

# 프로덕트 라인 기반 모바일 콘텐츠 시스템의 분석 및 설계

황지영 김지홍  
경원대학교 전자계산학과  
e-mail : [sgr4620@naver.com](mailto:sgr4620@naver.com) [wiskjh@kyungwon.ac.kr](mailto:wiskjh@kyungwon.ac.kr)

## Analysis and Design of Product Line based Mobile Contents System

Ji Young Hwang Ji Hong Kim  
Dept. of Computer Science, Kyungwon Univ.

### 요 약

매년 빠른 성장률을 보이는 모바일 콘텐츠 서비스는 아직까지 확립된 표준이 없는 상황에서도 다양한 응용이 예상되고 있어서, 새로운 시스템을 쉽고 빠르게 도입하기 위해서는 체계적인 아키텍처의 확립이 요구되고 있다. 콘텐츠 시스템은 콘텐츠에 따른 전송 방식, 또는 제공 형태에 따라 차이를 가지면서도 그 요구 사항과 아키텍처는 유사하여 재사용이 가능하기 때문에 소프트웨어의 핵심적인 공통 자산을 재사용하는 기술인 프로덕트 라인 기술을 적용하는 것이 필요하다. 본 논문에서는 프로덕트 라인 기술 중 UML 기반의 PLUS 방법을 사용하여 끊임없이 변화하는 모바일 콘텐츠 시장에 적용하여 분석, 설계를 통해 변화에 발 빠르게 대응할 수 있도록 한다. 아울러 산출된 아키텍처를 적용하여 새로운 모바일 서비스의 요구를 만족하는 프로토타입을 얻을 수 있었다.

### 1. 서론

현재 무선 인터넷을 기반으로 하는 모바일 서비스 중 특히 콘텐츠 서비스는 이동통신시스템의 기술 진보에 따라 계속적인 발전을 거듭해 나가고 있는 분야로서 대표적인 미래 유망 산업으로 손꼽히고 있다. 모바일 콘텐츠란 무선 단말기에서 제공되는 콘텐츠 서비스로서 일반적으로 게임, 벨소리, 통화연결음 등을 대표적으로 꼽을 수 있다[1].

국내 모바일 콘텐츠 시스템은 높은 모바일 기술과 신규서비스의 조기 실현에 따라 세계표준을 리드하면서 고부가가치를 실현할 수 있는 분야로써[2] 현 시스템 뿐만 아니라 차후에 실현될 새로운 서비스를 보다 쉽고 빠르게 도입시키기 위해서는 체계적인 아키텍처 등의 개발 기술을 확립하는 것이 필요하다[3].

대부분의 모바일 콘텐츠 시스템은 각각의 콘텐츠 종류에 따른 정보의 제공 방식이나 전송 형태에서만 차이를 보일 뿐 기능적/비기능적 요구 사항과 아키텍처는 거의 유사하여 재사용이 가능하다. 프로덕트 라

인은 다양한 요구 사항을 쉽고, 빠르게 반영하여 최상의 제품을 높은 생산성으로 생산하는 기술로써 기존 소프트웨어 생산 기술들은 소프트웨어 개발 생산성 향상에 제한적이므로[4] 소프트웨어 핵심 자산들을 체계적으로 자원화하고 조직적으로 재사용하는 소프트웨어 생산 기술인 프로덕트 라인(Product Line) 기술이 필요하다.

프로덕트 라인의 여러 기술 가운데 PLUS 방법론은 기존의 방법론과 달리 객체지향 분석 및 설계의 실질적인 표준인 UML 을 사용함으로써 RUP 와 같은 객체지향 방법과 연계되어 소프트웨어의 공통점과 차이점을 손쉽게 표현하고 이해할 수 있으며 개발 관련자간의 의사 소통을 원활하게 하는 이점이 있다.

본 논문에서는 시스템의 개발 기간과 비용을 줄이고 소프트웨어의 품질을 보장하며 사용자들의 다양한 요구 사항을 쉽고 빠르게 반영하기 위해 프로덕트 라인 기술 중 GOMAA 가 주장한 PLUS 방법을 이용하여 모바일 콘텐츠 시스템 개발에 적용하였다.

## 2. 관련 연구

### 2.1 Product Line - PLUS

프로덕트 라인은 개발 단계 초기에 제품군에 속하는 멤버들의 공통점과 차이점들을 분석하여 단위가 크고 전략적인 형태의 재사용을 가능하도록 함으로써 궁극적으로 개발의 생산성을 향상시키고자 하는 것이 목적이다[5-11].

GOMAA 가 제안한 PLUS(Product Line UML-Based Software)은 소프트웨어 프로덕트 라인을 UML 2.0 을 기반으로 표현하는 소프트웨어 디자인 메소드를 말한다. PLUS 는 UML 기반의 Single System 모델링 기법을 확장 시킨 방법으로 객체들의 공통점과 가변점을 명확히 모델화 한다.

UML 표기법이 지원하는 과정에 따라 PLUS 역시 요구사항 활동, 분석 활동, 그리고 설계 활동 단계로 진행된다[12].

- 요구사항 활동 : 기능적 요구사항을 파악하는 단계로 공통점과 가변점을 찾아 규정한다.
  - 프로덕트 라인 범위 : 프로덕트 라인의 범위 등을 결정
  - 유즈케이스 모델링 : 기능적 요구사항에 따른 액터와 유즈케이스의 분류
  - 피쳐 모델링 : 공통점과 가변점을 kernel, optional, alternative 로 분류
- 분석 활동 : 요구사항에서 분석된 문제점을 명확하게 이해하기 위해 심도 있게 분석하는 단계
  - 정적 모델링 : 일정 영역의 정적 모델 정의
  - 객체 구조화 : 유즈케이스에 따른 객체 설정
  - 다이내믹 모델링 : 유즈케이스 간의 상호작용
  - 상태 모델링 : 프로덕트 라인에서의 상태 의존적인 부분을 정의
  - Feature/class dependency analysis : 피쳐를 구현하는데 필요한 클래스가 무엇인지 결정
- 설계 활동 : 아키텍처 패턴들을 파악하여 현 문제점에 맞는 해법들을 찾아 종합하여 컴포넌트 기반의 아키텍처를 설계하는 단계
  - 소프트웨어 아키텍처 패턴 기반의 설계 : Structural 과 Communication 패턴의 이해
  - 소프트웨어 아키텍처 설계 : 컴포넌트 기반의 소프트웨어 아키텍처 설계

### 2.2 모바일 콘텐츠

모바일 혹은 무선 콘텐츠란 무선 단말기에서 서비스되는 모든 콘텐츠를 말한다.[13] 모바일 콘텐츠는 일반적으로 게임, 음악, 정보, 영상, 캐릭터, 출판 등으로 나뉘어지며[14] 무선 인터넷 서비스 종류에 따라 정보형, 대화형, 엔터테인먼트, 전자상거래, 위치기반 등의 콘텐츠로 나눌 수 있다. 또한 콘텐츠의 전송 형태에 따라 온라인, 다운로드 콘텐츠로 분류된다.

이미 일생 생활 속에 자리잡은 콘텐츠 서비스는 벨소리, 통화연결음 등의 음악 시장으로 시작되어 현재는 게임 시장이 모바일 콘텐츠의 대표 주자로 자리

잡아가면서 엔터테인먼트 콘텐츠가 강세를 보이고 있으며 곧 시행될 DMB 등을 통해 더욱더 강세를 보일 것으로 예상된다.[15-17]

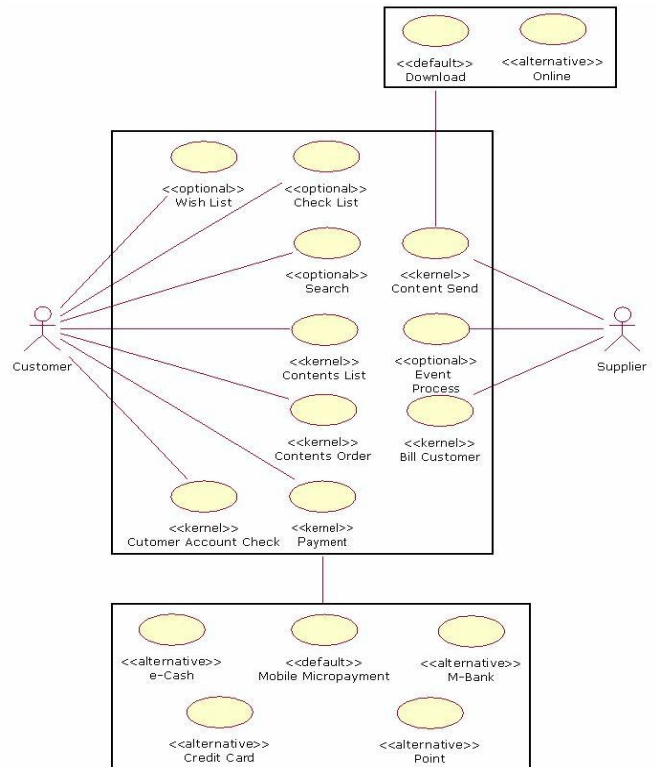
## 3. 모바일 콘텐츠 시스템의 분석 및 설계

모바일 콘텐츠 시스템에는 다양한 서비스가 존재하지만 서비스에 따라 반드시 포함되는 공통적인 기능과 서비스에 따라 구분되는 가변적인 기능들이 존재한다. 본 연구에서는 현재 가장 많이 이용되고 있는 엔터테인먼트 콘텐츠 서비스를 위한 시스템을 프로덕트 라인의 PLUS 방법론을[12] 적용하여 시스템의 특성을 UML 을 이용하여 표현하였다.

### 3.1 분석

분석에서는 현 시스템을 분석하여 유즈케이스 다이어그램을 토대로 Feature/Use case of dependencies Table 작성한다. 이를 바탕으로 Feature/Class of dependencies Table 을 작성한 후 아키텍처를 설계하는 과정을 알아보고 간단한 구현의 예를 통해 실제로 적용되는 모습을 살펴보도록 하겠다.

#### 3.1.1 유즈케이스 다이어그램



< 그림 1. Contents Service System use cases >

<그림 1>은 현 시스템을 통해 콘텐츠 시스템을 파악하여 유즈케이스 다이어그램으로 나타낸 것이다. 위의 유즈케이스를 사용하는 주체인 고객과 공급자의 입장으로 살펴보면 다음과 같다.

- 고객

- Content List: 콘텐츠 목록 확인
- Contents Order: 선택한 콘텐츠 주문
- Payment: 주문에 대한 값을 지불
- Customer Account Check: 고객 계정 확인
- Search: 고객 선택에 따라 콘텐츠 검색
- Check List: 고객 인증을 통해 현재 보유하고 있는 Contents 확인
- Wish List: 고객 인증을 통해 원하는 콘텐츠 목록 작성
- 공급자
  - Content Send: 요청한 콘텐츠 전달
  - Bill Customer: 대금 청구
  - Event Process: 특별한 행사를 진행

가변적인 유즈케이스의 경우 <<optional>> 또는 <<alternative>>로 분류할 수 있으며 이를 제외한 나머지 Use cases 는 모두 <<kernel>> 로 분류 할 수 있다. <<kernel>> 중에서 Payment 와 Send Content 는 각각의 수행 주체의 선택에 따라 달라지는 가변성을 포함하는 유즈케이스로 좀 더 세분화 할 수 있다. 둘 다 하나 이상은 선택되어야 하기 때문에 <<alternative>>로 분류 할 수 있으며 반드시 <<default>> 값이 필요하다.

3.1.2 Feature/use case dependencies

앞서 수행한 유즈케이스 다이어그램을 바탕으로 피처에 따른 유즈케이스의 dependency 를 모델로 표현 할 수 있으며 아래와 같이 표로도 표현할 수 있다.

Feature Name	Feature Category	Use Case Name	Use Case Category
System Kernel	common	Content List	kernel
		Content Order	kernel
		Customer Account Check	kernel
		Bill Customer	kernel
		Payment	kernel
		Content Send	kernel
Customer Service	optional	Search	optional
		Wish List	optional
		Check List	optional
Supplier Service	optional	Event Process	optional

< 표 1. Feature/use case dependencies of product line >

<표 1>은 피처를 중심으로 유즈케이스를 묶어서 표현한 것으로 optional 의 경우 각각의 주체에 따라 Customer Service 와 Supplier Service 로 묶어 표현하였다.

Feature Group Name	Feature Group Category	Feature in Feature Group	Feature Category
Payment	at-least-on-of	Mobile Micropayment	default
		M-Bank	alternative
		Credit Card	alternative
		e-Cash	alternative
		Point	alternative
Content Send	at-least-on-of	Download	default
		Online	alternative

< 표 2. Feature groups in product line >

<표 2 >는 가변성을 포함하고 있는 유즈케이스를 더 세분화 하여 표현한 것으로 같은 피처의 다른 유즈케이스를 모아 피처 그룹으로 표현한 것이다. 피처 그룹의 카테고리는 <<zero-or-one-of>>, <<exactly-one-of>>, <<at-least-one-of>>, <<zero-or-more-of>>로 나뉘지며 <<at-least-one-of>>는 그룹 내에서 적어도 하나는 선택해야 하는 것을 말한다. 반드시 임의의 선택 값이 존재해야 하기 때문에 피처 중 default 를 명시해주어야 한다. default 값으로는 현재 가장 많이 사용하고 있는 소액 결제 시스템(Mobile Micropayment)과 Download 방식으로 설정하였다.

3.2 설계

3.2.1 Feature/class dependencies

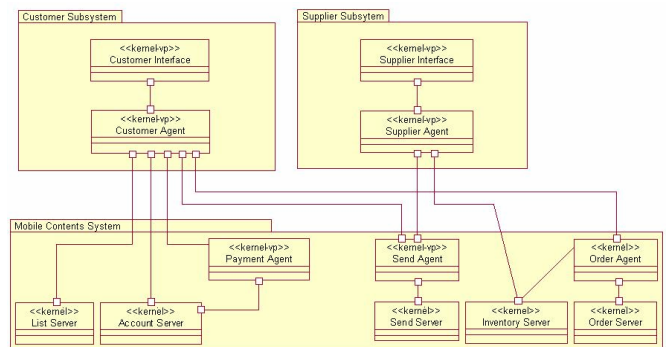
설계로 넘어가는 과정에서 Agent-based Conceptual View 를 작성하여 파악한 후 정적 모델을 통하여 <표 3>과 같이 피처에 따른 클래스의 dependency 를 살펴본다.

Feature Name	Feature Category	Class Name	Class Category
System Kernel	common	Customer Interface	kernel-vp
		Customer Agent	kernel-vp
		Supplier Interface	kernel-vp
		Supplier Agent	kernel-vp
		Content List Server	kernel
		Send Agent	kernel-vp
		Send Server	kernel-vp
		Inventory Server	kernel
		Order Agent	kernel
		Order Server	kernel
Customer Service	optional	Account Server	kernel
		Payment Agent	kernel-vp
		Search Agent	optional
Supplier Service	optional	Wish Agent	optional
		Check Agent	optional
		Event Agent	optional
		Event Server	optional

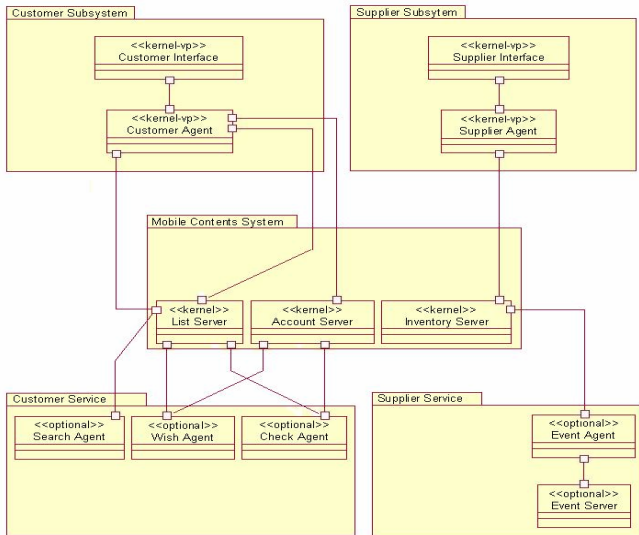
< 표 3. Feature/class dependencies of product line >

3.2.2 소프트웨어 아키텍처

본격적인 설계 단계에서는 위의 테이블을 바탕으로 레이어드 아키텍처를 작성하고 UML 2.0 을 이용하여 구체적인 소프트웨어 아키텍처를 설계하였다.



< 그림 2. Kernel Software Architecture >



< 그림 3. Kernel architecture and Optional >

### 3.3. 프로토타이핑

새로운 응용 인스턴스를 위해 3.2 절에서 설계한 소프트웨어 아키텍처를 사용하여 얻은 m-소싸움 서비스의 모델을 <그림 4>와 같이 WIPI 를 이용하여 [18] 프로토타이핑을 얻을 수 있었다.



< 그림 4. m-경마 서비스 프로토타입 >

## 4. 결론 및 향후 연구 과제

프로덕트 라인 기술은 이미 많은 분야에서 적용을 하여 그 효과를 입증하고 있다. 노키아의 모바일 브라우저 개발[19] 등 이미 다른 분야에서 그 효과를 입증하는 사례가 나오고 있으며 그 분야와 실적도 점차 확대되고 있다.

본 논문에서는 프로덕트 라인 기술을 적용하여 모바일 콘텐츠 시스템을 분석, 설계함으로써 하루가 다르게 변모하는 모바일 콘텐츠 시스템을 보다 효율적이고 생산성을 높일 수 있는 방향에 대해서 모색해 보았다. UML 을 기반으로 한 PLUS 는 소프트웨어의 개발 및 적용하는 모든 활동의 중심적인 역할을 하는데 충분하다. 이러한 방법을 통해 시스템을 분석하고 재사용성을 높임으로서 보다 빠른 속도의 생산성을 갖추게 되고 체계적인 관리를 통해 다양한 분야에 적용성을 높일 수 있다. 이를 바탕으로 점차적으로 모바일을 비롯한 실시간 시스템 등, 앞으로 발전해가는 모

든 분야에 즉각적으로 사용하여 최고의 효율을 얻을 수 있을 것으로 기대된다.

향후 연구로는 콘텐츠 시스템의 범위를 확대하여 엔터테인먼트 이외의 분야도 고려한 설계를 수행하고 설계된 다른 시스템과의 연관을 통해 구체적으로 구현을 해 봄으로써 시장에 직접적으로 사용할 수 있는 실질적인 결과물을 거둘 수 있을 것이다.

## 참고문헌

- [1] 고정민, “모바일 시대의 콘텐츠 비즈니스”, 삼성경제연구소 Issue Paper, 2003
- [2] 김희연, “모바일 콘텐츠 시장과 mDRM 현황”, 정보통신정책 동향 p60-61, 2005
- [3] 조성문, “모바일 콘텐츠 기술”, 디지털타임스 퍼플 &칼럼, 2005
- [4] 김행근, 손이경, “프로덕트라인 기반의 모바일소프트웨어 개발 프로세스”, 정보처리학회논문지, 2005.
- [5] P.Clements and Linda Northrop, “Software Product Lines : Practice and Patterns”, Addison Wesley, 2002.
- [6] 최승훈, “컴포넌트 기반 소프트웨어 프로덕트 라인에서의 가변성”, 자연과학 논문집 제 10 권, 2003.
- [7] Colin Atkinson, “Component-based Product Line Engineering with UML”, Addison Wesley, 2002.
- [8] Mari Matinlassi, “Comparison of Software Product Line Architecture Design Methods : COPA, FAST, FORM, Kobra and QADA”, International Conference on Software Engineering, Proceedings of the 26<sup>th</sup> International Conference on Software Engineering, Vol.00, pp.127-136, 2004.
- [9] 차정은, 김철홍, 양영중, “기존 제품 계열 메소드의 특징 비교”, 정보처리학회 추계학술발표대회 논문집 제 11 권 제 2 호, 2004.
- [10] 이재준, 강교철, “프로덕트 라인 소프트웨어 개발 프로세스”, 정보과학회지 제 20 권 제 3 호, 2002.
- [11] 강교철, 이재준, 김병길, 김사중, “임베디드 시스템 개발 생산성 향상을 위한 소프트웨어 제품 라인 공학”, 정보과학회지 제 22 권 제 6 호, 2004.
- [12] Hanssan Gomma, “Designing Software Product Lines with UML, Addison Wesley, 2004.
- [13] H.M. Deitel, P. J. Deitel, T. R. Nieto, K. Steinbuhler “Wireless Internet & mobile business: how to program”, Prentice Hall, 2002.
- [14] 정재영, “모바일 콘텐츠 사업의 성공요건”, LG 주간경제, 2005.06.08
- [15] 포켓스페이스, “모바일 게임 서비스의 성공사례”, 모바일 비즈니스 성공사례 세미나, 2002
- [16] SKT, “모바일 쿠폰 사례”, 모바일 비즈니스 성공 사례 세미나, 2002
- [17] 이기돈, “벨소리 다운로드 사례”, 모바일 비즈니스 성공사례 세미나, 2002
- [18] 김한규, 김석구, 안중현, “위피스쿨과 함께하는 위 피 모바일 프로그래밍”, 영진.COM, 2004
- [19] Ari Jaaksi, “Developing Mobile Browsers in a Product Line”, IEEE Software, July-Aug, pp.73-80, 2002.