An Approach to Defining Scope in Software Product Lines for the Telecommunication Domain

Radovan Cvetković[1] and Siniša Nešković[2]

[1] Telekom Srbija a.d., Technical Affairs Division,
Bulevar umetnosti 16a, 11000 Belgrade, Serbia
radovan.cvetkovic@telekom.rs
[2] University of Belgrade, Faculty of Organizational Sciences,
"Branislav Lazarevic" Laboratory for Information Systems,
Jove Ilica 154, 11000 Belgrade, Serbia
sinisa.neskovic@fon.bg.ac.rs

Abstract. The Next Generation Operations Systems and Software (NGOSS) is a solution framework for the development of Operations Support System/Business Support Systems (OSS/BSS) in telecom companies. This paper presents an approach to OSS/BSS building which is based on a specific combination of Software Product Lines Engineering (SPLE) and NGOSS. The focus of this paper is the first phase in SPLE which deals with the identification and scoping of software product families required to build an OSS/BSS. We present a generic architecture of required product families as well as a methodological procedure for their identification and scoping. Both are based on Enhanced Telecom Operation Map (eTOM), a process framework defined within NGOSS.

Keywords: Software Product Lines, Software Family, NGOSS, eTOM, OSS/BSS.

1 Introduction

The Operations Support System/Business Support Systems (OSS/BSS) represents a very complex information system which integrates business and technical subsystems of a telecommunication (telecom) company into a functionally coherent whole. TeleManagement Forum, an international telecom industry association, has developed a solution framework for the efficient building of OSS/BSS, which is called Next Generation Operations Systems and Software (NGOSS) [1]. Essentially, NGOSS represents a reference enterprise architecture framework for the telecom domain consisting of a set of reference models, methods and guidelines for OSS/BSS development. The Enhanced Telecom Operation Map (eTOM) is a business process reference model within NGOSS which identifies and categorizes all business activities that a telecom service provider will use. The eTOM framework supports two different perspectives of process groupings: 1) Horizontal process groupings represent a view of functionally related processes,

B. Catania, M. Ivanović, and B. Thalheim (Eds.): ADBIS 2010, LNCS 6295, pp. 555–558, 2010.

which represent core business functions within the telecom business domain, and which are used as basic units in the automation of end-to-end processes; 2) Vertical process groupings represent a view of end-to-end processes within the business which effectively support customer needs in a total. These vertical end-to-end process groupings are essentially crosscutting overlays onto the hierarchical top level horizontal groupings.

Due to NGOSS complexity, a high level abstraction of its reference models and informal development methods and guidelines, a fruitful utilization of NGOSS is very hard to achieve in practice. This paper presents an approach to OSS/BSS development based on a specific combination of Software Product-Line Engineering (SPLE) principles and NGOSS. SPLE is a method which creates software product lines as development platforms for (largely automated) production of a family of software products [2], [3]. The main idea is to employ SPLE and exploit NGOSS for the development of a range of software product lines which are collectively capable of producing an OSS/BSS tailored to the specific needs of a particular telecom company, similarly to the idea expressed in [4]. The focus of this paper is the first phase in SPLE which deals with the identification and scoping of software product families.

The rest of the paper is organized as follows. In Section 2 we introduce a generic architecture of software product families required to build an OSS/BSS. A methodological procedure for their identification and scoping is given in Section 3. The paper concludes with the paper's main contributions and a discussion related to our future work.

2 Generic Architecture of Software Product Families

A proposed generic architecture of product families in the telecom domain is given in Fig. 1 as a metamodel in the form of an UML class diagram. Derived from the eTOM framework, the metamodel identifies the types of product families, represented as UML classes in the diagram, as well as their mutual relationships, represented as UML associations.

OSS/BSS class represents a type of product family whose instances are families used to produce individual OSS/BSS systems tailored to specific needs of a particular telecom company. Due to their complexity, OSS/BSS family members are not built as monolithic software applications, but as complex software systems composed of members from other product families supporting particular telecom domain aspects.

These families are structured following eTOM into two distinct types: *Business Domain* for family types which support core telecom business functions (eTOM horizontal processes), and *End-to-End process* for family types supporting eTOM vertical processes. Information about which particular families of *Business Domain* and *End-to-End Process* families constitute a particular *OSS/BSS* family member is captured by *Has Domains* and *Has Processes* aggregations. Similarly, which *Business Domain* families are used by a particular *End-to-End Process* family is captured by the *Uses* aggregation. It is important

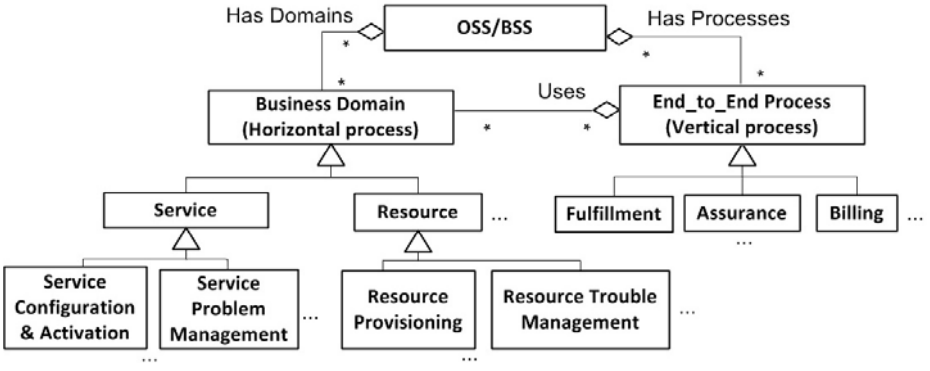

Fig. 1. Metamodel of Architecture of Software Product Families for the Telecom Domain

to observe that both *Business Domain* and *End-to-End Process* family types can be further specialized (subtyped) into family types supporting particular subdomains (illustrated in Fig. 1). Subtyping in this context means that each subtyped family type supports an additional functionality to its parent family type.

Since each family requires a separate product line for the production of its members, the generic architecture must be concretized (with classes and their instances) before the development of production lines is possible.

3 Identification and Scoping of Software Product Families

The concretization of the generic architecture is done through the identification and scoping of products families, defined as a separate process distinguished from the traditional Domain Engineering and Application Engineering processes [2].

Identification and scoping of software product families, given in Fig. 2 as an UML Activity Diagram, consists of the following activities:

- **eTOM to Feature Models Transformation** produces a set of feature models which are more convenient for further analysis
- **Software Product Families Type Identification** uses *eTOM feature models* to produce *Software Product Family Types Specification.* (i.e. it specifies classes in the metamodel of the generic architecture)
- **Software Product Families Identification** activity results with *Software product family specification* (i.e. it identifies instances of classes in the metamodel of the generic architecture)
- **Software Product Families Scope Definition** activity defines the high level scope for product families (i.e. defines commonalities of the identified product families)
- **Software Product Families Variability Definition** activity identifies the differences between members of the family and defines this variability

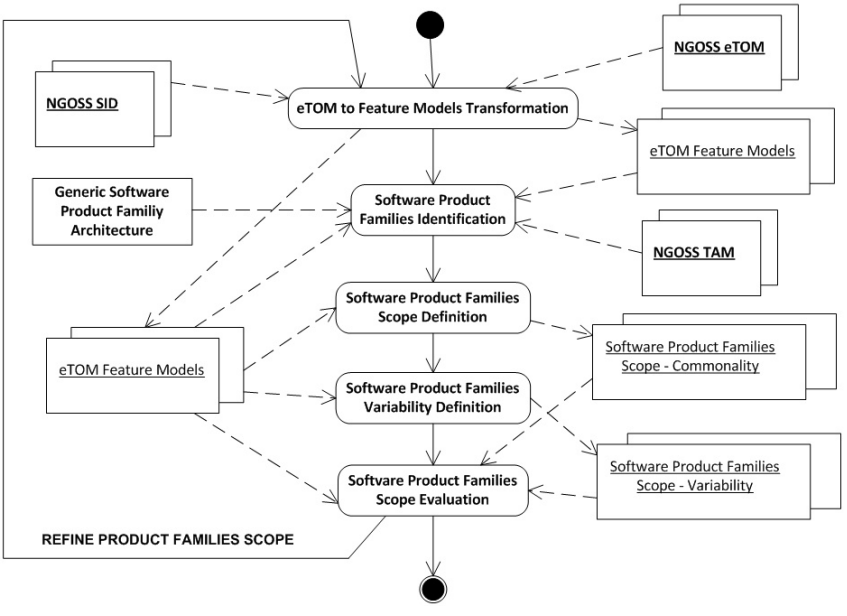

Fig. 2. Identification and Scoping Process

4 Conclusion

The presented approach augments traditional SPLE with: 1) the generic architecture of software product families for the telecom domain; 2) the method for the identification and scoping of product families.

The future work is related to the design and implementation of product lines for building these defined product families. The main challenge here is related to the realization of such an extremely complex software architecture consisting of a large number of mutually related product lines.

References

1. Reilly, J.P., Creaner, M.J.: NGOSS distilled: The Essential Guide to Next Generation Telecoms Management. TM Forum, UK (2005)
2. Pohl, K., Böckle, G., van der Linden, F.: Software Product Line Engineering Foundations, Principles and Techniques. Springer, Heidelberg (2005)
3. van Ommering, R.: Software Reuse in Product Populations. IEEE Trans. Software Eng. 31(7) (2005)
4. Bosch, J.: Using Software Product Families: Towards Compositionality. In: Dwyer, M.B., Lopes, A. (eds.) FASE 2007. LNCS, vol. 4422, pp. 1–10. Springer, Heidelberg (2007)