

Variation View를 이용한 Product-Line의 가변성 관리기법

황길승⁰ 송문섭 양영종

한국전자통신연구원 기반기술연구소 임베디드S/W기술센터 S/W공학연구원
{kshwang⁰, sirius, yangyj}@etri.re.kr

The Method of Variability Management in the Product Line Engineering using Variation View

Kil-Seung Hwang⁰ Moon-Sub Song Young-Jong Yang

Software Engineering Research Team, Embedded S/W Technology Center, Basic Research
Laboratory, ETRI(Electronic and Telecommunication Research Institute)

요 약

Product-Line 개발방법에서 가변성의 표현과 선택은 최종 Product의 형태를 결정하는 중요한 요소이므로 개발 전 단계에서의 일관성 있는 관리가 필요하다. 현재 Product-Line에서 가변성의 표현을 위한 다양한 방법들이 연구되고 있지만 실제적으로 개발 단계간의 가변성의 연관된 변화과정을 서술하고 관리할 수 있는 방법은 부족한 실정이다.

본 논문에서는 소프트웨어 개발단계의 각 계층 간 가변성을 표현하는 방법을 정의하고 이 방법으로 Product-Line의 가변성이 효율적으로 관리될 수 있음을 증명한다. 본 논문에서 제안하는 Variation View를 이용하면 S/W 전체의 가변성의 변화를 한눈에 파악할 수 있고, 가변성 결정에 따른 소프트웨어 변화를 예측할 수 있으며, 향후 evolution 과정에서도 공통 아키텍처를 유연하게 확장할 수 있는 방법을 제공할 수 있다.

1 서 론

80년대까지의 구조적인 S/W 개발방법은 S/W의 다양화, 대형화에 따른 요구의 증가로 90년대 초 객체지향 방법, 90년대 말 컴포넌트 기반 방법 등으로 진화해 왔고, 현재는 S/W 시스템 개발에 필요한 서비스 컴포넌트의 다양한 보급이 가능해지고, 컴포넌트 기반 기술이 성장함에 따라 시장성, 기술적 진보성이 보완된 Product-Line 기반 개발방법, Model-Driven 개발방법, Service 기반 개발방법 등의 한단계 진화한 S/W 개발기술이 도입되고 발전하기 시작하고 있다.

S/W Product-Line 은 공통의 유사한 기능을 가지고 있는 S/W 제품 또는 S/W 시스템의 집합을 의미하며 [1], S/W Product-Line 기반 개발이란 특정 영역의 시장과 용도의 요구사항을 만족시키기 위해 S/W 제품을 미리 구축된 S/W 아키텍처 등의 S/W 핵심자산을 재사용하여 개발하는 방법이다. 이 방법은 미리 구축된 S/W 핵심자산을 재사용하므로, 처음부터 전체 시스템을 개발하는 방식보다 쉽고, 빠르게 S/W를 생산할 수 있는 장점이 있다.

S/W Product-Line의 핵심은 S/W 제품군에 공통적인 공통아키텍처를 기반으로 하여 각 제품별로 가지는 특징들에 대한 가변성을 제어하는 기술이다. 그러므로, 성공적인 S/W Product-Line의 개발을 위해서는 S/W개발의 모든 추상화 단계에 적용할 수 있는 가변성 관리 기술의 개발이 필수적이다.

현재 S/W Product-Line을 기반으로 한 다양한 개발방법들이 있으며, 각 방법론들은 각각 고유한 방법으로 가변성 관리를 지원하고 있다. Kobra[2]나 FAST[3] 등의 대표적인 Product-Line 개발방법론에서는 가변성 관리 방법으로 Decision-Resolution Model을 사용한다. Kobra 에서는 Context Realization단계에서 Komponent Specification, Komponent Realization으로의 단계이동에서 가변성의 관리 방법으로 Decision-Resolution Model을 사용한다. Decision-Resolution Model은 각 단계별로 Variation Point를 기술하고 하위의 Variant의 선택결과에 따른 각 산출물의 Effect를 표의 형태로 기술한다. 이 방법은 각 단계의 중요한 가변성 변화를 간단하게 표현하여 최종적인 Resolution에 따른 S/W 아키텍처의 형태의 생성을 지원한다. 하지만 Decision-Resolution Model은 각 추상화 단계 사이의 연결방법이 없으므로 가변성의 변화에 의해 시스템이 어떻게 바뀌어 질지 쉽게 파악하기 힘들다는 단점이 있다. FORM[4]에서는 Feature들 간에 존재하는 세가지 형태의 가변적인 요소들을 다이어그램의 형태로 표현한다. 하지만 이 방법은 요구사항 분석단계에서 발생하는 가변성만을 대상으로 하고 있기 때문에 하위 단계에서의 가변성을 제어할 수 있는 방법을 제공하고 있지 못하다는 단점을 가지고 있다.

본 논문에서는 S/W 공통 아키텍처의 각 추상화 단계 사이의 가변성의 변화를 설계단계에서 파악하고 관리하기 위한 방법을 제안한다. 제안된 Variation View를 통하여 S/W 전체의 가변성 변화를 한눈에 파악할 수 있으며, 가변성의 결정에 따른 S/W의 변화를 예측할 수 있다. 이 방법을 통하여 S/W개발 프로세스의 전 단계에서의 간단한 가변성관리가 가능할 것으로 기대되며, 향후 S/W Product-Line의 핵심적인 기반기술로서의 역할을 수행할 것으로 생각된다.

본 논문의 구성은 다음과 같다. 2장에서는 Variation View의 개념에 대해 서술하고, 3장에서는 Variation View의 표현방법에 대해서 설명하고, 4장에서 결론을 맺는다.

2 Concept of Variation View

Product-Family 사이의 공통성과 가변성을 분리하는 이유는 유사한 구조를 가지는 Product Domain에서 공통적인 제품 아키텍처를 재사용하고 가변적인 부분을 효율적으로 관리함으로써 S/W개발 생산성과 효율성을 향상시키기 위해서이다. 그러므로 Product-Line 방법에서 Product Family들 사이의 공통적인 특징들과 가변적인 특징들 사이를 관리하는 것은 Product-Line을 구축하는데 필수적이며 가장 중요한 부분이라고 할 수 있다[5]. 일반적으로 이 활동을 가변성 관리라고 부른다.

Family 기반 개발 프로세스는 domain engineering 프로세스와 application engineering 프로세스 두 부분으로 나뉘어 있다. 가변성과 공통성의 검증은 domain engineering 프로세스의 domain analysis 과정에서 발생하여 프로세스에 따른 S/W의 각 추상화 단계별로 세분화된 가변성 요소들을 가지게 된다.

가변점(Variation Point)은 S/W 개발의 각 추상화 단계들 사이의 형태를 결정하는 연결고리의 역할을 수행한다. 가변점이 포함하고 있는 가변요소들(variants)과 resolution에 의한 효과(effect)는 추상화레벨이 낮아짐에 따라 상세화되고 세분화된다. 그리고 최종적인 application engineering 단계에서 가변점이 resolution됨에 따라 Target Product의 모양이 결정된다.

이러한 S/W 개발 전 단계에서의 가변성 변화 및 결정 과정을 한눈에 파악할 수 있는 방법의 필요성은 높다[6].

본 논문에서는 위와 같이 S/W의 아키텍처의 추상화 단계 변화에 따른 가변성의 변화를 아키텍처 레벨에서 파악하기 위해 아키텍처 Variation View를 정의하였다. 각각의 아키텍처 View들과 Variation View와의 관계는 그림 1과 같다.

Variation View는 전체 S/W의 variability를 명시적인 그래픽 방법으로 표현하는데 그 목적이 있다. 개발자는 S/W의 개발단계에서 발생하는 모든 가변점과 선택에 의한 효과들을 Variation View를 이용하여 표현할 수 있다

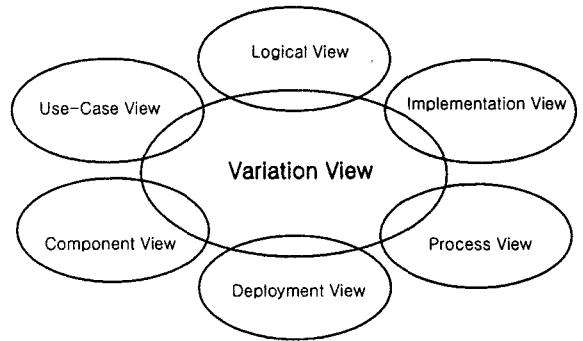


그림 1 Variation View와 각 View들간의 관계

Product-Line에서 S/W의 공통 아키텍처에서 발생하는 가변성들을 하나의 Architecture View로 표현하는 것은

1. Resolution 실패에 따른 오류 trace를 가능하게 하고
2. Resolution에 따른 Target Product의 모양을 쉽게 예측할 수 있게 하며
3. 핵심자산의 evolution시 유연하게 가변성을 확장할 수 있게 하고
4. 공통 아키텍처의 가변성을 명확하게 표현함으로써 재사용을 용이하게 한다.

따라서 Product-Line 방법에서의 공통 아키텍처의 표현에 Variation View는 필수적인 요소라고 할 수 있다.

3 Representation of Variation View

가변점과 선택에 따른 효과를 명시적으로 표현하기 위해서는 다이어그램 형태로 View를 나타내는 방법이 필요하다. Variation View는 특정 형식의 다이어그램으로 표현되며, Variation View의 메타모델과 그 구성요소에 대한 설명은 다음과 같다

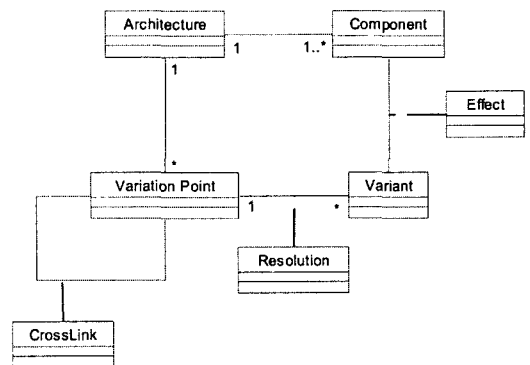


그림 2. Variation View의 메타모델 보여줄 수 있다. 가변점의 변화에 따른 S/W의 변화는

1. Conceptual Component
유사 기능들 간의 개념적인 집합. 최종적으로 Physical Component가 될 가능성이 높다.
2. Variation Point
가변점. 가변성이 발생하는 지점으로써 각 추상화 레벨 사이의 연결고리 역할을 한다.
3. Variants
Resolution을 통해 선택될 수 있는 경우의 집합. 선택된 Variant는 구체화된다.
4. Resolution
Variant를 선택하는 일련의 행위.
5. Effects
Resolution에 의해 발생하는 S/W의 변화.
6. Cross Link
각 추상화 레벨의 Variation간의 연결방법

위의 요소들은 아키텍처 전체의 가변성을 표현하기 위해 필요한 요소들으로써 Variation View 다이어그램의 요소들로 각각 표현된다. Variation View의 개략적 모습은 다음과 같다.

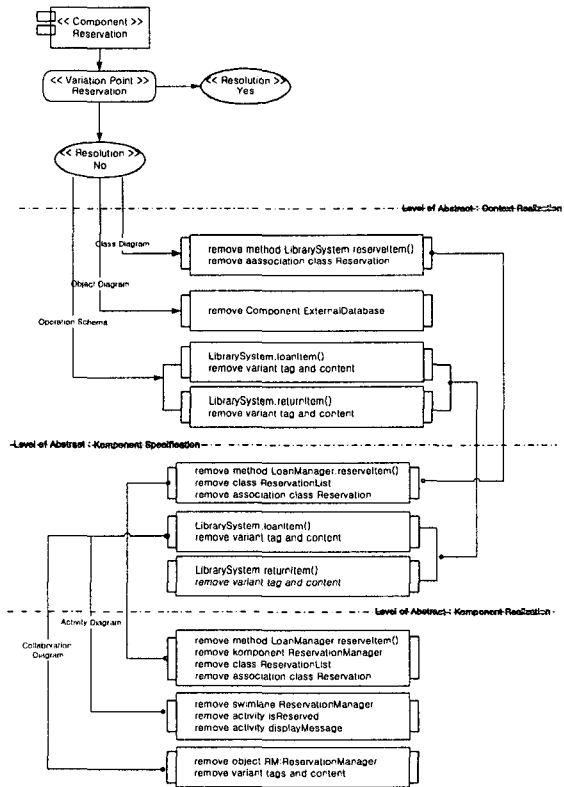


그림 3. Variation View

위의 그림과 같이 Variation view는 각각의 추상화 단계 사이의 변화요소들 간의 연결고리를 표현해 줌으로써 S/W 전체 개발단계에서의 가변성의 변화를 명시적으로

Variation View를 전체 Architecture의 구조에 맞추어 Resolution 함으로써 예측해 볼 수 있고, 변화의 흐름에 대한 추상화 단계간의 연결고리가 명시되어 있으므로 오류나 예외에 대한 추적이 용이하다. 개발단계에서 가변점은 다양한 추상화 단계에서 나타날 수 있으며 [7], 가변점에 따라 각 추상화 단계에서의 Effects가 다르게 나타날 수 있다. 이러한 Effect들 간의 상관관계를 정의해 줌으로써 가변성 관리를 용이하게 할 수 있다.

4 Conclusion

Product-Line 기반 S/W 개발방법은 아직은 기반이론에 포함된 다양한 요소기술들이 표준화되어 개발되지 않았기 때문에 많은 부분에서 이론적인 약점이 존재한다. 가변성 관리 분야도 그 일부로써 현재 몇몇 연구가 진행되고는 있지만 그 중요성에 비추어 볼 때 연구의 수준은 미미한 실정이다. 본 논문에서는 Product-Line 기반으로 S/W를 개발할 때 필수적으로 고려해야 하는 부분인 가변성관리 부분을 기존의 방법과 비교하여 효율성과 가시성을 높일 수 있는 방안을 제안하였다. 제안된 방법을 이용하면 S/W 아키텍처에 존재하는 가변성들을 체계적이고 효과적으로 관리할 수 있을 것으로 생각된다. 향후 본 연구는 Variation View의 표현방법을 체계적으로 정의해 나갈 것으로써 S/W 개발에서 나타날 수 있는 모든 가변요소들에 대한 명시적인 관리방법을 개발해 나갈 예정이다.

5 참고문헌

- [1] Northrop L., Framework for Software Product Line Practice, SEI Report, 2002. (<http://www.sei.cmu.edu/plp/framework.html>)
- [2] Colin Atkinson et al., *Component-based Product Line Engineering with UML*, Addison-Wesley, 2002.
- [3] David M. Weiss, Chi Tau Robert Lai, *Software Product-Line Engineering : A Family-Based Software Development Process*, Addison-Wesley, 1999.
- [4] Kang K., Kim S., Lee J., Kim K., Shin E., *FORM: A Feature-Oriented Reuse Method with Domain-Specific Reference Architectures*, Annals of Software Engineering, Vol.5, 1998.
- [5] Krueger, Charles W.(2002). "Variation Management for Software Product Lines", The Second Product Line Conference(SPLC 2), San Diego, CA, USA.
- [6] F. Bachmann, M. Goedicke, J. Leite, R. Nord, K. Pohl, B. Ramesh and A. Vilbig, "A Meta-model for Representing Variability in Product Family Development," in *Proceedings of the Intl. Conf on Product Family Engineering (PFE-5)*, Siena, Italy, Nov. 2003.
- [7] Svahnberg M., Van Gorp J., Bosch J., Research report: On the Notion of Variability in Software Product Lines. Blekinge Institute of Technology Research Report 2000:2, ISSN:1103-1581