BEHAVIOR:
1. The events of primary interest in this behavior are verify ID band and Patient’s stated name and birth date match and start assessment of OR intervention for Patient.
2. There are no events of secondary interest in this behavior.
3. start assessment of OR intervention for Patient is not allowed to occur until after verify ID band and Patient’s stated name and birth date match occurs.
4. verify ID band and Patient’s stated name and birth date match is not required to occur.
5. Even if verify ID band and Patient’s stated name and birth date match does occur, start assessment of OR intervention for Patient is not required to occur after verify ID band and Patient’s stated name and birth date match occurs.
6. After verify ID band and Patient’s stated name and birth date match occurs, verify ID band and Patient’s stated name and birth date match is allowed to occur again, zero or more times.
7. After verify ID band and Patient’s stated name and birth date match and the first subsequent start assessment of OR intervention for Patient occur:
   • start assessment of OR intervention for Patient is not allowed to occur again until after another verify ID band and Patient’s stated name and birth date match occurs, verify ID band and Patient’s stated name and birth date match is allowed to occur again and, if it does, then the situation is the same as when the first verify ID band and Patient’s stated name and birth date match occurred, meaning that the restrictions described in parts 5, 6, and 7 would again apply.

FSA and DNL for Property VPID.A.3
Event alphabet:
- A: verify ID band and physician order for the intervention match
- B: start assessment of OR intervention for Patient

SCOPE:
1. From the start of any event sequence through to the end of that event sequence, the behavior must hold.

BEHAVIOR:
1. The events of primary interest in this behavior are verify ID band and physician order for the intervention match and start assessment of OR intervention for Patient.
2. There are no events of secondary interest in this behavior.
3. start assessment of OR intervention for Patient is not allowed to occur until after verify ID band and physician order for the intervention match occurs.
4. verify ID band and physician order for the intervention match is not required to occur.
5. Even if verify ID band and physician order for the intervention match does occur, start assessment of OR intervention for Patient is not required to occur after verify ID band and physician order for the intervention match occurs.
6. After verify ID band and physician order for the intervention match occurs, but before the first subsequent start assessment of OR intervention for Patient occurs, verify ID band and physician order for the intervention match is allowed to occur again, zero or more times.
7. After verify ID band and physician order for the intervention match and the first subsequent start assessment of OR intervention for Patient occur:
   - start assessment of OR intervention for Patient is not allowed to occur again until after another verify ID band and physician order for the intervention match occurs; verify ID band and physician order for the intervention match is allowed to occur again and, if it does, then the situation is the same as when the first verify ID band and physician order for the intervention match occurred, meaning that the restrictions described in parts 5, 6, and 7 would again apply.

FSA and DNL for Property VPID.A.4
Event alphabet:

- A: confirm presence of ID band
- B: obtain Patient’s stated name and birth date
- START: begin preparations for intervention episode
- END: complete intervention episode
- C: A.5..k *, start assessment of OR intervention for Patient, verify ID band and Patient’s stated name and birth date match, verify ID band and physician order for the intervention match

SCOPE:
1. A restricted interval in the event sequence can have both a starting delimiter, begin preparations for intervention episode, and an ending delimiter, complete intervention episode.
2. The behavior is required to hold from an occurrence of begin preparations for intervention episode, if it ever occurs, through to the first subsequent occurrence of complete intervention episode, if it ever occurs.
3. If there are multiple occurrences of begin preparations for intervention episode without an occurrence of complete intervention episode in between them, only the first of those occurrences of begin preparations for intervention episode potentially starts a restricted interval; later occurrences of begin preparations for intervention episode within this restricted interval do not have an effect.
4. begin preparations for intervention episode is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if begin preparations for intervention episode does occur, complete intervention episode is not required to occur subsequently and if it does not occur subsequently, then the behavior is not required to hold for the remainder of the event sequence.
5. There might be many restricted intervals in the event sequence. If begin preparations for intervention episode occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of begin preparations for intervention episode, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a potential new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

BEHAVIOR:
1. The events of primary interest in this behavior are confirm presence of ID band and obtain Patient’s stated name and birth date.
2. The events of secondary interest in this behavior are A.5..k *, start assessment of OR intervention for Patient, verify ID band and Patient’s stated name and birth date match, and verify ID band and physician order for the intervention match.
3. obtain Patient’s stated name and birth date is not allowed to occur until after confirm presence of ID band occurs.
4. Before the first confirm presence of ID band occurs, all the events of secondary interest are allowed to occur zero or more times.
5. confirm presence of ID band is not required to occur.
6. Even if confirm presence of ID band does occur, obtain Patient’s stated name and birth date is not required to occur after confirm presence of ID band occurs.
7. confirm presence of ID band occurs, but before the first subsequent obtain Patient’s stated name and birth date occurs:
   - confirm presence of ID band is allowed to occur again, zero or more times;
   - None of the events of secondary interest are allowed to occur.
8. After confirm presence of ID band and the first subsequent obtain Patient’s stated name and birth date occur:
   - All the events of secondary interest are allowed to occur zero or more times;
   - Both confirm presence of ID band and obtain Patient’s stated name and birth date are allowed to occur again, zero or more times, and there are no restrictions on the occurrences of any future events.

FSA and DNL for Property VPID.A.k+1 (A.1 → A.2)
**Event alphabet:**
- A: obtain Patient’s stated name and birth date
- B: verify ID band and Patient’s stated name and birth date match
- START: begin preparations for intervention episode
- END: complete intervention episode
- C: A.5..k *, confirm presence of ID band, start assessment of OR intervention for Patient, verify ID band and physician order for the invention match

**SCOPE:**
1. A restricted interval in the event sequence can have both a starting delimiter, begin preparations for intervention episode, and an ending delimiter, complete intervention episode.
2. The behavior is required to hold from an occurrence of begin preparations for intervention episode, if it ever occurs, through to the first subsequent occurrence of complete intervention episode, if it ever occurs.
3. If there are multiple occurrences of begin preparations for intervention episode without an occurrence of complete intervention episode in between them, only the first of those occurrences of begin preparations for intervention episode potentially starts a restricted interval; later occurrences of begin preparations for intervention episode within this restricted interval do not have an effect.
4. begin preparations for intervention episode is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if begin preparations for intervention episode does occur, complete intervention episode is not required to occur subsequently and if it does not occur subsequently, then the behavior is not required to hold for the remainder of the event sequence.
5. There might be many restricted intervals in the event sequence. If begin preparations for intervention episode occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of begin preparations for intervention episode, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a potential new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

**BEHAVIOR:**
1. The events of primary interest in this behavior are obtain Patient’s stated name and birth date and verify ID band and Patient’s stated name and birth date match.
2. The events of secondary interest in this behavior are A.5..k *, confirm presence of ID band, start assessment of OR intervention for Patient, and verify ID band and physician order for the invention match.
3. verify ID band and Patient’s stated name and birth date match is not allowed to occur until after obtain Patient’s stated name and birth date occurs.
4. Before the first obtain Patient’s stated name and birth date occurs, all the events of secondary interest are allowed to occur zero or more times.
5. obtain Patient’s stated name and birth date is not required to occur. Even if obtain Patient’s stated name and birth date does occur, verify ID band and Patient’s stated name and birth date match is not required to occur after obtain Patient’s stated name and birth date occurs.
6. After obtain Patient’s stated name and birth date occurs, but before the first subsequent verify ID band and Patient’s stated name and birth date match occurs:
   - obtain Patient’s stated name and birth date is allowed to occur again, zero or more times;
   - None of the events of secondary interest are allowed to occur.
7. After obtain Patient’s stated name and birth date and the first subsequent verify ID band and Patient’s stated name and birth date match occur:
   - All the events of secondary interest are allowed to occur zero or more times;
   - Both obtain Patient’s stated name and birth date and verify ID band and Patient’s stated name and birth date match are allowed to occur again, zero or more times, and there are no restrictions on the occurrences of any future events.

**FSA and DNL for Property VPID.A.k+1 (A.2 → A.3)**
Event alphabet:
- A: verify ID band and Patient’s stated name and birth date match
- B: verify ID band and physician order for the invention match
- START: begin preparations for intervention episode
- END: complete intervention episode
- C: A.5..k *, confirm presence of ID band, obtain Patient’s stated name and birth date, start assessment of OR intervention for Patient

SCOPE:
1. A restricted interval in the event sequence can have both a starting delimiter, begin preparations for intervention episode, and an ending delimiter, complete intervention episode.
2. The behavior is required to hold from an occurrence of begin preparations for intervention episode, if it ever occurs, through to the first subsequent occurrence of complete intervention episode, if it ever occurs.
3. If there are multiple occurrences of begin preparations for intervention episode without an occurrence of complete intervention episode in between them, only the first of those occurrences of begin preparations for intervention episode potentially starts a restricted interval; later occurrences of begin preparations for intervention episode within this restricted interval do not have an effect.
4. begin preparations for intervention episode is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if begin preparations for intervention episode does occur, complete intervention episode is not required to occur subsequently and if it does not occur subsequently, then the behavior is not required to hold for the remainder of the event sequence.
5. There might be many restricted intervals in the event sequence. If begin preparations for intervention episode occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of begin preparations for intervention episode, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a potential new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

BEHAVIOR:
1. The events of primary interest in this behavior are verify ID band and Patient’s stated name and birth date match and verify ID band and physician order for the invention match.
2. The events of secondary interest in this behavior are A.5..k *, confirm presence of ID band, obtain Patient’s stated name and birth date, and start assessment of OR intervention for Patient.
3. verify ID band and physician order for the invention match is not allowed to occur until after verify ID band and Patient’s stated name and birth date match occurs.
4. Before the first verify ID band and Patient’s stated name and birth date match occurs, all the events of secondary interest are allowed to occur zero or more times.
5. verify ID band and Patient’s stated name and birth date match is not required to occur.
6. Even if verify ID band and Patient’s stated name and birth date match does occur, verify ID band and physician order for the invention match is not required to occur after verify ID band and Patient’s stated name and birth date match occurs.
7. After verify ID band and Patient’s stated name and birth date match occurs, but before the first subsequent verify ID band and physician order for the invention match occurs:
   - verify ID band and Patient’s stated name and birth date match is allowed to occur again, zero or more times;
   - None of the events of secondary interest are allowed to occur.
8. After verify ID band and Patient’s stated name and birth date match and the first subsequent verify ID band and physician order for the invention match occurs:
   - All the events of secondary interest are allowed to occur zero or more times;
   - Both verify ID band and Patient’s stated name and birth date match and verify ID band and physician order for the invention match are allowed to occur again, zero or more times, and there are no restrictions on the occurrences of any future events.

FSA and DNL for Property VPID.A.k+1 (A.3 → A.4)
C.5 Defense Blood Standard System

C.5.1 Description of the Domain

The Defense Blood Standard System (DBSS) case study was part of a larger software development project to improve a Department of Defense system that manages a national blood repository by tracking each unit of blood product from its donation to its eventual transfusion or disposal. In this case study, the domain experts were two blood bank laboratory technicians and one of the software development managers who was involved in the design of the DBSS software. Although we, the computer scientists, were primarily the ones who worked with PROPEL to formally specify the properties, the software development manager specified 3 of them herself using the QT and FSA template property views, with us observing in those cases. Due to our limited access to these domain experts, they have not vetted any of the informal and formal property specifications, so the specifications are considered to be in a draft stage. With the possible exception of the software development manager, these domain experts did not have any prior knowledge of property specification formalisms or RE.

Due to resource limitations, we focused on a subset of the domain that concerned how laboratory technicians should perform a type, screen, and crossmatch to identify units of donor blood that are compatible with a patient’s blood specimen. All the properties that were elucidated in this case study were assumed to be in the context of one patient; one physician or transfusion pathologist’s order for a type, screen, and crossmatch, where this order could request one or more units of blood product; one or more blood bank laboratory technicians; one or more units of blood product; and one instance of the DBSS software.

There are 11 informal property specifications in this case study, and they are given in Appendix C.5.2. See Section 5.4.3.2 for an explanation of the notation used in the informal specifications. Appendix C.5.3 gives the glossary for all of the bolded terms used in the informal specifications; see Section 5.4.3.1 for an explanation of the glossary structure. There are 22 formal property specifications in this case study, and they are given in Appendix C.5.4. The FSA and DNL specifications are given for each property, with some modifications made to these two views to enable the property specifications to fit on a single page. The FSA specification is shown in two parts: at the top is a mapping from parameter names to specifier-specified events, labeled “Event alphabet:”, and directly below that mapping is an FSA specification of the property. The FSA is assumed to be total and deterministic and its transitions are labeled with the parameters, rather than with the specifier-specified events. The DNL specification is also shown in two parts, the scope and

\footnote{For brevity, we do not show the transitions that go to a non-accepting trap state. When no transition is provided that explicitly allows an event to occur, it should be assumed that an occurrence of that event puts the FSA into a non-accepting trap state.}
the behavior, but a third part, the preamble, is elided. The preamble is the same for every DNL property specification, and can be found in Section 3.2.4.
C.5.2 Organization of the Informal Property Specifications

Defense Blood Standard System Properties

A. CONSTRAINTS ON EVERY CROSSMATCH PROCESS

Before a unit of blood product can be assigned to a Patient:

7 A.1 an ABO / Rh identification and an antibody screen must be performed on an up-to-date blood specimen from that Patient.

Origin: [6] 3.6C.P003

7 A.2 that unit of blood product must be identified as compatible with that Patient's up-to-date blood specimen.

7 A.3 DBSS must generate the ABO / Rh identification and an antibody screen paperwork if the order is for a type and screen or for a type, screen, and crossmatch.

Origin: [4] 3.6C.P003

B. RESPONSES TO BLOOD SPECIMEN TESTING RESULTS

7 B.1 If a Patient's up-to-date blood specimen has a negative antibody screen, if that Patient does not have a history of clinically-significant red cell antibodies or of ABO / Rh incompatibility, and if the immediate-spin ABO / Rh compatibility testing for a Patient's up-to-date blood specimen and a potentially-compatible unit of blood product produces a positive result, only then can that potentially-compatible unit of blood product be identified as compatible without need of further testing.

Origin: [5] IX.H.8

If a Patient's blood specimen has a positive antibody screen, if that Patient has a history of clinically-significant red cell antibodies or of ABO / Rh incompatibility, or if that Patient's history is unknown:

7 B.2 then if that Patient's blood specimen is more than 72 hours old, an up-to-date blood specimen must be obtained from that Patient before an ABO / Rh identification and antibody screen can be performed for that Patient.


7 B.3 full ABO / Rh and antibody compatibility testing must be performed to identify compatible units of blood product and to ensure availability of compatible units of blood product, whether the order is for a type and screen or for a type, screen, and crossmatch.

Origin: [5] LB note

7 B.4 After choosing a potentially-compatible unit of blood product, if there is evidence of an ABO / Rh or antibody incompatibility between that unit of blood product and a Patient's up-to-date blood specimen, an AHG and IgG phase crossmatch must produce a positive result before that unit of blood product can be identified as compatible.


C. CONSTRAINTS FOR A LIST OF POTENTIALLY-COMPATIBLE UNITS OF BLOOD PRODUCT

7 C.1 DBSS cannot provide a list of potentially-compatible units of blood product for a Patient until after an ABO / Rh identification and an antibody screen is performed on an up-to-date blood specimen from that Patient.

Origin: [6] 3, LCDR Alfonso

7 C.2 DBSS can only put units of blood product that are ABO / Rh - compatible with a Patient's up-to-date blood specimen on the list of potentially-compatible units of blood product for that Patient.

7 C.3 If a Patient's blood specimen has a positive antibody screen, if that Patient has a history of clinically-significant red cell antibodies, or if that Patient's history is unknown, antibody identification must be performed on an up-to-date blood specimen before DBSS can provide a list of potentially-compatible units of blood product, unless an emergency supervisor override occurs.


Figure C.31. Defense Blood Standard System Case Study Informal Specifications - Page 1
C. CONSTRAINTS FOR A LIST OF POTENTIALLY-COMPATIBLE UNITS OF BLOOD PRODUCT

C.3 If a Patient’s blood specimen has a positive antibody screen or if that Patient has a history of clinically-significant red cell antibodies, DBSS can only put units of blood product that lack the corresponding antigens on the list of potentially-compatible units of blood product.


These properties are based on:

[2] AABB, “Standards for Blood Banks and Transfusion Services, 23rd edition”, sections 5.15.1 and 5.15.2

---

Figure C.32. Defense Blood Standard System Case Study Informal Specifications - Page 2
### C.5.3 Glossary

**Defense Blood Standard System Property Glossary**

<table>
<thead>
<tr>
<th>Term</th>
<th>Definition</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>ABO / Rh identification</strong></td>
<td>perform on an up-to-date blood specimen</td>
<td></td>
</tr>
<tr>
<td><strong>ABO / Rh incompatibility</strong></td>
<td>have evidence of</td>
<td></td>
</tr>
<tr>
<td><strong>AHG and IgG phase crossmatch</strong></td>
<td>produce positive result</td>
<td>The crossmatch identifies a unit of blood product that is compatible with an up-to-date blood specimen.</td>
</tr>
<tr>
<td><strong>Antibody identification</strong></td>
<td>perform on an up-to-date blood specimen</td>
<td></td>
</tr>
<tr>
<td><strong>Antibody incompatibility</strong></td>
<td>have evidence of</td>
<td>Antibodies might not be present in the up-to-date blood specimen, but once the Patient has produced an antibody at any point in their history, they will always be able to produce antibodies if they are exposed to that antigen in the future.</td>
</tr>
<tr>
<td><strong>Antibody screen</strong></td>
<td>be negative</td>
<td>Have a negative result: there are no clinically-significant antibodies present.</td>
</tr>
<tr>
<td></td>
<td>be positive</td>
<td>Have a positive result: there are clinically-significant antibodies present.</td>
</tr>
<tr>
<td></td>
<td>perform on an up-to-date blood specimen</td>
<td></td>
</tr>
<tr>
<td><strong>Blood specimen</strong></td>
<td>be (in)compatible with a unit of blood product</td>
<td></td>
</tr>
<tr>
<td></td>
<td>be up-to-date</td>
<td>Be less than 72 hours old.</td>
</tr>
<tr>
<td></td>
<td>have negative antibody screen</td>
<td>Have a negative result: there are no clinically-significant antibodies present.</td>
</tr>
<tr>
<td></td>
<td>have positive antibody screen</td>
<td>Have a positive result: there are clinically-significant antibodies present.</td>
</tr>
<tr>
<td></td>
<td>obtain from Patient</td>
<td></td>
</tr>
<tr>
<td><strong>Compatibility testing</strong></td>
<td>This testing is done for both ABO / Rh and antibody compatibility.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>be full</td>
<td></td>
</tr>
<tr>
<td></td>
<td>be immediate-spin</td>
<td></td>
</tr>
<tr>
<td></td>
<td>perform</td>
<td></td>
</tr>
<tr>
<td></td>
<td>produce positive result</td>
<td>The tested unit of blood product is compatible with the Patient's up-to-date blood specimen.</td>
</tr>
<tr>
<td><strong>DBSS</strong></td>
<td>generate ABO / Rh identification and antibody screen paperwork</td>
<td>Added the word “paperwork”, since there was no clear definition of what was being generated and all conversations with DEs indicated that DBSS was not expected to perform the ABO / Rh identification or antibody screen itself, only the human blood bank technicians were expected to do that.</td>
</tr>
<tr>
<td></td>
<td>provide a list of potentially-compatible units of blood product</td>
<td>A unit of blood product cannot be assigned to a Patient unless it is on the list of potentially-compatible candidate units of blood product that DBSS provides.</td>
</tr>
<tr>
<td>Term</td>
<td>Definition</td>
<td>Notes</td>
</tr>
<tr>
<td>-------------------------------------------</td>
<td>---------------------------------------------------------------------------</td>
<td>----------------------------------------------------------------------</td>
</tr>
<tr>
<td>DBSS</td>
<td>provide a list of potentially-compatible units of blood product</td>
<td></td>
</tr>
<tr>
<td>emergency supervisor override</td>
<td></td>
<td></td>
</tr>
<tr>
<td>order for a type and screen</td>
<td>Order for an ABO / Rh identification and antibody screen on an up-to-date Patient blood specimen.</td>
<td></td>
</tr>
<tr>
<td>order for a type, screen, and crossmatch</td>
<td>Order for a full ABO / Rh and antibody compatibility testing until compatible unit(s) of blood product are identified and can be released from the blood bank repository.</td>
<td></td>
</tr>
<tr>
<td>Patient</td>
<td></td>
<td></td>
</tr>
<tr>
<td>have a unit of blood product assigned to</td>
<td></td>
<td></td>
</tr>
<tr>
<td>have history of ABO / Rh incompatibility</td>
<td></td>
<td></td>
</tr>
<tr>
<td>have history of clinically-significant red cell antibodies</td>
<td></td>
<td></td>
</tr>
<tr>
<td>have unknown history</td>
<td></td>
<td></td>
</tr>
<tr>
<td>obtain blood specimen from</td>
<td></td>
<td></td>
</tr>
<tr>
<td>red cell antibodies</td>
<td></td>
<td></td>
</tr>
<tr>
<td>be clinically-significant</td>
<td></td>
<td></td>
</tr>
<tr>
<td>have history of clinically-significant</td>
<td></td>
<td></td>
</tr>
<tr>
<td>unit of blood product</td>
<td></td>
<td></td>
</tr>
<tr>
<td>assign to a Patient</td>
<td></td>
<td></td>
</tr>
<tr>
<td>be (in)compatible with an up-to-date blood specimen</td>
<td>ABO / Rh - and antibody - compatible.</td>
<td></td>
</tr>
<tr>
<td>be identified as compatible with an up-to-date blood specimen</td>
<td></td>
<td></td>
</tr>
<tr>
<td>be potentially compatible with an up-to-date blood specimen</td>
<td></td>
<td></td>
</tr>
<tr>
<td>choose</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ensure availability of compatible with an up-to-date blood specimen</td>
<td></td>
<td></td>
</tr>
<tr>
<td>lack corresponding antigens</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
C.5.4 Formal Property Specifications

Event alphabet:
- A: perform an ABO/Rh identification on Patient blood specimen
- B: assign a unit of blood product to a Patient
- START: receive order for a type and screen OR for a type, screen, and crossmatch
- END: release a unit of blood product from blood bank repository
- C: perform an antibody screen on Patient blood specimen

SCOPE:
1. A restricted interval in the event sequence can have both a starting delimiter, receive order for a type and screen OR for a type, screen, and crossmatch, and an ending delimiter, release a unit of blood product from blood bank repository.
2. The behavior is required to hold from an occurrence of receive order for a type and screen OR for a type, screen, and crossmatch, if it ever occurs, through to the first subsequent occurrence of release a unit of blood product from blood bank repository, if it ever occurs.
3. If there are multiple occurrences of receive order for a type and screen OR for a type, screen, and crossmatch within this restricted interval do not have an effect.
4. receive order for a type and screen OR for a type, screen, and crossmatch is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if receive order for a type and screen OR for a type, screen, and crossmatch does occur, release a unit of blood product from blood bank repository is not required to occur subsequently and if it does not occur subsequently, then the behavior is not required to hold for the remainder of the event sequence.
5. There might be many restricted intervals in the event sequence. If receive order for a type and screen OR for a type, screen, and crossmatch occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of receive order for a type and screen OR for a type, screen, and crossmatch, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a potential new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

BEHAVIOR:
1. The events of primary interest in this behavior are perform an ABO/Rh identification on Patient blood specimen and assign a unit of blood product to a Patient.
2. The event of secondary interest in this behavior is perform an antibody screen on Patient blood specimen.
3. assign a unit of blood product to a Patient is not allowed to occur until after perform an ABO/Rh identification on Patient blood specimen occurs.
4. Before the first perform an ABO/Rh identification on Patient blood specimen occurs, perform an antibody screen on Patient blood specimen is allowed to occur zero or more times.
5. perform an ABO/Rh identification on Patient blood specimen is not required to occur.
6. Even if perform an ABO/Rh identification on Patient blood specimen does occur, assign a unit of blood product to a Patient is not required to occur after perform an ABO/Rh identification on Patient blood specimen occurs.
7. After perform an ABO/Rh identification on Patient blood specimen occurs, but before the first subsequent assign a unit of blood product to a Patient occurs:
   - perform an ABO/Rh identification on Patient blood specimen is allowed to occur again, zero or more times;
   - perform an antibody screen on Patient blood specimen is allowed to occur zero or more times.
8. After perform an ABO/Rh identification on Patient blood specimen and the first subsequent assign a unit of blood product to a Patient occurs:
   - perform an antibody screen on Patient blood specimen is allowed to occur zero or more times;
   - Both perform an ABO/Rh identification on Patient blood specimen and assign a unit of blood product to a Patient are allowed to occur again, zero or more times, and there are no restrictions on the occurrences of any future events.

FSA and DNL for Property DBSS.A.1a
Event alphabet:
- A: perform an antibody screen on Patient blood specimen
- B: assign a unit of blood product to a Patient
- START: receive order for a type and screen OR for a type, screen, and crossmatch
- END: release a unit of blood product from blood bank repository
- C: perform an ABO/Rh identification on Patient blood specimen

**SCOPE:**
1. A restricted interval in the event sequence can have both a starting delimiter, receive order for a type and screen OR for a type, screen, and crossmatch, and an ending delimiter, release a unit of blood product from blood bank repository.
2. The behavior is required to hold from an occurrence of receive order for a type and screen OR for a type, screen, and crossmatch, if it ever occurs, through to the first subsequent occurrence of release a unit of blood product from blood bank repository, if it ever occurs.
3. If there are multiple occurrences of receive order for a type and screen OR for a type, screen, and crossmatch without an occurrence of release a unit of blood product from blood bank repository in between them, only the first of those occurrences of receive order for a type and screen OR for a type, screen, and crossmatch potentially starts a restricted interval; later occurrences of receive order for a type and screen OR for a type, screen, and crossmatch within this restricted interval do not have an effect.
4. receive order for a type and screen OR for a type, screen, and crossmatch is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if receive order for a type and screen OR for a type, screen, and crossmatch does occur, release a unit of blood product from blood bank repository is not required to occur subsequently and if it does not occur subsequently, then the behavior is not required to hold for the remainder of the event sequence.
5. There might be many restricted intervals in the event sequence. If receive order for a type and screen OR for a type, screen, and crossmatch occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of receive order for a type and screen OR for a type, screen, and crossmatch, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a potential new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

**BEHAVIOR:**
1. The events of primary interest in this behavior are perform an antibody screen on Patient blood specimen and assign a unit of blood product to a Patient.
2. The event of secondary interest in this behavior is perform an ABO/Rh identification on Patient blood specimen.
3. assign a unit of blood product to a Patient is not allowed to occur until after perform an antibody screen on Patient blood specimen occurs.
4. Before the first perform an antibody screen on Patient blood specimen occurs, perform an ABO/Rh identification on Patient blood specimen is allowed to occur zero or more times.
5. perform an antibody screen on Patient blood specimen is required to occur, whether or not assign a unit of blood product to a Patient eventually occurs.
6. assign a unit of blood product to a Patient is not required to occur after perform an antibody screen on Patient blood specimen occurs.
7. After perform an antibody screen on Patient blood specimen occurs, but before the first subsequent assign a unit of blood product to a Patient occurs:
   - perform an antibody screen on Patient blood specimen is allowed to occur again, zero or more times;
   - perform an ABO/Rh identification on Patient blood specimen is allowed to occur zero or more times.
8. After perform an antibody screen on Patient blood specimen and the first subsequent assign a unit of blood product to a Patient occurs:
   - both perform an antibody screen on Patient blood specimen and assign a unit of blood product to a Patient are allowed to occur again, zero or more times, and there are no restrictions on the occurrences of any future events.

FSA and DNL for Property DBSS.A.1b

528
Event alphabet:
- A: identify unit of blood product i as compatible
- B: assign unit of blood product i to Patient

**SCOPE:**
1. From the start of any event sequence through to the end of that event sequence, the behavior must hold.

**BEHAVIOR:**
1. The events of primary interest in this behavior are identify unit of blood product i as compatible and assign unit of blood product i to Patient.
2. There are no events of secondary interest in this behavior.
3. assign unit of blood product i to Patient is not allowed to occur until after identify unit of blood product i as compatible occurs.
4. identify unit of blood product i as compatible is not required to occur.
5. Even if identify unit of blood product i as compatible does occur, assign unit of blood product i to Patient is not required to occur after identify unit of blood product i as compatible occurs.
6. After identify unit of blood product i as compatible occurs, but before the first subsequent assign unit of blood product i to Patient occurs, identify unit of blood product i as compatible is allowed to occur again, zero or more times.
7. After identify unit of blood product i as compatible and the first subsequent assign unit of blood product i to Patient occur:
   - assign unit of blood product i to Patient is not allowed to occur again until after another identify unit of blood product i as compatible occurs; identify unit of blood product i as compatible is allowed to occur again and, if it does, then the situation is the same as when the first identify unit of blood product i as compatible occurred, meaning that the restrictions described in parts 5, 6, and 7 would again apply.

FSA and DNL for Property DBSS.A.2a
Event alphabet:
- **A**: obtain up-to-date blood specimen
- **START**: find that blood specimen is more than 72 hours old
- **END**: identify a unit of blood product as compatible

**SCOPE:**
1. A restricted interval in the event sequence can have both a starting delimiter, **find that blood specimen is more than 72 hours old**, and an ending delimiter, **identify a unit of blood product as compatible**.
2. The behavior is required to hold from an occurrence of **find that blood specimen is more than 72 hours old**, if it ever occurs, through to the first subsequent occurrence of **identify a unit of blood product as compatible**, if it ever occurs.
3. If there are multiple occurrences of **find that blood specimen is more than 72 hours old** without an occurrence of **identify a unit of blood product as compatible** in between them, only the first of those occurrences of **find that blood specimen is more than 72 hours old** potentially starts a restricted interval; later occurrences of **find that blood specimen is more than 72 hours old** within this restricted interval do not have an effect.
4. **find that blood specimen is more than 72 hours old** is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if **find that blood specimen is more than 72 hours old** does occur, **identify a unit of blood product as compatible** is not required to occur subsequently and if it does not occur subsequently, then the behavior is not required to hold for the remainder of the event sequence.
5. There might be many restricted intervals in the event sequence. If **find that blood specimen is more than 72 hours old** occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of **find that blood specimen is more than 72 hours old**, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a potential new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

**BEHAVIOR:**
1. The event of primary interest in this behavior is **obtain up-to-date blood specimen**.
2. There are no events of secondary interest in this behavior.
3. **obtain up-to-date blood specimen** is required to occur at least once.

**FSA and DNL for Property DBSS.A.2b**
Event alphabet:
- A: receive order for a type and screen OR for a type, screen, and crossmatch
- B: DBSS generates ABO/Rh identification paperwork
- C: DBSS generates antibody screen paperwork

**SCOPE:**
1. From the start of any event sequence through to the end of that event sequence, the behavior must hold.

**BEHAVIOR:**
1. The events of primary interest in this behavior are receive order for a type and screen OR for a type, screen, and crossmatch and DBSS generates ABO/Rh identification paperwork.
2. The event of secondary interest in this behavior is DBSS generates antibody screen paperwork.
3. If receive order for a type and screen OR for a type, screen, and crossmatch occurs, DBSS generates ABO/Rh identification paperwork is required to occur subsequently.
4. Before the first receive order for a type and screen OR for a type, screen, and crossmatch occurs:
   - DBSS generates ABO/Rh identification paperwork is allowed to occur zero or more times;
   - DBSS generates antibody screen paperwork is allowed to occur zero or more times.
5. If receive order for a type and screen OR for a type, screen, and crossmatch occurs, but before the first subsequent DBSS generates ABO/Rh identification paperwork occurs:
   - receive order for a type and screen OR for a type, screen, and crossmatch is not allowed to occur again;
   - DBSS generates antibody screen paperwork is allowed to occur zero or more times.
6. After receive order for a type and screen OR for a type, screen, and crossmatch occurs, but before the first subsequent DBSS generates ABO/Rh identification paperwork occurs:
   - receive order for a type and screen OR for a type, screen, and crossmatch is allowed to occur zero or more times;
   - DBSS generates antibody screen paperwork is allowed to occur zero or more times, before another receive order for a type and screen OR for a type, screen, and crossmatch occurs; receive order for a type and screen OR for a type, screen, and crossmatch is allowed to occur again and, if it does, then the situation is the same as when the first receive order for a type and screen OR for a type, screen, and crossmatch occurred, meaning that the restrictions described in parts 3, 6, and 7 would again apply.

FSA and DNL for Property DBSS.A.3a
Event alphabet:
- A: receive order for a type and screen OR for a type, screen, and crossmatch
- B: DBSS generates antibody screen paperwork
- C: DBSS generates ABO/Rh identification paperwork

SCOPE:
1. From the start of any event sequence through to the end of that event sequence, the behavior must hold.

BEHAVIOR:
1. The events of primary interest in this behavior are receive order for a type and screen OR for a type, screen, and crossmatch and DBSS generates antibody screen paperwork.
2. The event of secondary interest in this behavior is DBSS generates ABO/Rh identification paperwork.
3. If receive order for a type and screen OR for a type, screen, and crossmatch occurs, DBSS generates antibody screen paperwork is required to occur subsequently.
4. If receive order for a type and screen OR for a type, screen, and crossmatch occurs:
   - DBSS generates antibody screen paperwork is allowed to occur zero or more times;
   - DBSS generates ABO/Rh identification paperwork is allowed to occur zero or more times.
5. Before the first receive order for a type and screen OR for a type, screen, and crossmatch occurs:
   - DBSS generates antibody screen paperwork is allowed to occur zero or more times;
   - DBSS generates ABO/Rh identification paperwork is allowed to occur zero or more times.
6. After receive order for a type and screen OR for a type, screen, and crossmatch occurs, but before the first subsequent DBSS generates antibody screen paperwork occurs:
   - receive order for a type and screen OR for a type, screen, and crossmatch is not allowed to occur again;
   - DBSS generates ABO/Rh identification paperwork is allowed to occur zero or more times.
7. After receive order for a type and screen OR for a type, screen, and crossmatch and the first subsequent DBSS generates antibody screen paperwork occur:
   - DBSS generates ABO/Rh identification paperwork is allowed to occur zero or more times;
   - DBSS generates antibody screen paperwork is allowed to occur again, zero or more times, before another receive order for a type and screen OR for a type, screen, and crossmatch occurs; receive order for a type and screen OR for a type, screen, and crossmatch is allowed to occur again and, if it does, then the situation is the same as when the first receive order for a type and screen OR for a type, screen, and crossmatch occurred, meaning that the restrictions described in parts 3, 6, and 7 would again apply.

FSA and DNL for Property DBSS.A.3b
perform full ABO/Rh and antibody compatibility testing for unit of blood product i

BEHAVIOR:

1. The events of primary interest in this behavior are perform full ABO/Rh and antibody compatibility testing for unit of blood product i and identify unit of blood product i as compatible.

2. The events of secondary interest in this behavior are determine that immediate-spin ABO/Rh testing shows compatible result and determine that the Patient does not have a problematic history.

3. If there are multiple occurrences of receive order for a type and screen OR a type, screen, and crossmatch without an occurrence of determine that the Patient’s up-to-date blood specimen has a negative antibody screen in between them, only the first of those occurrences of receive order for a type and screen OR a type, screen, and crossmatch starts a restricted interval; later occurrences of receive order for a type and screen OR a type, screen, and crossmatch within this restricted interval do not have an effect.

4. receive order for a type and screen OR a type, screen, and crossmatch is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if receive order for a type and screen OR a type, screen, and crossmatch does occur, determine that the Patient’s up-to-date blood specimen has a negative antibody screen is not required to occur subsequently. Even if determine that the Patient’s up-to-date blood specimen has a negative antibody screen does not occur subsequently, the behavior is still required to hold, until the end of the event sequence.

5. There might be many restricted intervals in the event sequence. If receive order for a type and screen OR a type, screen, and crossmatch occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of receive order for a type and screen OR a type, screen, and crossmatch, meaning that the restrictions described in parts 6, 7, and 8 would again apply for a new restricted interval.

6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

7. After perform full ABO/Rh and antibody compatibility testing for unit of blood product i occurs, but before the first subsequent identify unit of blood product i as compatible occurs:
   - perform full ABO/Rh and antibody compatibility testing for unit of blood product i is allowed to occur again, zero or more times:
     - All the events of secondary interest are allowed to occur zero or more times.

8. After perform full ABO/Rh and antibody compatibility testing for unit of blood product i and the first subsequent identify unit of blood product i as compatible occur:
   - identify unit of blood product i as compatible is not allowed to occur again until after another perform full ABO/Rh and antibody compatibility testing for unit of blood product i occurs; perform full ABO/Rh and antibody compatibility testing for unit of blood product i is allowed to occur again and, if it does, then the situation is the same as when the first perform full ABO/Rh and antibody compatibility testing for unit of blood product i occurred, meaning that the restrictions described in parts 6, 7, and 8 would again apply.

SCOPE:

1. A restricted interval in the event sequence can have both a starting delimiter, receive order for a type and screen OR a type, screen, and crossmatch, and an ending delimiter, determine that the Patient’s up-to-date blood specimen has a negative antibody screen.

2. The behavior is required to hold from an occurrence of receive order for a type and screen OR a type, screen, and crossmatch, if it ever occurs, through to the first subsequent occurrence of determine that the Patient’s up-to-date blood specimen has a negative antibody screen, if it ever occurs.

3. If there are multiple occurrences of receive order for a type and screen OR a type, screen, and crossmatch without an occurrence of determine that the Patient’s up-to-date blood specimen has a negative antibody screen in between them, only the first of those occurrences of receive order for a type and screen OR a type, screen, and crossmatch starts a restricted interval; later occurrences of receive order for a type and screen OR a type, screen, and crossmatch within this restricted interval do not have an effect.

4. receive order for a type and screen OR a type, screen, and crossmatch is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if receive order for a type and screen OR a type, screen, and crossmatch does occur, determine that the Patient’s up-to-date blood specimen has a negative antibody screen is not required to occur subsequently. Even if determine that the Patient’s up-to-date blood specimen has a negative antibody screen does not occur subsequently, the behavior is still required to hold, until the end of the event sequence.

5. There might be many restricted intervals in the event sequence. If receive order for a type and screen OR a type, screen, and crossmatch occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of receive order for a type and screen OR a type, screen, and crossmatch, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a new restricted interval.

6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

BEHAVIOR:

1. The events of primary interest in this behavior are perform full ABO/Rh and antibody compatibility testing for unit of blood product i and identify unit of blood product i as compatible.

2. The events of secondary interest in this behavior are determine that immediate-spin ABO/Rh testing shows compatible result and determine that the Patient does not have a problematic history.

3. If there are multiple occurrences of receive order for a type and screen OR a type, screen, and crossmatch without an occurrence of determine that the Patient’s up-to-date blood specimen has a negative antibody screen in between them, only the first of those occurrences of receive order for a type and screen OR a type, screen, and crossmatch starts a restricted interval; later occurrences of receive order for a type and screen OR a type, screen, and crossmatch within this restricted interval do not have an effect.

4. receive order for a type and screen OR a type, screen, and crossmatch is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if receive order for a type and screen OR a type, screen, and crossmatch does occur, determine that the Patient’s up-to-date blood specimen has a negative antibody screen is not required to occur subsequently. Even if determine that the Patient’s up-to-date blood specimen has a negative antibody screen does not occur subsequently, the behavior is still required to hold, until the end of the event sequence.

5. There might be many restricted intervals in the event sequence. If receive order for a type and screen OR a type, screen, and crossmatch occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of receive order for a type and screen OR a type, screen, and crossmatch, meaning that the restrictions described in parts 6, 7, and 8 would again apply for a new restricted interval.

6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

7. After perform full ABO/Rh and antibody compatibility testing for unit of blood product i occurs, but before the first subsequent identify unit of blood product i as compatible occurs:
   - perform full ABO/Rh and antibody compatibility testing for unit of blood product i is allowed to occur again, zero or more times:
     - All the events of secondary interest are allowed to occur zero or more times.

8. After perform full ABO/Rh and antibody compatibility testing for unit of blood product i and the first subsequent identify unit of blood product i as compatible occur:
   - identify unit of blood product i as compatible is not allowed to occur again until after another perform full ABO/Rh and antibody compatibility testing for unit of blood product i occurs; perform full ABO/Rh and antibody compatibility testing for unit of blood product i is allowed to occur again and, if it does, then the situation is the same as when the first perform full ABO/Rh and antibody compatibility testing for unit of blood product i occurred, meaning that the restrictions described in parts 6, 7, and 8 would again apply.
BEHAVIOR:
1. The events of primary interest in this behavior are perform full ABO/Rh and antibody compatibility testing for unit of blood product i and identify unit of blood product i as compatible.
2. The events of secondary interest in this behavior are determine that immediate-spin ABO/Rh testing shows compatible result and determine that the Patient does not have a problematic history.
3. The behaviors are not required to hold until the end of a restricted interval; later occurrences of receive order for a type and screen OR a type, screen, and crossmatch within this restricted interval do not have an effect.
4. The behavior is not required to hold anywhere in the event sequence. Even if receive order for a type and screen OR a type, screen, and crossmatch occurs, the behavior is still required to hold, until the end of the event sequence.
5. The behavior is not required to occur subsequently. Even if determine that the Patient does not have a problematic history occurs, the behavior is still required to hold, until the end of the event sequence.
6. There are no restrictions imposed on the occurrences of any event outside of the restricted interval(s).

SCOPE:
1. A restricted interval in the event sequence can have both a starting delimiter, receive order for a type and screen OR a type, screen, and crossmatch, and an ending delimiter, determine that the Patient does not have a problematic history.
2. The behavior is required to hold from an occurrence of receive order for a type and screen OR a type, screen, and crossmatch, if it ever occurs, through to the first subsequent occurrence of determine that the Patient does not have a problematic history, if it ever occurs.
3. If there are multiple occurrences of receive order for a type and screen OR a type, screen, and crossmatch without an occurrence of determine that the Patient does not have a problematic history in between them, only the first of those occurrences of receive order for a type and screen OR a type, screen, and crossmatch starts a restricted interval; later occurrences of receive order for a type and screen OR a type, screen, and crossmatch within this restricted interval do not have an effect.
4. A restricted interval in the event sequence can have both a starting delimiter, receive order for a type and screen OR a type, screen, and crossmatch, and an ending delimiter, determine that the Patient does not have a problematic history.
5. There might be many restricted intervals in the event sequence. If receive order for a type and screen OR a type, screen, and crossmatch occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of receive order for a type and screen OR a type, screen, and crossmatch, meaning that the restrictions described in parts 6, 7, and 8 would again apply for a new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).
Event alphabet:
- A: perform full ABO/Rh and antibody compatibility testing for unit of blood product i
- B: identify unit of blood product i as compatible
- START: receive order for a type and screen OR a type, screen, and crossmatch
- END: determine that immediate-spin ABO/Rh testing shows compatible result
- C: determine that the Patient does not have a problematic history, determine that the Patient’s up-to-date blood specimen has a negative antibody screen

**SCOPE:**
1. A restricted interval in the event sequence can have both a starting delimiter receive order for a type and screen OR a type, screen, and crossmatch, and an ending delimiter, determine that immediate-spin ABO/Rh testing shows compatible result.
2. The behavior is required to hold from an occurrence of receive order for a type and screen OR a type, screen, and crossmatch, if it ever occurs, through to the first subsequent occurrence of determine that immediate-spin ABO/Rh testing shows compatible result, if it ever occurs.
3. If there are multiple occurrences of receive order for a type and screen OR a type, screen, and crossmatch without an occurrence of determine that immediate-spin ABO/Rh testing shows compatible result in between them, only the first of those occurrences of receive order for a type and screen OR a type, screen, and crossmatch within this restricted interval do not have an effect.
4. receive order for a type and screen OR a type, screen, and crossmatch is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if receive order for a type and screen OR a type, screen, and crossmatch does occur, determine that immediate-spin ABO/Rh testing shows compatible result is not required to occur subsequently. Even if determine that immediate-spin ABO/Rh testing shows compatible result does not occur subsequently, the behavior is still required to hold, until the end of the event sequence.
5. There might be many restricted intervals in the event sequence. If receive order for a type and screen OR a type, screen, and crossmatch occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of receive order for a type and screen OR a type, screen, and crossmatch, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

**BEHAVIOR:**
1. The events of primary interest in this behavior are perform full ABO/Rh and antibody compatibility testing for unit of blood product i and identify unit of blood product i as compatible.
2. The events of secondary interest in this behavior are determine that the Patient does not have a problematic history and determine that the Patient’s up-to-date blood specimen has a negative antibody screen.
3. identify unit of blood product i as compatible is not allowed to occur until after perform full ABO/Rh and antibody compatibility testing for unit of blood product i occurs.
4. Before the first perform full ABO/Rh and antibody compatibility testing for unit of blood product i occurs, all the events of secondary interest are allowed to occur zero or more times.
5. perform full ABO/Rh and antibody compatibility testing for unit of blood product i is not required to occur.
6. Even if perform full ABO/Rh and antibody compatibility testing for unit of blood product i does occur, identify unit of blood product i as compatible is not required to occur after perform full ABO/Rh and antibody compatibility testing for unit of blood product i occurs.
7. After perform full ABO/Rh and antibody compatibility testing for unit of blood product i occurs, but before the first subsequent identify unit of blood product i as compatible occurs:
   - perform full ABO/Rh and antibody compatibility testing for unit of blood product i is allowed to occur again, zero or more times;
   - All the events of secondary interest are allowed to occur zero or more times.
8. After perform full ABO/Rh and antibody compatibility testing for unit of blood product i and the first subsequent identify unit of blood product i as compatible occur:
   - All the events of secondary interest are allowed to occur zero or more times;
   - identify unit of blood product i as compatible is not allowed to occur again until after another perform full ABO/Rh and antibody compatibility testing for unit of blood product i occurs; perform full ABO/Rh and antibody compatibility testing for unit of blood product i is allowed to occur again and, if it does, then the situation is the same as when the first perform full ABO/Rh and antibody compatibility testing for unit of blood product i occurred, meaning that the restrictions described in parts 6, 7, and 8 would again apply.
Event alphabet:
- A: find that blood specimen is more than 72 hours old
- B: obtain up-to-date blood specimen
- START: determine that a Patient’s situation is problematic
- END: perform an ABO/Rh identification
- C: perform an antibody screen

SCOPE:
1. A restricted interval in the event sequence can have both a starting delimiter, determine that a Patient’s situation is problematic, and an ending delimiter, perform an ABO/Rh identification.
2. The behavior is required to hold from an occurrence of determine that a Patient’s situation is problematic, if it ever occurs, through to the first subsequent occurrence of perform an ABO/Rh identification, if it ever occurs.
3. If there are multiple occurrences of determine that a Patient’s situation is problematic without an occurrence of perform an ABO/Rh identification in between them, only the first of those occurrences of determine that a Patient’s situation is problematic potentially starts a restricted interval; later occurrences of determine that a Patient’s situation is problematic within this restricted interval do not have an effect.
4. determine that a Patient’s situation is problematic is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if determine that a Patient’s situation is problematic does occur, perform an ABO/Rh identification is not required to occur subsequently and if it does not occur subsequently, then the behavior is not required to hold for the remainder of the event sequence.
5. There might be many restricted intervals in the event sequence. If determine that a Patient’s situation is problematic occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of determine that a Patient’s situation is problematic, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a potential new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

BEHAVIOR:
1. The events of primary interest in this behavior are find that blood specimen is more than 72 hours old and obtain up-to-date blood specimen.
2. The event of secondary interest in this behavior is perform an antibody screen.
3. If find that blood specimen is more than 72 hours old occurs, obtain up-to-date blood specimen is required to occur subsequently.
4. Before the first find that blood specimen is more than 72 hours old occurs:
   - obtain up-to-date blood specimen is allowed to occur zero or more times;
   - perform an antibody screen is allowed to occur zero or more times.
5. find that blood specimen is more than 72 hours old is not required to occur.
6. After find that blood specimen is more than 72 hours old occurs, but before the first subsequent obtain up-to-date blood specimen occurs:
   - find that blood specimen is more than 72 hours old is not allowed to occur again;
   - perform an antibody screen is allowed to occur zero or more times.
7. After find that blood specimen is more than 72 hours old and the first subsequent obtain up-to-date blood specimen occur:
   - perform an antibody screen is allowed to occur zero or more times;
   - obtain up-to-date blood specimen is allowed to occur again, zero or more times, before another find that blood specimen is more than 72 hours old occurs; find that blood specimen is more than 72 hours old is allowed to occur again and, if it does, then the situation is the same as when the first find that blood specimen is more than 72 hours old occurred, meaning that the restrictions described in parts 3, 6, and 7 would again apply.

FSA and DNL for Property DBSS.B.2a
Event alphabet:
- A: find that blood specimen is more than 72 hours old
- B: obtain up-to-date blood specimen
- START: determine that a Patient’s situation is problematic
- END: perform an antibody screen
- C: perform an ABO/Rh identification

**SCOPE:**
1. A restricted interval in the event sequence can have both a starting delimiter, determine that a Patient’s situation is problematic, and an ending delimiter, perform an antibody screen.
2. The behavior is required to hold from an occurrence of determine that a Patient’s situation is problematic, if it ever occurs, through to the first subsequent occurrence of perform an antibody screen, if it ever occurs.
3. If there are multiple occurrences of determine that a Patient’s situation is problematic without an occurrence of perform an antibody screen in between them, only the first of those occurrences of determine that a Patient’s situation is problematic potentially starts a restricted interval; later occurrences of determine that a Patient’s situation is problematic within this restricted interval do not have an effect.
4. determine that a Patient’s situation is problematic is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if determine that a Patient’s situation is problematic does occur, perform an antibody screen is not required to occur subsequently and if it does not occur subsequently, then the behavior is not required to hold for the remainder of the event sequence.
5. There might be many restricted intervals in the event sequence. If determine that a Patient’s situation is problematic occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of determine that a Patient’s situation is problematic, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a potential new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

**BEHAVIOR:**
1. The events of primary interest in this behavior are find that blood specimen is more than 72 hours old and obtain up-to-date blood specimen.
2. The event of secondary interest in this behavior is perform an ABO/Rh identification.
3. If find that blood specimen is more than 72 hours old occurs, obtain up-to-date blood specimen is required to occur subsequently.
4. Before the first find that blood specimen is more than 72 hours old occurs:
   - obtain up-to-date blood specimen is allowed to occur zero or more times;
   - perform an ABO/Rh identification is allowed to occur zero or more times.
5. find that blood specimen is more than 72 hours old is not required to occur.
6. After find that blood specimen is more than 72 hours old occurs, but before the first subsequent obtain up-to-date blood specimen occurs:
   - find that blood specimen is more than 72 hours old is not allowed to occur again;
   - perform an ABO/Rh identification is allowed to occur zero or more times.
7. After find that blood specimen is more than 72 hours old and the first subsequent obtain up-to-date blood specimen occur:
   - perform an ABO/Rh identification is allowed to occur zero or more times;
   - obtain up-to-date blood specimen is allowed to occur again, zero or more times, before another find that blood specimen is more than 72 hours old occurs; find that blood specimen is more than 72 hours old is allowed to occur again and, if it does, then the situation is the same as when the first find that blood specimen is more than 72 hours old occurred, meaning that the restrictions described in parts 3, 6, and 7 would again apply.

FSA and DNL for Property DBSS.B.2b
**Event alphabet:**
- **A**: determine that Patient’s situation is problematic
- **B**: perform full ABO/Rh compatibility testing for unit of blood product i
- **C**: perform full antibody compatibility testing for unit of blood product i
- **START**: receive order for a type and screen OR a type, screen, and crossmatch
- **END**: identify unit of blood product i as compatible

**SCOPE:**
1. A restricted interval in the event sequence can have both a starting delimiter, receive order for a type and screen OR a type, screen, and crossmatch, and an ending delimiter, identify unit of blood product i as compatible.
2. The behavior is required to hold from an occurrence of receive order for a type and screen OR a type, screen, and crossmatch, if it ever occurs, through to the first subsequent occurrence of identify unit of blood product i as compatible, if it ever occurs.
3. If there are multiple occurrences of receive order for a type and screen OR a type, screen, and crossmatch within this restricted interval do not have an effect.
4. receive order for a type and screen OR a type, screen, and crossmatch is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if receive order for a type and screen OR a type, screen, and crossmatch does occur, identify unit of blood product i as compatible is not required to occur subsequently. Even if identify unit of blood product i as compatible does not occur subsequently, the behavior is still required to hold, until the end of the event sequence.
5. There might be many restricted intervals in the event sequence. If receive order for a type and screen OR a type, screen, and crossmatch occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of receive order for a type and screen OR a type, screen, and crossmatch, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

**BEHAVIOR:**
1. The events of primary interest in this behavior are determine that Patient’s situation is problematic and perform full ABO/Rh compatibility testing for unit of blood product i.
2. The event of secondary interest in this behavior is perform full antibody compatibility testing for unit of blood product i.
3. If determine that Patient’s situation is problematic occurs, perform full ABO/Rh compatibility testing for unit of blood product i is required to occur subsequently.
4. Before the first determine that Patient’s situation is problematic occurs:
   - perform full ABO/Rh compatibility testing for unit of blood product i is allowed to occur zero or more times;
   - perform full antibody compatibility testing for unit of blood product i is allowed to occur zero or more times.
5. determine that Patient’s situation is problematic is not required to occur.
6. After determine that Patient’s situation is problematic occurs, but before the first subsequent perform full ABO/Rh compatibility testing for unit of blood product i occurs:
   - perform full ABO/Rh compatibility testing for unit of blood product i is allowed to occur again, zero or more times;
   - perform full antibody compatibility testing for unit of blood product i is allowed to occur zero or more times.
7. After determine that Patient’s situation is problematic and the first subsequent perform full ABO/Rh compatibility testing for unit of blood product i occur:
   - both determine that Patient’s situation is problematic and perform full ABO/Rh compatibility testing for unit of blood product i are allowed to occur again, zero or more times, and there are no restrictions on the occurrences of any future events.

**FSA and DNL for Property DBSS.B.3a**
**Event alphabet:**
- **A:** determine that Patient’s situation is problematic
- **B:** perform full antibody compatibility testing for unit of blood product i
- **C:** perform full ABO/Rh compatibility testing for unit of blood product i
- **START:** receive order for a type and screen OR a type, screen, and crossmatch
- **END:** identify unit of blood product i as compatible

**SCOPE:**
1. A restricted interval in the event sequence can have both a starting delimiter, receive order for a type and screen OR a type, screen, and crossmatch, and an ending delimiter, identify unit of blood product i as compatible.
2. The behavior is required to hold from an occurrence of receive order for a type and screen OR a type, screen, and crossmatch, if it ever occurs, through to the first subsequent occurrence of identify unit of blood product i as compatible, if it ever occurs.
3. If there are multiple occurrences of receive order for a type and screen OR a type, screen, and crossmatch without an occurrence of identify unit of blood product i as compatible in between them, only the first of those occurrences of receive order for a type and screen OR a type, screen, and crossmatch starts a restricted interval; later occurrences of receive order for a type and screen OR a type, screen, and crossmatch within this restricted interval do not have an effect.
4. Receive order for a type and screen OR a type, screen, and crossmatch is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if receive order for a type and screen OR a type, screen, and crossmatch does occur, identify unit of blood product i as compatible is not required to occur subsequently. Even if identify unit of blood product i as compatible does not occur subsequently, the behavior is still required to hold, until the end of the event sequence.
5. There might be many restricted intervals in the event sequence. If receive order for a type and screen OR a type, screen, and crossmatch occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of receive order for a type and screen OR a type, screen, and crossmatch, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

**BEHAVIOR:**
1. The events of primary interest in this behavior are determine that Patient’s situation is problematic and perform full antibody compatibility testing for unit of blood product i.
2. The event of secondary interest in this behavior is perform full ABO/Rh compatibility testing for unit of blood product i.
3. If determine that Patient’s situation is problematic occurs, perform full antibody compatibility testing for unit of blood product i is required to occur subsequently.
4. Before the first determine that Patient’s situation is problematic occurs:
   - perform full antibody compatibility testing for unit of blood product i is allowed to occur zero or more times;
   - perform full ABO/Rh compatibility testing for unit of blood product i is allowed to occur zero or more times.
5. Determine that Patient’s situation is problematic is not required to occur.
6. After determine that Patient’s situation is problematic occurs, but before the first subsequent perform full antibody compatibility testing for unit of blood product i occurs:
   - perform full antibody compatibility testing for unit of blood product i is allowed to occur again, zero or more times;
   - perform full ABO/Rh compatibility testing for unit of blood product i is allowed to occur zero or more times.
7. After determine that Patient’s situation is problematic and the first subsequent perform full antibody compatibility testing for unit of blood product i occur:
   - Both determine that Patient’s situation is problematic and perform full antibody compatibility testing for unit of blood product i are allowed to occur again, zero or more times, and there are no restrictions on the occurrences of any future events.

FSA and DNL for Property DBSS.B.3b
**Event alphabet:**
- A: find evidence of an ABO/Rh OR antibody incompatibility with unit of blood product i
- B: determine that AHG and IgG phase crossmatch shows compatible result with unit of blood product i
- START: choose potentially-compatible unit of blood product i
- END: identify unit of blood product i as compatible

**SCOPE:**
1. A restricted interval in the event sequence can have both a starting delimiter, choose potentially-compatible unit of blood product i, and an ending delimiter, identify unit of blood product i as compatible.
2. The behavior is required to hold from an occurrence of choose potentially-compatible unit of blood product i, if it ever occurs, through to the first subsequent occurrence of identify unit of blood product i as compatible, if it ever occurs.
3. If there are multiple occurrences of choose potentially-compatible unit of blood product i without an occurrence of identify unit of blood product i as compatible in between them, only the last of those occurrences of choose potentially-compatible unit of blood product i potentially starts a restricted interval; each of those occurrences of choose potentially-compatible unit of blood product i resets the beginning of this restricted interval.
4. choose potentially-compatible unit of blood product i is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if choose potentially-compatible unit of blood product i does occur, identify unit of blood product i as compatible is not required to occur subsequently and if it does not occur subsequently, then the behavior is not required to hold for the remainder of the event sequence.
5. There might be many restricted intervals in the event sequence. If choose potentially-compatible unit of blood product i occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of choose potentially-compatible unit of blood product i, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a potential new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

**BEHAVIOR:**
1. The events of primary interest in this behavior are find evidence of an ABO/Rh OR antibody incompatibility with unit of blood product i and determine that AHG and IgG phase crossmatch shows compatible result with unit of blood product i.
2. There are no events of secondary interest in this behavior.
3. If find evidence of an ABO/Rh OR antibody incompatibility with unit of blood product i occurs, determine that AHG and IgG phase crossmatch shows compatible result with unit of blood product i is required to occur subsequently.
4. Before the first find evidence of an ABO/Rh OR antibody incompatibility with unit of blood product i occurs, determine that AHG and IgG phase crossmatch shows compatible result with unit of blood product i is allowed to occur zero or more times.
5. find evidence of an ABO/Rh OR antibody incompatibility with unit of blood product i is not required to occur.
6. After find evidence of an ABO/Rh OR antibody incompatibility with unit of blood product i occurs, but before the first subsequent determine that AHG and IgG phase crossmatch shows compatible result with unit of blood product i occurs, find evidence of an ABO/Rh OR antibody incompatibility with unit of blood product i is allowed to occur again, zero or more times.
7. After find evidence of an ABO/Rh OR antibody incompatibility with unit of blood product i and the first subsequent determine that AHG and IgG phase crossmatch shows compatible result with unit of blood product i occur:
   - Both find evidence of an ABO/Rh OR antibody incompatibility with unit of blood product i and determine that AHG and IgG phase crossmatch shows compatible result with unit of blood product i are allowed to occur again, zero or more times, and there are no restrictions on the occurrences of any future events.

FSA and DNL for Property DBSS.B.4
Event alphabet:
- A: perform ABO/Rh identification on Patient blood specimen
- B: DBSS provides a list of potentially-compatible units of blood product
- C: perform antibody screen on Patient blood specimen

**SCOPE:**
1. From the start of any event sequence through to the end of that event sequence, the behavior must hold.

**BEHAVIOR:**
1. The events of primary interest in this behavior are perform ABO/Rh identification on Patient blood specimen and DBSS provides a list of potentially-compatible units of blood product.
2. The event of secondary interest in this behavior is perform antibody screen on Patient blood specimen.
3. DBSS provides a list of potentially-compatible units of blood product is not allowed to occur until after perform ABO/Rh identification on Patient blood specimen occurs.
4. Before the first perform ABO/Rh identification on Patient blood specimen occurs, perform antibody screen on Patient blood specimen is allowed to occur zero or more times.
5. perform ABO/Rh identification on Patient blood specimen is not required to occur.
6. Even if perform ABO/Rh identification on Patient blood specimen does occur, DBSS provides a list of potentially-compatible units of blood product is not required to occur after perform ABO/Rh identification on Patient blood specimen occurs.
7. After perform ABO/Rh identification on Patient blood specimen occurs, but before the first subsequent DBSS provides a list of potentially-compatible units of blood product occurs:
   - perform ABO/Rh identification on Patient blood specimen is allowed to occur again, zero or more times;
   - perform antibody screen on Patient blood specimen is allowed to occur zero or more times.
8. After perform ABO/Rh identification on Patient blood specimen and the first subsequent DBSS provides a list of potentially-compatible units of blood product occur:
   - perform antibody screen on Patient blood specimen is allowed to occur zero or more times;
   - DBSS provides a list of potentially-compatible units of blood product is not allowed to occur again until after another perform ABO/Rh identification on Patient blood specimen occurs; perform ABO/Rh identification on Patient blood specimen is allowed to occur again and, if it does, then the situation is the same as when the first perform ABO/Rh identification on Patient blood specimen occurred, meaning that the restrictions described in parts 6, 7, and 8 would again apply.

FSA and DNL for Property DBSS.C.1a
Event alphabet:
- A: perform antibody screen on Patient blood specimen
- B: DBSS provides a list of potentially-compatible units of blood product
- C: perform ABO/Rh identification on Patient blood specimen

SCOPE:
1. From the start of any event sequence through to the end of that event sequence, the behavior must hold.

BEHAVIOR:
1. The events of primary interest in this behavior are perform antibody screen on Patient blood specimen and DBSS provides a list of potentially-compatible units of blood product.
2. The event of secondary interest in this behavior is perform ABO/Rh identification on Patient blood specimen.
3. DBSS provides a list of potentially-compatible units of blood product is not allowed to occur until after perform antibody screen on Patient blood specimen occurs.
4. Before the first perform antibody screen on Patient blood specimen occurs, perform ABO/Rh identification on Patient blood specimen is allowed to occur zero or more times.
5. perform antibody screen on Patient blood specimen is not required to occur.
6. Even if perform antibody screen on Patient blood specimen does occur, DBSS provides a list of potentially-compatible units of blood product is not required to occur after perform antibody screen on Patient blood specimen occurs.
7. After perform antibody screen on Patient blood specimen occurs, but before the first subsequent DBSS provides a list of potentially-compatible units of blood product occurs:
   - perform antibody screen on Patient blood specimen is allowed to occur again, zero or more times;
   - perform ABO/Rh identification on Patient blood specimen is allowed to occur zero or more times.
8. After perform antibody screen on Patient blood specimen and the first subsequent DBSS provides a list of potentially-compatible units of blood product occurs:
   - perform ABO/Rh identification on Patient blood specimen is allowed to occur again and, if it does, then the situation is the same as when the first perform antibody screen on Patient blood specimen occurred, meaning that the restrictions described in parts 6, 7, and 8 would again apply.

FSA and DNL for Property DBSS.C.1b
Event alphabet:
- A: obtain up-to-date blood specimen
- START: find that blood specimen is more than 72 hours old
- END: perform ABO/Rh identification OR antibody screen on Patient blood specimen

SCOPE:
1. A restricted interval in the event sequence can have both a starting delimiter, **find that blood specimen is more than 72 hours old**, and an ending delimiter, **perform ABO/Rh identification OR antibody screen on Patient blood specimen**.
2. The behavior is required to hold from an occurrence of **find that blood specimen is more than 72 hours old**, if it ever occurs, through to the first subsequent occurrence of **perform ABO/Rh identification OR antibody screen on Patient blood specimen**, if it ever occurs.
3. If there are multiple occurrences of **find that blood specimen is more than 72 hours old** without an occurrence of **perform ABO/Rh identification OR antibody screen on Patient blood specimen** in between them, only the first of those occurrences of **find that blood specimen is more than 72 hours old** potentially starts a restricted interval; later occurrences of **find that blood specimen is more than 72 hours old** within this restricted interval do not have an effect.
4. **find that blood specimen is more than 72 hours old** is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if **find that blood specimen is more than 72 hours old** does occur, **perform ABO/Rh identification OR antibody screen on Patient blood specimen** is not required to occur subsequently and if it does not occur subsequently, then the behavior is not required to hold for the remainder of the event sequence.
5. There might be many restricted intervals in the event sequence. If **find that blood specimen is more than 72 hours old** occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of **find that blood specimen is more than 72 hours old**, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a potential new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

BEHAVIOR:
1. The event of primary interest in this behavior is **obtain up-to-date blood specimen**.
2. There are no events of secondary interest in this behavior.
3. **obtain up-to-date blood specimen** is required to occur at least once.

FSA and DNL for Property DBSS.C.1c
**Event alphabet:**
- A: DBSS puts unit of blood product i on the list
- START: determine that unit of blood product i is not ABO/Rh-compatible
- END: determine that unit of blood product i is ABO/Rh-compatible

**SCOPE:**
1. A restricted interval in the event sequence can have both a starting delimiter, determine that unit of blood product i is not ABO/Rh-compatible, and an ending delimiter, determine that unit of blood product i is ABO/Rh-compatible.
2. The behavior is required to hold from an occurrence of determine that unit of blood product i is not ABO/Rh-compatible, if it ever occurs, through to the first subsequent occurrence of determine that unit of blood product i is ABO/Rh-compatible, if it ever occurs.
3. If there are multiple occurrences of determine that unit of blood product i is not ABO/Rh-compatible without an occurrence of determine that unit of blood product i is ABO/Rh-compatible in between them, only the first of those occurrences of determine that unit of blood product i is not ABO/Rh-compatible starts a restricted interval; later occurrences of determine that unit of blood product i is not ABO/Rh-compatible within this restricted interval do not have an effect.
4. Determine that unit of blood product i is not ABO/Rh-compatible is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if determine that unit of blood product i is not ABO/Rh-compatible does occur, determine that unit of blood product i is ABO/Rh-compatible is not required to occur subsequently. Even if determine that unit of blood product i is ABO/Rh-compatible does not occur subsequently, the behavior is still required to hold, until the end of the event sequence.
5. There might be many restricted intervals in the event sequence. If determine that unit of blood product i is not ABO/Rh-compatible occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of determine that unit of blood product i is not ABO/Rh-compatible, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

**BEHAVIOR:**
1. The event of primary interest in this behavior is DBSS puts unit of blood product i on the list.
2. There are no events of secondary interest in this behavior.
3. DBSS puts unit of blood product i on the list is never allowed to occur.

---

**FSA and DNL for Property DBSS.C.2a**
Event alphabet:

- A: obtain up-to-date blood specimen
- START: find that blood specimen is more than 72 hours old
- END: DBSS puts unit of blood product on the list

**SCOPE:**

1. A restricted interval in the event sequence can have both a starting delimiter, `find that blood specimen is more than 72 hours old`, and an ending delimiter, `DBSS puts unit of blood product on the list`.
2. The behavior is required to hold from an occurrence of `find that blood specimen is more than 72 hours old`, if it ever occurs, through to the first subsequent occurrence of `DBSS puts unit of blood product on the list`, if it ever occurs.
3. If there are multiple occurrences of `find that blood specimen is more than 72 hours old` without an occurrence of `DBSS puts unit of blood product on the list` in between them, only the first of those occurrences of `find that blood specimen is more than 72 hours old` potentially starts a restricted interval; later occurrences of `find that blood specimen is more than 72 hours old` within this restricted interval do not have an effect.
4. `find that blood specimen is more than 72 hours old` is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if `find that blood specimen is more than 72 hours old` does occur, `DBSS puts unit of blood product on the list` is not required to occur subsequently and if it does not occur subsequently, then the behavior is not required to hold for the remainder of the event sequence.
5. There might be many restricted intervals in the event sequence. If `find that blood specimen is more than 72 hours old` occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of `find that blood specimen is more than 72 hours old`, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a potential new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

**BEHAVIOR:**

1. The event of primary interest in this behavior is `obtain up-to-date blood specimen`.
2. There are no events of secondary interest in this behavior.
3. `obtain up-to-date blood specimen` is required to occur at least once.

FSA and DNL for Property DBSS.C.2b
**Event alphabet:**
- **A**: perform antibody identification
- **B**: DBSS provides a list of potentially-compatible units of blood product
- **START**: determine that Patient has problematic antibody situation
- **END**: supervisor does an emergency override

**SCOPE:**
1. A restricted interval in the event sequence can have both a starting delimiter, determine that Patient has problematic antibody situation, and an ending delimiter, supervisor does an emergency override.
2. The behavior is required to hold from an occurrence of determine that Patient has problematic antibody situation, if it ever occurs, through to the first subsequent occurrence of supervisor does an emergency override, if it ever occurs.
3. If there are multiple occurrences of determine that Patient has problematic antibody situation without an occurrence of supervisor does an emergency override in between them, only the last of those occurrences of determine that Patient has problematic antibody situation starts a restricted interval; each of those occurrences of determine that Patient has problematic antibody situation resets the beginning of this restricted interval.
4. determine that Patient has problematic antibody situation is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if determine that Patient has problematic antibody situation does occur, supervisor does an emergency override is not required to occur subsequently. Even if supervisor does an emergency override does not occur subsequently, the behavior is still required to hold, until the end of the event sequence.
5. There might be many restricted intervals in the event sequence. If determine that Patient has problematic antibody situation occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of determine that Patient has problematic antibody situation, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

**BEHAVIOR:**
1. The events of primary interest in this behavior are perform antibody identification and DBSS provides a list of potentially-compatible units of blood product.
2. There are no events of secondary interest in this behavior.
3. DBSS provides a list of potentially-compatible units of blood product is not allowed to occur until after perform antibody identification occurs.
4. perform antibody identification is not required to occur.
5. Even if perform antibody identification does occur, DBSS provides a list of potentially-compatible units of blood product is not required to occur after perform antibody identification occurs.
6. After perform antibody identification occurs, but before the first subsequent DBSS provides a list of potentially-compatible units of blood product occurs, perform antibody identification is not allowed to occur again.
7. After perform antibody identification and the first subsequent DBSS provides a list of potentially-compatible units of blood product occur:
   - Both perform antibody identification and DBSS provides a list of potentially-compatible units of blood product are allowed to occur again, zero or more times, and there are no restrictions on the occurrences of any future events.

FSA and DNL for Property DBSS.C.3a
**Event alphabet:**
- A: obtain up-to-date blood specimen
- START: find that blood specimen is more than 72 hours old
- END: perform antibody identification

**SCOPE:**
1. A restricted interval in the event sequence can have both a starting delimiter, **find that blood specimen is more than 72 hours old**, and an ending delimiter, **perform antibody identification**.
2. The behavior is required to hold from an occurrence of **find that blood specimen is more than 72 hours old**, if it ever occurs, through to the first subsequent occurrence of **perform antibody identification**, if it ever occurs.
3. If there are multiple occurrences of **find that blood specimen is more than 72 hours old** without an occurrence of **perform antibody identification** in between them, only the first of those occurrences of **find that blood specimen is more than 72 hours old** potentially starts a restricted interval; later occurrences of **find that blood specimen is more than 72 hours old** within this restricted interval do not have an effect.
4. **find that blood specimen is more than 72 hours old** is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if **find that blood specimen is more than 72 hours old** does occur, **perform antibody identification** is not required to occur subsequently and if it does not occur subsequently, then the behavior is not required to hold for the remainder of the event sequence.
5. There might be many restricted intervals in the event sequence. If **find that blood specimen is more than 72 hours old** occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of **find that blood specimen is more than 72 hours old**, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a potential new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

**BEHAVIOR:**
1. The event of primary interest in this behavior is **obtain up-to-date blood specimen**.
2. There are no events of secondary interest in this behavior.
3. **obtain up-to-date blood specimen** is required to occur at least once.

FSA and DNL for Property DBSS.C.3b
Event alphabet:
- **A**: DBSS puts unit of blood product \( i \) on the list
- **START**: determine that unit of blood product \( i \) contains problematic antigens
- **END**: determine that unit of blood product \( i \) does not contain problematic antigens

**SCOPE:**
1. A restricted interval in the event sequence can have both a starting delimiter, **determine that unit of blood product \( i \) contains problematic antigens**, and an ending delimiter, **determine that unit of blood product \( i \) does not contain problematic antigens**.
2. The behavior is required to hold from an occurrence of **determine that unit of blood product \( i \) contains problematic antigens**, if it ever occurs, through to the first subsequent occurrence of **determine that unit of blood product \( i \) does not contain problematic antigens**, if it ever occurs.
3. If there are multiple occurrences of **determine that unit of blood product \( i \) contains problematic antigens** without an occurrence of **determine that unit of blood product \( i \) does not contain problematic antigens** in between them, only the last of those occurrences of **determine that unit of blood product \( i \) contains problematic antigens** resets the beginning of this restricted interval.
4. **determine that unit of blood product \( i \) contains problematic antigens** is not required to occur and if it never occurs, then the behavior is not required to hold anywhere in the event sequence. Even if **determine that unit of blood product \( i \) contains problematic antigens** does occur, **determine that unit of blood product \( i \) does not contain problematic antigens** is not required to occur subsequently. Even if **determine that unit of blood product \( i \) does not contain problematic antigens** does not occur subsequently, the behavior is still required to hold, until the end of the event sequence.
5. There might be many restricted intervals in the event sequence. If **determine that unit of blood product \( i \) contains problematic antigens** occurs after the end of a restricted interval, then the situation would be the same as after the first occurrence of **determine that unit of blood product \( i \) contains problematic antigens**, meaning that the restrictions described in parts 2, 3, 4, and 5 would again apply for a new restricted interval.
6. There are no restrictions imposed on the occurrences of any events outside of the restricted interval(s).

**BEHAVIOR:**
1. The event of primary interest in this behavior is **DBSS puts unit of blood product \( i \) on the list**.
2. There are no events of secondary interest in this behavior.
3. **DBSS puts unit of blood product \( i \) on the list** is never allowed to occur.

**FSA and DNL for Property DBSS.C.4**
APPENDIX D

DNL TRANSLATION STUDY
D.1 Detailed Study Data

Table D.1. DNL Translation Study Raw Data Summary

<table>
<thead>
<tr>
<th>Property Complexity Category</th>
<th>Total</th>
<th>Exact Match</th>
<th>1 Diff</th>
<th>2+ Diffs</th>
</tr>
</thead>
<tbody>
<tr>
<td>G1: Global scope, 1-event behavior</td>
<td>14</td>
<td>13</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>AB1: After/Before scope, 1-event behavior</td>
<td>6</td>
<td>5</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>B1: Between scope, 1-event behavior</td>
<td>8</td>
<td>1</td>
<td>2</td>
<td>5</td>
</tr>
<tr>
<td>G2: Global scope, 2-event behavior</td>
<td>14</td>
<td>9</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>AB2: After/Before scope, 2-event behavior</td>
<td>8</td>
<td>2</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>B2: Between scope, 2-event behavior</td>
<td>6</td>
<td>0</td>
<td>1</td>
<td>5</td>
</tr>
</tbody>
</table>

Table D.1 summarizes the raw data given in Table D.2 in terms of the number of differences between the participants’ FSAs and the associated Propel FSAs, and presents this information for each of the property complexity categories that are defined in Table 6.1. The data in Table D.1 is similar to what is given in Table 6.2, but there are three differences. In Table D.1, the data for the G1 property complexity category is given, the number of participant FSAs that were considered “close” (see the definition given in Section 6.1), but were not an exact match, is given in the 1 Diff column, and the number of participant FSAs that were not considered “close” is given explicitly, in the 2+ Diffs column.

Table D.2 gives the raw data for each of the properties that participants translated from DNL to an FSA. Each property specification’s unique ID is composed of two parts, “X.Y”, where X is the ID of the translation packet that the property specification was a part of\(^1\), and Y is the order in which the property specification appeared in its translation packet. The Complexity column describes the property complexity category, using the abbreviations given in Table D.1, above. The Transition Diffs column gives the number of transition differences between the participant’s FSA and the associated Propel FSA, where a transition difference is defined to be an outgoing transition whose destination state is not the same in the two FSAs. The State Diffs column gives the number of state differences between the participant’s FSA and the associated Propel FSA, where a state difference is defined to be a state that is found in both FSAs, but whose accepting status is not the same in both FSAs. The Notes column describes situations where there is no clear mapping between all the participant FSA’s states and all the associated Propel FSA’s states. Such participant FSAs were never considered an exact or a “close” match.

Table D.2. DNL Translation Study Raw Data

<table>
<thead>
<tr>
<th>Property ID</th>
<th>Complexity</th>
<th>Transition Diffs</th>
<th>State Diffs</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1</td>
<td>G1</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

\(^1\)Since 40 translation packets were created in the hopes that more people would participate in the study, the actual packet IDs were in the range 1..40. For simplicity in Table D.2, the 14 translation packets that were used are re-numbered from 1..14.
Table D.2. DNL Translation Study Raw Data (Continued)

<table>
<thead>
<tr>
<th>Property ID</th>
<th>Complexity</th>
<th>Transition Diffs</th>
<th>State Diffs</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.2</td>
<td>AB1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.3</td>
<td>G2</td>
<td>2</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>1.4</td>
<td>B2</td>
<td></td>
<td></td>
<td>missing state; 2 states don’t map to anything in Propel FSA</td>
</tr>
<tr>
<td>2.1</td>
<td>G1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2.2</td>
<td>B2</td>
<td>3</td>
<td></td>
<td>missing behavior trap state</td>
</tr>
<tr>
<td>2.3</td>
<td>AB1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2.4</td>
<td>G2</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3.1</td>
<td>G1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3.2</td>
<td>AB1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3.3</td>
<td>G2</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3.4</td>
<td>B2</td>
<td>3</td>
<td>1</td>
<td>missing behavior trap state</td>
</tr>
<tr>
<td>4.1</td>
<td>G1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4.2</td>
<td>B2</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4.3</td>
<td>G2</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4.4</td>
<td>AB1</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>5.1</td>
<td>G1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5.2</td>
<td>AB1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5.3</td>
<td>G2</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5.4</td>
<td>B2</td>
<td>4</td>
<td></td>
<td>missing behavior trap state</td>
</tr>
<tr>
<td>6.1</td>
<td>AB1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6.2</td>
<td>B2</td>
<td>5</td>
<td>1</td>
<td>missing state</td>
</tr>
<tr>
<td>6.3</td>
<td>G1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6.4</td>
<td>G2</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>7.1</td>
<td>G1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7.2</td>
<td>B1</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>7.3</td>
<td>G2</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7.4</td>
<td>AB2</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8.1</td>
<td>G1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8.2</td>
<td>B1</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>8.3</td>
<td>G2</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8.4</td>
<td>AB2</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9.1</td>
<td>G1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9.2</td>
<td>B1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9.3</td>
<td>G2</td>
<td>3</td>
<td></td>
<td>extra state</td>
</tr>
<tr>
<td>9.4</td>
<td>AB2</td>
<td>4</td>
<td></td>
<td>extra state</td>
</tr>
<tr>
<td>10.1</td>
<td>G1</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>10.2</td>
<td>B1</td>
<td>3</td>
<td></td>
<td></td>
</tr>
<tr>
<td>10.3</td>
<td>G2</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>10.4</td>
<td>AB2</td>
<td>3</td>
<td></td>
<td></td>
</tr>
<tr>
<td>11.1</td>
<td>G2</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>11.2</td>
<td>B1</td>
<td>1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>11.3</td>
<td>AB2</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>11.4</td>
<td>G1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>12.1</td>
<td>G1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>12.2</td>
<td>B1</td>
<td>6</td>
<td></td>
<td>missing behavior trap state</td>
</tr>
<tr>
<td>12.3</td>
<td>G2</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>12.4</td>
<td>AB2</td>
<td>1</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Continued on next page
<table>
<thead>
<tr>
<th>Property ID</th>
<th>Complexity</th>
<th>Transition Diffs</th>
<th>State Diffs</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>13.1</td>
<td>G2</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>13.2</td>
<td>AB2</td>
<td>4</td>
<td></td>
<td>missing state</td>
</tr>
<tr>
<td>13.3</td>
<td>B1</td>
<td>6</td>
<td></td>
<td>2 missing states, 1 is the behavior trap state</td>
</tr>
<tr>
<td>13.4</td>
<td>G1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>14.1</td>
<td>G1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>14.2</td>
<td>B1</td>
<td>3</td>
<td></td>
<td></td>
</tr>
<tr>
<td>14.3</td>
<td>G2</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>14.4</td>
<td>AB2</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
D.2 FSA Description and Example Property Translation Sheet

Some Information That You May Find Useful

What Is a Property?
A property is a concise description of how some aspect of a system should behave. We can see how a system behaves by observing the sequences of events that it generates. Thus, to describe a particular aspect of the desired system behavior, we can create a property that restricts the generated sequences to just those that are acceptable. Each property is composed of two parts: a behavior and a scope. The behavior specifies the restrictions on the occurrences of events, and the scope specifies the parts of the event sequences within which those restrictions apply.

Formal Definition of a Finite-State Automaton (FSA) Representation of a Property
If a sequence of events violates the restrictions made by a property, then the FSA representation of that property does not accept that sequence. The FSA representation of a property is defined to be the tuple \(<S, s, A, \Sigma, \delta>\), where 
- \(S\) is the finite set of states,
- \(s \in S\) is the unique start state,
- \(A \subseteq S\) is the set of accepting states,
- \(\Sigma\) is the event alphabet, and
- \(\delta \subseteq S \times \Sigma \times S\) is a transition relation that must be deterministic (that is, for \(s_i, s_k, s_l \in S\) and \(e_j \in \Sigma\), if \(\delta(s_i, e_j, s_k)\) and \(\delta(s_i, e_j, s_l)\) then \(s_k = s_l\)). A sequence of events \(e_1 e_2 ... e_n \in \Sigma^*\) is accepted by the FSA if a sequence of states \(s_0 s_1 ... s_n\) exists in \(S\) such that:
1. \(s_0 = s\) (the sequence of states must begin in the start state),
2. \(s_n \in A\) (the sequence of states must end in an accepting state), and
3. \(\delta(s_i, e_i, s_{i+1})\) for all \(i \in [0 ... n-1]\) (given the current state, follow the outgoing transition labeled with the next event to get to the next state).

A property’s alphabet, \(\Sigma\), is the set of events that are relevant to the meaning of the property. For each property description that you are asked to translate into an FSA, the alphabet is the following set of events: \{A, B, C, D, E\}.

An Example Graphical Depiction of an FSA

In the example FSA given above, the FSA accepts the sequence A B E A. Conversely, the FSA does not accept the sequence A B.

If the current state has no outgoing transition labeled with the next event in the sequence, then the FSA does not accept that sequence. For example, the sequence A D would violate the property above, since state 2 has no outgoing transition labeled with event D.

An Example Property Translation

ALPHABET OF EVENTS: \{A, B, C, D, E\}

PROPERTY DESCRIPTION:

scope:
After the first occurrence of C, the behavior must hold. C is not required to occur and if C does not occur, then the behavior is not required to hold. If C does occur, then there are no restrictions imposed on the occurrences of any events before the first occurrence of C.

behavior:
A occurs one or more times.

A POSSIBLE FSA TRANSLATION:

Figure D.1. FSA Description and Example Property Translation Sheet Given to Study Participants
D.3 Example Property From a Participant’s Translation Packet

ALPHABET OF EVENTS: \(\{A, B, C, D, E\}\)

PROPERTY DESCRIPTION:

**scope:**
The behavior must hold between the first occurrence of \(C\) and the first subsequent occurrence of \(D\). Further occurrences of \(C\) after that first occurrence of \(C\) do not have an effect.

\(C\) is required to occur and \(D\) is required to subsequently occur.

There are no restrictions imposed on the occurrences of any events except for those restrictions stated above.

---------------

**behavior:**

\(A\) causes \(B\) to occur.

\(A\) is required to occur. Before the first \(A\) occurs, the events in the alphabet of this property, including \(B\), can occur any number of times.

After \(A\) occurs and before the first subsequent \(B\) occurs:
- neither \(A\) nor any other events in the alphabet of this property can occur.

After the first subsequent \(B\) occurs:
- the events in the alphabet of this property, other than \(A\) or \(B\), could occur any number of times;
- \(B\) cannot occur again until after another \(A\) occurs;
- \(A\) can occur and if it does, then the situation should be regarded as exactly the same as when the first \(A\) occurred, meaning that all restrictions described on the events would again apply.

FSA TRANSLATION:

**START TIME:**

Figure D.2. Example Property Translation Sheet
BIBLIOGRAPHY


[104] Erich Gamma, Richard Helm, Ralph Johnson, and John Vlissides. Design Patterns: Elements of Reusable Object-Oriented Software. Addison-Wesley, 1995.


[266] Zijiang Yang, Christine Chung, and In-Ho Moon. FormalCheck query language compared with CTL, 1999.


