Towards a Framework for Collaborative and Coordinated Product Configuration

Marcilio Mendonca
University of Waterloo, Canada
marcilio@cs.uwaterloo.ca

Toacy Oliveira
PUC-RS, Brazil
toacy@inf.pucrs.br

Krzysztof Czarnecki
University of Waterloo, Canada
czarnecki@acm.org

Donald Cowan
University of Waterloo, Canada
dcowan@csg.uwaterloo.ca

Abstract
In system-family approaches, product configuration is the activity of selecting the features desired for a software product. Although this process is typically collaborative this aspect has long been neglected. On the other hand, enabling collaborative product configuration brings new and challenging problems such as the proper coordination of configuration decisions. This paper introduces a framework for collaborative configuration that addresses the major issues that arise in this context. Some aspects of the framework can be customized to accommodate specific collaboration goals.

Categories and Subject Descriptors D.2.1 Requirements/Specifications:Tools.

General Terms Design, Human Factors, Verification.

Keywords collaborative product configuration, feature modeling, software product lines.

1. Introduction
In system-family approaches such as software product lines [1] product derivation is achieved by means of customization of reusable core assets. Variability models (e.g. feature models [2]) are used as key elements to represent the commonalities and variabilities of a product family. More importantly, such models serve as basis for product configuration, i.e., the selection of the features that will be available in the final version of the product. For instance, in feature models product features are organized hierarchically offering stakeholders a common ground for configuration decisions. As a result of the configuration process, configuration models are produced containing a list of the desired product features. Configuration models are especially important in supporting the automation of production processes [3].

Product configuration is typically a collaborative process in which a group of decision makers with particular interests and perspectives on the product under development interact and negotiate configuration decisions. The benefits of teamwork include better knowledge management allowing dispersed skillful people to participate in the configuration process and improved legitimacy of the decision-making process as decisions are made by proper qualified parties.

1.1 Problem Statement
Though the notion of collaboration is fully applicable in the context of software product configuration, too little attention has been devoted to this aspect. Current product configuration approaches rely on the application engineer to properly interpret and translate user requirements into correct configuration choices. This process is error-prone and risky as misinterpretations in requirements can lead to inconsistent configuration decisions and thus invalid software products. On the other hand, supporting collaboration is also challenging as it introduces new elements of complexity. In the following we list the key questions that arise in this context.

• Which kind of models can be used to support a collaborative product configuration perspective?
• What types of decision conflicts are expected to occur?
• How can decision conflicts be identified and resolved? Can conflict detection be automated?
• How can reasoning on collaborative configuration be properly supported? Which aspects of collaborative configuration are worth measuring?
• What kind of tool support can be provided?

2. Proposed Research
The research proposed aims at developing a framework to support collaborative and coordinated product configuration. The framework will provide a set of components, some of which can be customized to suit particular collaboration goals. For instance, a pessimistic approach to decision conflicts may fit well in a scenario where a high number of conflicts are expected and when the collaboration process is performed asynchronously. Oppositely, an optimistic approach can be used in a synchronous low-conflicting scenario. The framework combines concepts of the realms of software product lines [1] and computer-supported cooperative work (CSCW) [10].

2.1 Expected Contributions
The research proposes a new perspective on product configuration that promotes collaboration and coordination throughout the configuration process. The major expected contributions are listed below.

• A framework for product configuration collaboration that allows for the customization of its components to suit particular collaboration goals.
• The specification of metrics for collaborative configuration that includes measuring aspects such as decision power and coordination complexity.
• Support for collaborative configuration reasoning.
• The specification of algorithms for automatic conflict detection, classification and resolution.
• The development of software tools to support the proposed collaborative configuration meta-model concepts.

2.2 Research to Date
We started our research by studying the use of process languages in the context of object-oriented application frameworks [8]. In particular, we ran and reported a case study on the use of RDL [6] to describe the instantiation steps of frameworks. We then proposed extensions to the RDL process language to support aspect-oriented abstractions [4]. However, motivated by the applicability of our background in a more advantageous context, i.e., software configuration, and encouraged by preliminary successful results on staged configuration [9], we decided to concentrate our efforts on enabling a collaborative product configuration process. Currently, we have identified and exploited some of the major components of the product configuration collaboration framework. Thus, we have developed a preliminary version of the approach [5] that focuses on decision conflicts detection and resolution. An algorithm to identify conflicts and generate process models from annotated feature models has been developed in Java along with data structures to represent feature models, configuration models, decision sets, decision roles, and process models. The immediate goal was to produce a simple tool to investigate the various nuances of collaboration models through case studies.

3. The Framework
The proposed framework captures the common and variable elements that apply in the context of collaborative product configuration. The major elements identified so far are described as follows.

Meta-model: A meta-model is provided adapting concepts of the fields of product configuration and computer-supported cooperative work. The meta-model is described using the Meta-Object Facility specification [11] and is used to scope and validate product configuration collaboration models.

Vocabulary: A vocabulary with key concepts and definitions is provided complementing the meta-model specification. Examples of concepts described include feature, feature dependency, configuration decision, decision roles, decision spaces, decision conflicts, configuration stages, and so forth.

Metrics and Reasoning: The framework will also propose metrics for collaborative configuration to allow measuring aspects such as decision power and coordination complexity. As well, support for reasoning on collaborative configuration using propositional logic [12] will be exploited.

Algorithms: The framework offers algorithms for conflict detection, classification and resolution. A preliminary version of the algorithms has already been detailed in [5].

Tool support: A support tool for drawing and decorating feature models with collaboration annotations is under development. The architecture is being implemented as an object-oriented application framework which facilitates future extensions. Collaboration requirements may be accommodated by configuring runtime parameters (no compilation required) or by extending the framework hotspots. The support tool is being developed as an Eclipse [7] plug-in.

3.1 Tuning the Framework’s Parameters
As we mentioned, some components of the proposed framework are customizable. Examples of such components include:

• The refinement and/or addition of model constraints to specify how features are grouped into configuration decision spaces (e.g. constraints to specify that a decision space must be a well-formed sub-tree or must have at least one open decision)
• The indication of the desired strategy for decision conflict management (e.g. lock-based, early conflict detection, configuration merge)

4. Conclusion
In this paper we described our research on collaborative product configuration. The research aims at specifying a framework to support collaboration and coordination over product configuration activities. The framework is customizable which allows particular collaboration goals to be accommodated.

5. Acknowledgements
This work has been partially supported by the Natural Sciences and Engineering Research Council (NSERC) and CAPES (Brazilian Ministry of Education Agency).

6. References