IDEA GROUP PUBLISHING
701 E. Chocolate Avenue, Suite 200, Hershey PA 17033, USA
Tel: 717/533-8845; Fax 717/533-8661; URL-http://www.idea-group.com
ITP4711

Tracing Variability in Software Product Families

Kannan Mohan
Georgia State University
Department of Computer Information Systems
9th Floor, 35 Broad Street, Atlanta GA, 30303
Phone: (404) 651 3869, Fax: (404) 651 3842
kmohan@cis.gsu.edu

ABSTRACT

Significant economic benefits can be achieved by software development organizations involved in customized software product development by adopting a product family approach. Managing variability, which refers to how members of a product family differ from each other, demands extensive process knowledge. Traceability, the capability to link various conceptual and physical artifacts, from requirements to other outputs of the development process, has been recognized as one of the prominent approaches that can be used to support knowledge intensive processes. A common problem in establishing traceability is identifying and capturing the relevant knowledge elements that will be useful in later phases of the development life cycle. Here, I investigate the use of traceability in managing variability in software product family development. This research focuses on the development of a traceability model for managing variability in product families. This model will identify the various knowledge elements that can be captured and that will be useful in managing variability. I adopt a three-phased approach in this dissertation. In the first phase, I use grounded theory method to develop a traceability model. In the second phase, I develop a prototype software system that is used to capture and use the knowledge represented by the traceability model. In the third phase, a laboratory experiment will be conducted to evaluate the usefulness of this traceability knowledge.

1. INTRODUCTION

Significant economic benefits can be achieved by software development organizations involved in customized software product development by adopting a software product family approach. Product family development is considered to be a knowledge intensive process (Bronsword and Clements 1996). The development process involves managing variations among different members of the product family by identifying common and variable aspects in the domain under consideration. Variability refers to how members of a product family differ from each other.

Prior research (Conklin and Begeman 1988) has suggested a variety of approaches to manage process knowledge for software development. Traceability, the capability to link various conceptual and physical artifacts, from requirements to other outputs of the development process, has been identified as one of the most prominent approaches that can used to support knowledge intensive processes (Gotel and Finkelstein 1994).

Motivated by the effectiveness of traceability in addressing the problems in knowledge intensive processes and the need for process knowledge management in handling variations in product families, **I use traceability to manage variability in software product family development.**

2. RESEARCH OBJECTIVES

The primary objectives of this dissertation are to:

- Develop a traceability model for variability management in the development of product families
- Build a prototype software system that will facilitate the capture and use of traceability knowledge
- Validate the usefulness of the knowledge represented by the traceability model, in managing variability in software product families

These objectives lead to the following research questions:

4. How do we trace requirements that demand variations to the variability points in the system? What knowledge about variability do we need to capture?
5. What are the capabilities that should be provided in a prototype software system for managing variability using traceability knowledge?
6. How do we assess the usefulness of the traceability knowledge in variability management in software product family development? Does this traceability knowledge represented by the model enhance maintenance performance for software product families?

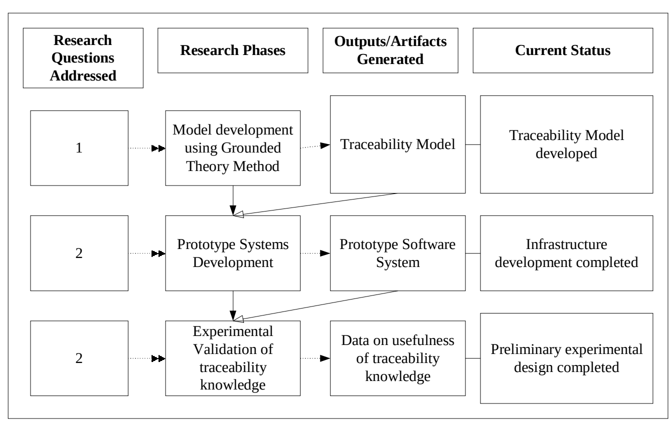
3. RESEARCH METHODOLOGY

3.1 Phase 1: Traceability Model Development using Grounded Theory Method

To get an insight into the various pieces of process knowledge that are to be captured and related to each other, we need to understand how developers of a product family handle variability and various problems associated with it. I use Grounded Theory Method (Strauss and Corbin 1990) to develop a traceability model comprising of concepts representing knowledge elements related to variability. It includes appropriate relationships among the various concepts identified. Data collection and analysis were done iteratively, and analysis was progressively directing collection. Data collected using multiple methods were used for triangulation.

Grounded theory approach is an inductive theory development methodology. Since there has not been extensive research done on identifying the knowledge elements to be captured and linked with each other to facilitate effective variability management in software product families, a generative approach like grounded theory method is suitable. Also, grounded theory ap-

Figure 1: Research Phases



proach is an appropriate method to generate theory rooted in practice. Here, I develop a traceability model that is grounded in practice, and that is reflective of what is done and needed by product family developers for variability management. Data collection was done from two sites, which facilitated comparison and elaboration of concepts that emerged from one to those from the other.

3.1.1 Description of the sites

One of the data collection sites is a software development firm (WMSCo) involved in the development of a warehouse management system (WMS) that is used to handle inventory sent to and from warehouses depending on orders from customers. WMSCo has developed a 'base' version of the product, which encompasses most of the functionality required by its entire customer base.

The second site is an organization (ECTCo) that develops software for electronically controlled diesel engines. They develop a core software build that can be used in different types of engines with customizations.

Six developers were interviewed from WMSCo. Two developers and a project manager were interviewed from ECTCo. The interviews were semi-structured and focused on how developers manage variability and the various issues faced by them. The total duration of all the interviews is about 34 hours. Project-related documents were also used as data sources. The purpose of these interviews was to get insights into the critical aspects of variability management and the specific needs faced by product family developers to manage knowledge about variability. Subjects were questioned about variability scenarios that they had handled in the past, and the needs for variability-related knowledge. The interviews were selectively transcribed and coded. Data analysis was done using open coding, axial coding and selective coding (Strauss and Corbin 1990) in an iterative and overlapped fashion. This process of iterative coding led to the generation of the traceability model discussed below.

The nodes in Figure 2 represent physical or conceptual objects involved in product family development. The links between them represent traceability between them. Variability demanded by variable requirements is made explicit by variation points incorporated in various design objects. These variation points are implemented using appropriate techniques or mechanisms. Variation points are bound to specific variant end products.

3.2 Phase 2: Prototype Systems Development

A prototype software system has been developed to demonstrate the feasibility of establishing traceability to support variability management using the traceability model. Functionalities in the system that addresses these requirements are described in Table 1.

Figure 3 shows the architecture of the prototype system. The arrows in the figure represent information and request flows.

3.3 Phase 3: Experimental Validation

Past research has investigated the effect of documentation on design and maintenance quality. Maintenance performance has been used in several studies as a dependent variable (Banker et al. 1998). I argue that the use of traceability knowledge as specified by the model in Figure 2, will significantly reduce the maintenance effort and enhance the maintenance quality. Also, I argue that maintenance task complexity will partially moderate the effect of traceability knowledge on maintainability of the design.

Figure 2: Traceability Model for Variability Management

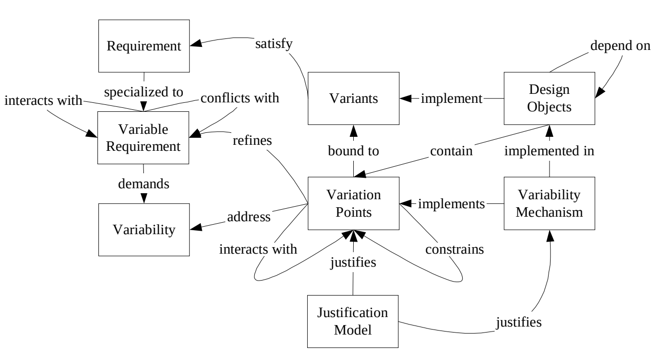
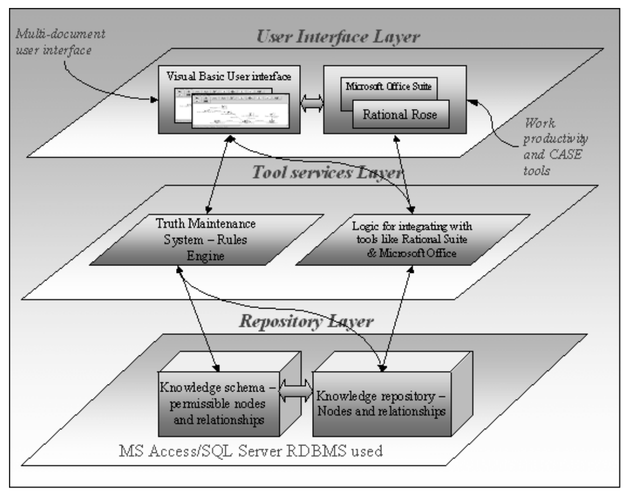


Table 1: Requirements and Capabilities of the Prototype System

Requirement	Capability
Structuring traceability knowledge	Diagrammatic interface to represent traceability knowledge and a knowledge schema that controls the trace data types and link types
Linking to design artifacts	Integration with work productivity and development tools
Identifying artifacts that are impacted by changes	Truth maintenance system: Propagation of status changes to related elements
Linking to knowledge elements of other projects	Facility to link elements from multiple models and sub-models
Tool-independent use of knowledge	HTML publisher that generates a HTML page with embedded pictures of the traceability knowledge captured with descriptions

Figure 3: Prototype System Architecture



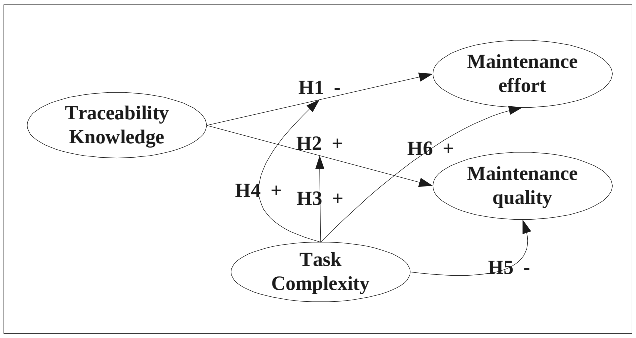
A 2x2 factorial design is appropriate as there is one independent variable and one moderator variable with 2 levels (Figure 5). Each subject will be provided a product family design model. Half of the subjects will receive designs that are supported by traceability knowledge documented using our traceability model and the other half will receive designs with standard documentation. UML has been used to create the product family design model. The subjects will be asked to perform maintenance tasks on each of the design model. The independent and moderator variables are operationalized by the treatment. Expert ratings on objective complexity of the tasks would be used as a measure for task complexity. The subjects will be asked to keep track of the time taken to solve each task. Expert ratings of the quality of the subjects' performance will be used as a measure for maintenance quality. We will test for differences due to demographic variables like experience to systems design using Unified Modeling Language and experience in performing maintenance tasks. Multiple Analysis of Variance (MANOVA) will be used to analyze the data.

4. EXPECTED CONTRIBUTIONS

The traceability model developed in this dissertation will provide a reference framework that can be tailored to project-specific needs. The capture and use of traceability knowledge is expected to enhance maintainability of the system. The key contributions of this dissertation are as follows:

- Traceability Model - Identification of useful knowledge elements related to variability.
- The prototype software system that provides tool support for traceability.

Figure 4: Conceptual Model to be tested



This research is expected to contribute to both theory and practice. Research can focus on tailoring the model produced to suit to different types of product families by identifying specific attributes of these families. Practitioners can use the models developed here as a reference to identify specific elements of knowledge that are to be captured to facilitate effective variability management.

5. LIMITATIONS AND FUTURE RESEARCH

The system that has been developed and used to demonstrate the usefulness of the traceability knowledge is just a research prototype. An industry standard implementation with user-friendly interfaces is beyond the scope of this work. Use of students as subjects for the experimental validation might pose a threat to external validity.

Figure 5: Factorial Design with Repeated Measures

	Complexity	
	Simple	Complex
With standard documentation	Group G11	Group G12
With traceability knowledge structured using our model	Group G21	Group G22

In this dissertation, the focus is on managing variability than on managing commonality. Future research will identify various issued related to commonality in product families and focus on traceability for product platforms. Development of appropriate sub-models that are related to specific elements in the traceability model developed in this dissertation, will be a potential research direction. Identification and retrieval of patterns of process knowledge capture is another area of research we are currently investigating.

6. FEEDBACK SOUGHT

In this dissertation, three different methodological paradigms have been used, viz., grounded theory approach (to develop the traceability model), design research methodology (to demonstrate the feasibility of the traceability approach by building a prototype system), and experimental approach (to evaluate the usefulness of the traceability knowledge in managing variability). I would like to have feedback on the appropriateness of such a combination of approaches to developing and validating a solution. As part of this dissertation, an experimental approach is used to evaluate the usefulness of the traceability knowledge in managing variability, but the various capabilities of the prototype system are yet to be evaluated. Several knowledge management based approaches have been proposed in the past to solve different problems in managing knowledge intensive processes. I would like to have suggestions on various validation techniques and the theoretical background that can be used to evaluate the effectiveness of knowledge management based approaches and the capabilities of the associated prototype systems.

7. REFERENCES

Banker, R., G. Davis and S. Slaughter (1998). “Software development practices, software complexity and software maintenance performance: A field study.” *Management Science* 44(4): 433-450.

Bronsword, L. and P. Clements (1996). A Case Study in Successful Product Line Development. *Pittsburgh, PA*, Software Engineering Institute, Carnegie Mellon University.

Conklin, E. J. and M. Begeman (1988). GIBIS: A hypertext tool for exploratory policy discussion. *Proceedings of CSCW’88.*

Gotel, O. C. Z. and A. C. W. Finkelstein (1994). An Analysis of Requirements Traceability Problem. *IEEE International Conference on Requirements Engineering*, Colorado Springs, IEEE Computer Society Press.

Strauss, A. and J. Corbin (1990). Basics of Qualitative Research: Grounded Theory Procedures and Techniques. Newbury Park, CA, Sage Publications.

APPENDIX

Detailed Dissertation Plan