

# 회차 모델 기반의 프로덕트 라인 아키텍처 변화성 설계

김수연<sup>o</sup>, 김지영, 정란<sup>\*</sup>, 김행곤  
대구가톨릭대학교 컴퓨터공학과, 삼척대학교 컴퓨터공학과\*  
e-mail:{gjtns9384, kimjy, hangkon}@cu.ac.kr,  
jungran@samcheok.ac.kr\*

## Design of Product Line Architecture Variability based on Feature Model

Su-Youn Kim<sup>o</sup>, Ji-Young Kim, Ran-Jung<sup>\*</sup>, Haeng-Kon Kim  
Dept. of Computer Engineering, Catholic University of Daegu  
Dept. of Computer Engineering, Samcheok University\*

### 요 약

프로덕트 라인은 다양하고 빠르게 변화하는 시장의 요구사항과 특정 도메인 영역에 속하는 애플리케이션 간의 재사용 가능한 아키텍처 및 컴포넌트의 구성으로부터 연관된 시스템 구축시 생산성과 품질의 향상을 제공함으로써 현재 많은 관심의 초점이 되고 있다. 또한, 이들 프로덕트 라인에서는 프로덕트들 사이의 공통성과 변화성에 초점을 두고 이들 분류 방법으로 회차 모델링이라는 개념을 주로 사용하여 분석하고 있다. 또한 재사용 가능한 아키텍처는 많은 변화 계획들과 메카니즘을 포함하고 있다. 하지만, 아키텍처를 설계하기 위한 아키텍처에서의 변화성 관리에 대한 명확한 방법이 미흡하다. 따라서 본 논문에서는 재사용 가능한 아키텍처를 설계하기 위해 변화성의 명확한 표현과 아키텍처에서의 적절한 위치를 식별하기 위해, 다양한 변화성 타입을 정의하고, 회차 모델을 기반으로한 아키텍처의 변화성과 아키텍처의 컴포넌트 관련성에서의 변화성 표현 방법을 기술하고, 사례 연구로 웹기반 교육 시스템 개발에 적용하고자 한다.

### 1. 서론

프로덕트 라인은 연관된 시스템 그룹의 아키텍처를 기반으로 공통성과 변화성 부분으로 구성되고, 변화성 부분은 수정되거나 변경될 수 있다. 또한 프로덕트 개발의 생산성, 비용 및 품질 향상을 위해 재사용 가능한 컴포넌트들을 공유함으로써 최상의 애플리케이션 개발을 위한 파라다임으로 인식되고 있다. 프로덕트 라인 개발 방법에서는 영역내의 여러 애플리케이션들 간의 차이점과 공통점을 분류하는데 회차 모델링이라는 분석 방법을 주로 사용하고 있다. 회차 모델링은 도메인 공학의 한 방법으로서 도메인 언어를 중심으로 도메인 내의 공통점과 차이점을 찾아내어 구조화하여 분석하고, 분석된 결과를 바탕으로 도메인에 적합한 설계 모델을 만들고, 도메인 내에서 가치 있는 소프트웨어 컴포넌트를 추출해낸다 [1]. 회차 모델링을 기반으로 하여 추출해낸 재사용성 있는 소프트웨어는 시스템 개발 초기부터 재사용성을 고려하여 생산되었기 때문에, 분석 대상 영역에서 변화 가능한 많은 응용 시스템에 대하여 적용성이 높고, 분석 단계 모델을 통해 재사용 될 수 있다는 점에서 매우 가치가 높다고 할 수 있다. 또한 소프트웨어 아키텍처는 변화에 대한 일반적인 계획과 이러한 변화들을 지원하기 위한 아키텍처의 메카니즘을 제공한다. 그러나 지금까지 이러한 변화들이 일어나는 상황을 이해하는 것과 특별한 상황에

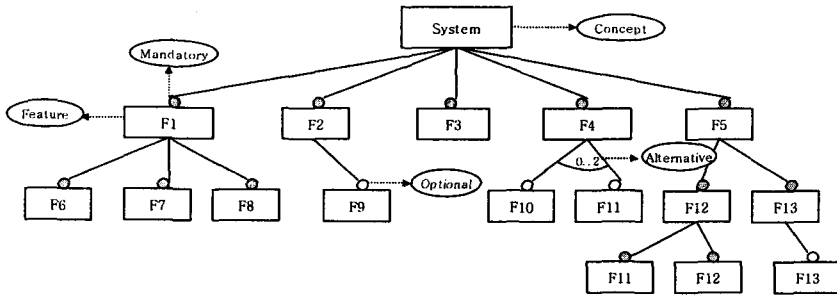
서도 가능하게 하는 옵션들을 기록하는 것은 명확히 이루어지지 못하였다[2]. 또한, 아키텍처가 오랜 기간 동안 많은 프로덕트 버전에서 사용되어 진다거나, 다른 프로덕트들의 설계를 위해 사용되어 지는 아키텍처에서의 프로덕트라인 문맥에서라면, 매우 중요하게 다루어진다. 즉, 명백한 변화성의 표현과, 아키텍처에서 변경이 되는 적절한 위치를 식별하는 것은 중요하다.

따라서 본 논문에서는 회차 모델 기반의 아키텍처 기술에서의 다양한 타입의 변화성을 기술하고, 변화성의 타입을 표현하는 방법과, 변경이 지원되는 아키텍처에서 위치를 쉽게 찾기 위해서 필요한 아키텍처에 추가될 특성들에 대해 제시하고 사례연구를 통해 적용하고자 한다.

### 2. 관련 연구

#### 2.1 회차 모델

회차 모델은 프로덕트 라인 요구사항 분석의 핵심 결과로 프로덕트 라인 구성의 기능적, 비기능적인 요소 뿐 아니라 공통성과 변화성을 획득하고 다양한 관점을 제공할 수 있다. 사용자는 회차 모델을 통해 프로덕트 라인의 기능을 이해할 수 있고, 개발자들은 다양한 프로덕트 변화의 개발을 유도할 수 있다. (그림 1)은 회차모델의 간략한 예를 보여주고 있다. 분석 모델로서의 회차 모델링은 하나의 모델로 회차들의 의미를 표준화하고, 통합함으로써,



(그림 1) 다이어그램으로 표현된 휘처 모델

표준 용어들을 정의하고 문제들에 대한 손쉬운 의사소통을 가능하게 하고, 응용 시스템들 사이의 공통점과 차이점을 평가하기 위한 잣대로 사용될 수 있으며, 한 영역을 특성화하고, 다른 영역들과 비교하기 위하여 사용될 수 있다. 앞서 기술한 바와 같이 휘처 모델은 영역 내의 용어를 중심으로 하여 영역을 분석하기 때문에, 영역 내의 특성을 최대한 살려서 표현할 수 있게 된다. 이러한 영역 용어를 기반으로 추출된 휘처들은 다른 영역과 차별되어 드러나는 것으로서, 한 영역을 특성화하고 비교하는데 도움을 준다[3,4,5].

**2.2 프로덕트 라인의 변화성**

동일한 프로덕트 군에 속하는 프로덕트들은 많은 공통성을 가지지만, 또한 프로덕트 사이의 변화성도 있다. 변화성은 다양한 사용자, 다양한 설계와 구현요구사항에 따라 제공되고, 도메인 분석 시에 구현요구사항에 따라 제공되고, 도메인 분석 시에 식별된다. 따라서 변화성은 프로덕트 라인 아키텍처 설계 단계에서 고려되어야 하고, 코드 레벨이 아닌 아키텍처레벨에서 다루어져야 한다. 또한 변화성은 컴포넌트 조립 시에 컴포넌트 행위가 변경될 수 있는 특정 변화점에서 가능하고, 컴포넌트 설계 동안에 결정된다. 변화성은 다음과 같은 범주로 분류된다. 휘처 변화성은 특정 휘처의 정의와 구현에서 변화성, 하드웨어 플랫폼 변화성은 제어기, 메모리, 장비 등의 타입에서 변화성, 성능과 속성 변화성은 요구된 성능과 동시에 지원 및 오류 관리와 같은 속성에서의 변화성을 의미한다. 이런 변화성의 종류는 <표1>과 같이 나타낼 수 있다[1,2,6].

<표 1> 변화성의 종류와 지원 기술

변화성의 종류	지원방법에 대한 기술
합 수	특정 기능이 어떤 프로덕트에는 존재하지만, 다른 프로덕트에는 존재하지 않음
데이터	특정 데이터 구조가 프로덕트 상에서 다른 프로덕트에게 다양하게 존재 · 비구조화 (text string), 구조화(structured format)
제어흐름	특정 상호작용 패턴이 프로덕트 상에서 다른 프로덕트에게 다양하게 존재
기술	플랫폼은 function과 같은 동일한 방식으로 다양하게 존재 · 미들웨어, OS, 하드웨어의 특정 종류
품질목표	특정 품질 목표도 프로덕트에 따라 다양하게 존재
개발환경	환경에 따라 프로덕트 상호작용의 형태가 다양하게 존재

**3. 아키텍처에서의 변화성 설계 방법**

프로덕트 라인 아키텍처는 변화성의 설계 요소를 포함해야 한다. 구조적인 변화성은 아키텍처 설계 동안 선택적인 설계 옵션으로 종종 아키텍처 변화점의 집합으로 표현한다. 하지만, 휘처 모델은 특정 설계를 함축하지 않고, 아키텍처를 구조화하기 위한 가이드 라인을 제공한다. 또한 최종 아키텍처는 성능과 품질과 설계 제한 조건을 만족하는 기능적인, 비기능적인 요구사항을 고려해야 한다.

**3.1 휘처 모델링**

관련연구의 (그림 1)은 간단한 휘처 모델링의 예를 나타낸 것이다. 이 휘처 모델은 다양한 변화성을 보여주고, 요구사항 사이의 변화성을 만족하기 위해 아키텍처에서의 변화점을 제시하고 어떻게 변화점을 표현할 것 인지의 가이드라인을 제공한다.

**3.2 변화성 타입**

아키텍처에서 다양한 요구사항을 표현하는 것은 프로덕트 라인 아키텍처에서 프로덕트들 사이의 변화성을 다루기 위해 다양한 선택 사항을 포함하는 경우이다. 이러한 변화성의 타입은 크게 3가지 형태로 정의되며, 변화성의 표기는 UML의 다중성 표기를 기반으로 5가지 형태로 식별될 수 있다<표 2>. 문맥에서의 보다 쉬운 이해를 위해 <표 3>과 같은 Variant 타입의 기본 표기를 정의한다.

**3.3 프로덕트 라인 아키텍처에서의 변화성 표현**

(그림 2-a)는 두 프로덕트의 컴포넌트 뷰를 보여준다. 이들 컴포넌트 뷰는 앞의 (그림 1)의 휘처 모델 예에서 어떤 휘처를 선택하느냐에 따라 서로 다른 아키텍처의 표현을 나타낸다. 그리고 하나의 휘처는 하나의 컴포넌트에 대응된다고 가정한다. 이 그림에서 두 프로덕트의 차이점은 하나의 컴포넌트의 위치이다(컴포넌트 B, 컴포넌트 D).

<표 2> 변화성의 타입

변화성 타입	설명
Optional	어떤 기능인 한 프로덕트에 포함되고 다른 프로덕트에는 불포함 · optional의 관점은 선택적인 기능성을 갖는 다른 기능들의 관계와 프로덕트에서 포함되지 않은 선택적인 기능인 경우 이러한 관계들이 발생할 수 있음을 나타냄
Alternative	추가될 수 있는 여러 선택 중의 하나가 추가
Set of Alternative	추가될 수 있는 여러 선택 중에 여러 개가 추가되는 것

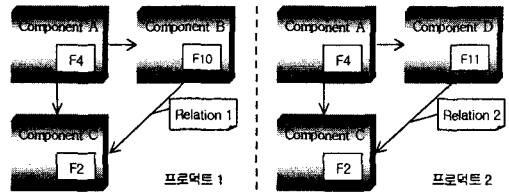
<표 3> 변화성 타입 표기 예

변화성 표기	변화성 타입	의 미
Variant A 0	Optional	프로덕트에 포함되는 정확히 하나의 구현에 존재하는 것
Variant B 1	Alternative	Variant의 다중 실현이 존재하고, 정확히 하나가 프로덕트에 포함되어야 함
Variant C 1..*	Set of Alternative	Variant의 다중 실현이 존재하고, 적어도 하나가 프로덕트에 포함되어야 함
Variant D 0..1	Optional Alternative	Variant의 다중 실현이 존재하고 그 중의 하나가 프로덕트에 포함될 수 있음
Variant E 0..*	Optional set of Alternative	Variant의 다중 실현이 존재하고, 그것의 모음이 프로덕트에 포함될 수 있음

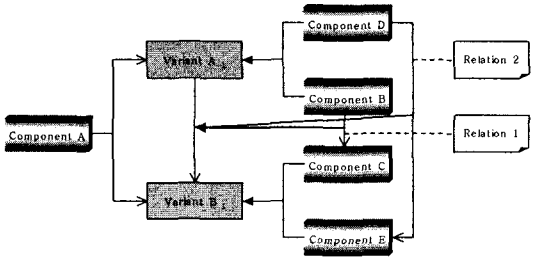
이들 변화성을 지원하기 위해 아키텍처는 하나의 다이어그램으로 통합될 수 있다. (그림 2-b)는 하나의 표현으로 두 개의 아키텍처를 표현한다. Variation A는 변화점(VP)을 의미하고, 컴포넌트와 구별하기 위해 1차원의 그레이 박스를 사용한다. 또한 이들 Variation A의 가능한 구현 기능을 추적하기 위해 (그림 2-c)과 같이 나타낼 수 있다.

3.4 컴포넌트 관련성에 대한 변화성 표현

변화성은 아키텍처의 컴포넌트 사이에서 관련성을 위해 발생할 수 있다. (그림 3)은 두 프로덕트에 대한 컴포넌트 뷰로 컴포넌트 사이의 관련성에 대한 변화성을 표현한다. 프로덕트 1은 컴포넌트 B와 C 사이에서 "Relation 1"을 발생시키고, 프로덕트 2에서의 모듈 D는 "Relation 2"의 관련성을 발생시킨다. 또한, 이들 관련성에 대한 변화는 연결된 컴포넌트의 변화성을 포함한다. 그러므로 컴포넌트 C는 Variant B로 표현 가능하다. (그림 4)는 Variant A와 Variant B 사이의 의존성을 하나의 큰 그림으로 통합한 것이다.



(그림 3) Relation의 변화성



(그림 4) 변화점

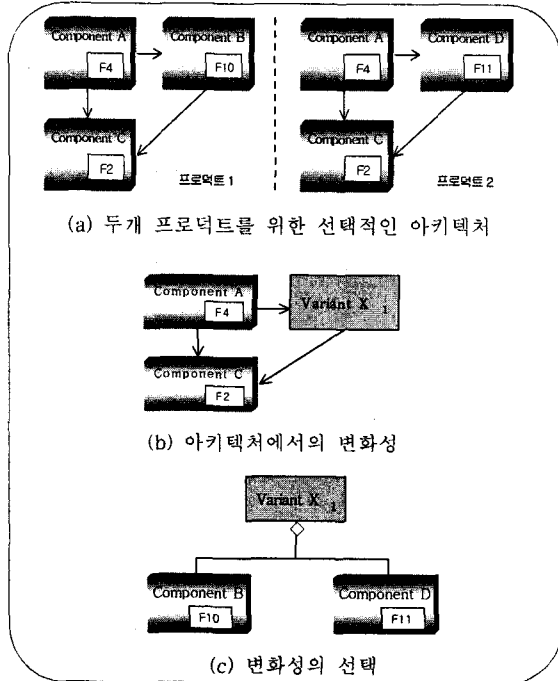
4. 사례연구

4.1 웹기반 교육 시스템에서의 위치 모델

위치 모델은 프로덕트 라인 요구사항 분석의 핵심 결과로 프로덕트 라인 구성의 기능적, 비기능적 요소뿐 아니라 공통성과 변화성을 획득하고 다양한 관점을 제공할 수 있다. 사용자는 위치 모델을 통해 프로덕트 라인의 기능을 이해할 수 있고, 개발자들은 다양한 프로덕트 변화의 개발을 유도할 수 있다. (그림 5)는 웹기반 교육 시스템에 대한 위치 모델의 간략한 예를 보여준다. 제시된 웹기반 교육 시스템은 사용자 관리에서 회원인증과 회원 가입, 회원정보 수정 등의 사용자의 관리를 하게 되고 사용자는 학생의 주체인 학생과 교사로 나눌 수 있으며 교사의 경우에는 평가 부분과 연관성을 가지고 학생은 콘텐츠 부분과 연관성을 가진다. 평가는 다시 하위 영역으로 결과관리, 평가기준 관리, 평가문항작성을 가지게 된다. 콘텐츠는 본시 학습콘텐츠와 평가 후 피드백으로 제시되는 재학습 콘텐츠, 평가문항 콘텐츠로 구성되며 재학습은 평가결과에 따라 상·중·하로 각기 재학습 콘텐츠를 구성하여 제공하게 된다. 위치 모델은 웹기반 교육 시스템 프로덕트 라인의 기능을 트리 형태로 구조화하고, 응용은 시스템의 기능들을 포함하고 있다.

4.2 웹기반 교육 시스템에서의 변화성 모델링

변화성은 제시된 웹기반 교육 시스템을 통해 표현되어 질 수 있다. (그림 6)은 두 프로덕트 사이에 대한 컴포넌트 뷰로 컴포넌트 사이의 관련성에 대한 변화성을 표현한다. 프로덕트 1의 F3(교사)과 매핑되어 있는 컴포넌트 B는 F2(회원인증)과 매핑되어 있는 컴포넌트 C사이에서 "Display 평가"를 발생시키고, 프로덕트 2의 F4(학생)과 매핑되어 있는 컴포넌트 D는 F2(회원인증)과 매핑되어 있는 컴포넌트 C 사이에서 "Display 콘텐츠"를 발생시키게 된다. (그림6)에서 알 수 있듯이, 이 그림에서 두 프로덕트의 차이점은 하나의 컴포넌트와 위치이다(컴포넌트



(그림 2) 변화성 표현

