Chapter 2.24
Modeling Variant User Interfaces for Web-Based Software Product Lines

Suet Chun Lee
BUSINEX, Inc., USA

ABSTRACT

Software product line (SPL) is a software engineering paradigm for software development. SPL is important in promoting software reuse, leading to higher productivity and quality. A software product within a product line often has specific functionalities that are not common to all other products within the product line. Those specific functionalities are termed "variant features" in a product line. SPL paradigm involves the modeling of variant features. However, little work in SPL investigates and addresses the modeling of variant features specific to UI. UML is the de facto modeling language for object-oriented software systems. It is known that UML needs better support in modeling UIs. Thus, much research developed UML extensions to improve UML support in modeling UIs. Yet little of this work is related to developing such extensions for modeling UIs for SPLs in which variant features specific to user interfaces (UI) modeling must be addressed. This research develops a UML extension, WUIML, to address these problems. WUIML defines elements for modeling variant features specific to UIs for Web-based SPLs. The model elements in WUIML extend from the metaclass and of the UML2.0 metamodel. WUIML integrates the modeling of variant features specific to UIs to UML. For example, in a Web-based patient registration SPL, member products targeting British users may use British date format in the user interface, while member products targeting United States users may use United States date format in the user interface. Thus, this is a variant feature for this product line. WUIML defines a model element, XOR, to represent such exclusive or conditions in a product line user interface model. WUIML would reduce SPL engineers' efforts needed in UI development. To validate the WUIML research outcome, a case study was conducted. The results of this empirical study indicate that modeling UIs for Web-based SPLs using WUIML is more effective and efficient than using standard UML.

INTRODUCTION

Software product line (SPL) (Chastek, Donohoe, Kang, & Thiel, 2001; Clements & Northrop, 2002; SEI, 2005a) is a software engineering paradigm to develop software products. One important step in the SPL paradigm is the modeling of the functional features of software products across the product line. The features are called common core. An even more important step in the SPL paradigm is the modeling of the specific functional features within a particular member product in a product line. These specific functional features are called variant features because they are the features that differentiate member products in the product line. Then based on the model, a product is "assembled" by reusing the common core and selected variant features.

Unified Modeling Language (UML) (OMG, 2003b, 2004; Rumbaugh, Jacobson, & Booch, 2005) is a standard object-oriented modeling language. UML includes multiple views and diagram types to capture software functionalities from user perspective. However, UML seems to have not been developed for modeling user interface specific issues (Kovacevic, 1998; Silva & Paton, 2003). One of the usages of user interface models is that, in model-based user interface management systems (MB-UIMSs) (Griffiths et al., 2001; Szekely, Sukaviriya, Castells, Muthukumarasamy, & Salcher, 1996), user interface models can be used to generate user interface codes. There are extensions of UML (Blankenhorn & Jeckle, 2004; Nunes, 2003; Silva, 2002) to make UML better support user interface modeling. Yet, these extensions often assume the modeling of a single system instead of a SPL. On the other hand, although standard UML (OMG, 2003b, 2004) seems to have not been developed to support the modeling of SPLs, there are works (Gomaa, 2004; Gomaa & Gianturco, 2002; Ziadi, Hélouët, & Jézéquel, 2004) on extending UML to improve UML supports in modeling SPLs. Yet, these works do not focus on user interface modeling. Currently, many software products are Web-based. However, some (Silva, 2002) observe that there are specific modeling challenges for modeling user interfaces of Web-based software systems.

Thus, it is not clear how to model variant features for user interface specific issues in Web-based software product lines. This is an important barrier to overcome if software product line development of Web-based products is to take greater advantage of software reuse objectives: increased quality, decreased effort, or decreased time to market. Therefore, this paper is concerned with reporting research about developing a UML extension, Web User Interface Modeling Language (WUIML) that decreases effort by increasing effectiveness and efficiency needed in using UML to model user interfaces for Web-based software product lines. User interface development has been found (Myers, 1989) to account for a significant amount of overall software development work. WUIML would improve SPL software engineering paradigm in the user interface development perspective by reducing engineers' efforts needed in user interface modeling.

BACKGROUND AND RELATED WORK

Unified Modeling Language

Unified Modeling Language (UML) (Booch, Rumbaugh, & Jacobson, 1999; OMG, 2003b, 2004; Scott, 2004) is a graphical language for specifying software systems. UML is a standard of the Object Management Group (OMG; see http://www.omg.org). The most current version for UML is UML 2.0 (OMG, 2003b, 2004). This research considers UML in UML 2.0 context.

UML is a standardized notation for object-oriented development. UML consists of views, diagrams, model elements, and general mechanisms. Views are used to present different aspects of complex systems from both the "system" in the