Chapter 16
Applying UML Extensions in Modeling Software Product Line Architecture of a Distribution Services Platform

Liliana Dobrica
University Politehnica of Bucharest, Romania

Eila Ovaska
VTT Technical Research Centre of Finland, Finland

ABSTRACT

UML provides the means to use specific variation mechanisms to describe hierarchical systems. However, it does not support a description of variation, as it is required for service architecture. UML built-in extension mechanisms refine its specification. This chapter presents the extensions of the UML for representing variations in the software product line architecture of middleware services. The product line is defined as a middleware services framework that includes several products. The products realize different functionality by using various modern software technologies of spontaneous networks. Architecture design produces descriptions at two abstraction levels from multiple viewpoints. The modeling of service architectures benefits from a more familiar and widely used notation that improves stakeholders' understanding of the architectural artifacts. A standard based notation also enables more extensive tool support for manipulating architecture models.

INTRODUCTION

For several years the focus of our research has been product line architecture design and analysis. An important goal is to define a method for modeling software product line architecture of a distribution services platform. An essential issue is to explicitly represent variation and indicate locations for which changes are allowed in design. In this way, the diagrammatic description of the product line architecture defined by using the method helps in instantiating it for a particular product or in its evolution for future use. From the product line architecture documented diagrammatically, it is easy to detect what kind of modifications, omis-

DOI: 10.4018/978-1-61692-874-2.ch016

sions and extensions are permitted, expected or required. Initially, the method can be described by defining and using a framework that consisted of, among other ingredients, an underlying model, referring to the kinds of constructs are represented, manipulated and analyzed by the model and, a language, which is a concrete means of describing the constructs, considering possible diagrammatic notations. In order to achieve an optimal method, these ingredients can be defined or selected more properly. Some of them may already be available, from the literature, tool vendors or as open source distribution, whereas others may have to be specially developed.

Nowadays a considerable interest exists in applying UML to modeling the reference architecture of a software product line. The work in this chapter presents UML extensions for the management of variability in the space of software product line architectures. The extensions are described based on several predefined architectural views and exemplified on a case study. The aim is to introduce and utilize new constructs that indicate variability and represent a profile of the extended UML concepts intended primarily for use in modeling product line architectures. The new constructs are used together with other UML concepts of superstructure to provide a complete modeling tool set.

The case study that is exemplified to validate the UML notation extensions for architectural designs is represented by a set of four software products. The goal is to model the architecture of a software framework for distributed middleware services that includes common and variable features. Moreover, the result, the model, takes account of not only the known styles and design patterns, but also separation of concerns, variability sources, and locality criteria. By separating different concerns in distinct views and identifying different variability sources we manage complexity and reduce the risk of non-anticipation variation. Locality criteria facilitate determining what information goes into architecture model and what implementation specific is.

Our case study defines and applies concepts and principles about product line architecture modeling as a state of art research needed for theory development to establish the basis for understanding the research domain. The definition of middleware as "a variety of distributed computing services and application development supporting environments that operate between the application logic and underlying system" (Charles, 1999) is similar to software product line definition that is "a set of products sharing a common, managed set of features that satisfy the specific needs of a particular mission" (Clements & Northrop, 2000, 2002). Thus, building the architecture of a framework for distributed middleware services is equivalent to modeling product line architecture. Distributed computing services represent the shared common, managed set of features that satisfy the specific distribution needs.

This chapter is organized as follows. As a background, in the beginning we discuss concepts related to architecture modeling and variability. Our focus is on different types of variability that are visible in an architecture description. Next section introduces our perspective on modeling variability by UML extensions for service architecture description. Then we introduce the case study to illustrate our approach in modeling variability for product line architecture of distributed middleware services. Finally, in the last section, we provide some future research directions and concluding remarks.

BACKGROUND

Architecture Description

Components and services. The architecture of any software system is defined in terms of components and interactions among those components. Product line architecture is defined in the same