

쇼핑몰 시스템의 제품 라인 범위

조영호^o 윤병권 최윤석 정기원
 숭실대학교 컴퓨터학부

snatcher93@hanmail.net, superman@it.soongsil.ac.kr, yschoi@it.soongsil.ac.kr, chong@computing.soongsil.ac.kr

A Product Line Scope of Shopping Mall Systems

Youngho Cho^o Byungkwon Yoon Yunseok Choi Kiwon Chong
 School of Computing, Soongsil Univ.

요약

대부분의 B2C 쇼핑몰 시스템은 사용자 인터페이스에서 큰 차이를 보일 뿐 기능적/비기능적 요구사항과 아키텍처는 유사하며 쇼핑몰 시스템 구성요소들의 재사용성은 매우 높다. 이에 B2C 쇼핑몰 시스템의 제품 라인 구축 시 입력물인 1)제품 제약사항, 2)스타일, 패턴 및 프레임워크, 3)생산 제약사항, 4)생산 전략, 5)기존 구성품의 재구조화를 정의하고 B2C 쇼핑몰의 모델 분류와 기본적인 사용자 요구사항을 바탕으로 쇼핑몰 시스템의 공통점과 차이점을 제품 라인 범위로 제시한다. 일반적인 B2C 쇼핑몰 시스템이 제공하여야 하는 고객 정보 관리, 상품 정보 관리, 주문관리, 보안, 상품 운송 등의 기본 기능이 공통점으로 추출되었으며 가격의 고정 여부, 사용자 인터페이스, 멀티미디어 서비스 지원 여부, 포털 사이트와 비슷한 검색 기능, 온라인 지불 기능의 지원과 사용자 인터페이스 및 데이터베이스 스키마 등이 차이점으로 추출되었다.

1. 서론

대부분의 B2C 쇼핑몰 시스템은 콘텐츠나 판매 방식 및 사용자 인터페이스에서 큰 차이를 보일 뿐 고객 관리, 상품 정보 관리, 주문 정보 접수 등의 많은 기능적/비기능적 요구사항과 아키텍처는 유사하며 이러한 것들의 재사용성은 매우 높다. 따라서 B2C 쇼핑몰 시스템의 요구사항 중 공통점과 차이점을 식별하고 기본 아키텍처와 같이 시스템에 공통적인 부분을 공유한다면 생산성과 경제성을 높일 수 있다.

제품 라인(Product Line)은 공통 특성을 공유하는 제품을 효율적으로 개발하기 위한 기술로 특정 영역에 공통적으로 사용되는 부분의 공유를 통해 재사용성, 생산성 및 경제성을 높일 수 있는 방법이다.

제품 라인 프레임워크를 구축하는 활동인 핵심 구성품 개발 프로세스(Core Asset Development Process)는 1)제품 제약사항(Product Constraints), 2)스타일, 패턴 및 프레임워크(Styles, Patterns, and Frameworks), 3)생산 제약사항(Production Constraints), 4)생산 전략(Production Strategy), 5)기존 구성품의 재구조화(Inventory of Preexisting Assets)의 다섯 가지 입력물을 반복적인 프로세스로 처리하여 산출물인 제품 라인 범위(Product Line Scope), 핵심 구성품(Core Assets), 생산 계획(Production Plan)을 정의하는 프로세스이다. 세 가지 산출물 중 제품 라인 범위는 제품 라인을 구성할 제품의 리스트를 명시하고 제품 간의 공통점과 차이점을 식별한 것으로 나머지 두개의 산출물을 정의하기 위한 기반이 된다.

제품 라인 범위를 명확히 정의해야만 목적 영역에 대한 고품질의 제품을 라인에 구축할 수 있다. 제품 라인 범위가 너무 클 경우에는 변화를 수용할 수 있는 핵심 구성품을 추출하기 어려우며, 너무 작을 경우에는 변화를 수용하는데 제한을 받게 된다. 또한 일반화되고 예측 가능한 제품의 집합을 제품 라인의 대상으로 설정함으로써 생산성 및 경제성, 재사용성을 증가시킬 수 있다.

본 논문에서는 최근 활발히 진행되고 있는 B2C 쇼핑몰 시스템 개발에 제품 라인을 적용하기 위한 기초 단계로 쇼핑몰 시스템의 제품 라인 범위를 제시한다. 이를 위해 쇼핑몰 시스템의 요구사항을 분석하여 핵심 구성품 개발 프로세스의 다섯 가지 입력물을 정의하고, 쇼핑몰 시스템에서 공유 가능한 공통점과 공유가 불가능한 차이점을 식별하여 B2C 쇼핑몰 시스템을 위한 제품 라인의 제품 라인 범위를 정의한다.

2. 쇼핑몰 시스템을 위한 다섯 가지 입력물

2.1 제품 제약사항

제품 제약사항은 제품 라인을 구성하는 제품이 어떠한 공통점과 차이점을 가지고 있는 지와 어떠한 행위 특성을 제공해야 하는지를 나타낸 것이다.[2]

현재의 B2C 쇼핑몰 시스템은 <표 1>과 같이 Storefront Model, Auction Model, Portal Model, Dynamic Pricing Model로 구분할 수 있다.[3]

<표 1> B2C 쇼핑몰 시스템의 네 가지 모델[3]

Storefront Model	일반적인 인터넷 비즈니스 모델로 상인이 Web 상에서 물건을 판매할 수 있도록 transaction processing, security, online payment, information storage를 포함한다. Storefront Model을 만들기 위해서는 상인이 웹 상에서 물건의 카탈로그를 나열하고 그에 따른 주문 정보를 받을 수 있어야 하며, 상품을 고객에게 운송하고 고객 정보를 관리할 수 있어야 한다.
Auction Model	Internet User들이 log-on를 하여 입찰자나 판매자의 역할을 수행할 수 있는 포럼으로서의 사이트를 제공한다. 판매자로서 팔고자 하는 상품을 내걸고, 입찰자로서 원하는 상품을 선택하고 입찰가격을 제시한다.
Portal Model	사이트 방문자들이 원하는 정보를 찾을 수 있도록 도와준다. 이러한 모델은 일반적으로 Web 검색과 함께 스포츠, 뉴스, 날씨 등의 추가적인 정보도 제공한다. Portal Model은 Horizontal Portal과

	Vertical Portal의 두 가지로 나눌 수 있다. Horizontal Portal은 일반적인 Search Engine을 말하는 것으로 광범위한 정보를 통합하는 모델을 말한다. Vertical Model은 흥미있는 한 분야에 대한 특정 정보만을 제공하는 모델을 말한다.
Dynamic Pricing Model	상품의 가격이 고정적이지 않고 유동적인 모델. 고객이 원하는 상품에 대한 가격을 결정할 수 있는 Name-Your-Price 모델, 다양한 상인들을 검색하여 원하는 가격이나 서비스를 선택할 수 있도록 하는 Comparison Pricing Model, 고객들이 하나의 상품을 저가에 공동구매할 수 있는 Demand-Sensitive Pricing Model, 고객이 상품을 교환할 수 있도록 하는 Bartering Model, 항상 고객에게 상품의 가격을 할인해주는 Rebate Model, 상품과 서비스를 무료로 제공하면서 전적으로 광고 수입에 의존하는 모델, 게임이나 TV 방송, 라디오등의 콘텐츠를 무료로 제공하는 대신 다른 회사의 광고를 통해 수익을 산출하는 Offering Free Products and Services로 나눌 수 있다.

연구결과 B2C 쇼핑몰 시스템에 대한 기본적인 사용자 요구사항이 공통점으로 추출되었으며 <표 1>의 각 모델에 특화된 요구사항이 차이점으로 추출되었다. 각 모델간에 공통점으로 추출된 사항은 추후 핵심 구성품에 포함시킬 수 있으며 차이점으로 추출된 사항은 변경 가능하거나 대체 가능한 컴포넌트로 포함시킬 수 있다.

<표 2>는 <표 1>의 B2C 쇼핑몰 모델과 현재 운영중인 쇼핑몰 시스템에 대한 기본적인 사용자 요구사항에 따라 제품 제약사항을 정의한 것이다.

<표 2> B2C 쇼핑몰 시스템의 제품 제약사항

부분	기능	요구사항
Front Shop	고객등록	고객정보의 등록, 갱신 및 삭제 기능
	상품 정보 검색	상품명에 의한 검색 기능 제조 업체명에 의한 검색 기능 상품가격대별 검색 기능 상품에 대한 상세 정보 제공 기능
	장바구니	구입물품 추가 기능 구입물품 삭제 기능
	지불	상품 배송지 선택 기능 카드 정보 입력 기능
Back Office	배달정보	배송 정보 확인 기능 주문 취소 기능 이전 주문 기록 조회 기능
	고객관리	고객 정보의 생성, 수정 및 삭제 기능 고객 주문 현황 분석 기능
	상품관리	상품정보의 등록, 갱신 및 삭제 기능 재고 관리 기능 판매 정보 관리 기능
	지불관리	고객 정보 인증 기능 지불 처리 기능
판매통계	배달관리	배달 상태 모니터링 기능 배달 정보 관리 기능
	판매통계	판매 정보 검색 기능 기간(주, 월, 분기, 년)별 판매 정보 검색 기능 상품별 판매 자료 검색 기능

2.2 스타일, 패턴 및 프레임워크

현재 다양한 형태의 웹 어플리케이션 아키텍처가 사용되고 있으나 초기의 웹 사이트 아키텍처를 확장하여 사용자가 비즈니스 로직을 호출할 수 있도록 함으로써 서버의 비즈니스 상태를 동적으로 변경할 수 있도록 한 아키텍처가 널리 사용되고 있다. 이것은 웹 어플리케이션이 적어도 클라이언트 브라우저, 웹 서버, 어플리케이션 서버의 세 가지 중요한 아키텍처 컴포넌트를 가진다는 것을 의미한다.

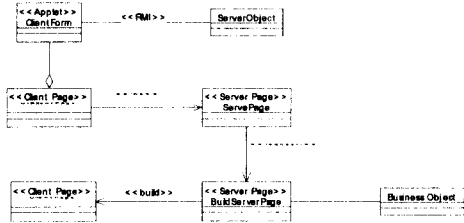
현재 운영되고 있는 다양한 종류의 웹 어플리케이션은 클라이언트 브라우저와 어플리케이션 서버측의 역할에 따라 Thin Web Client, Thick

Web Client, Web Delivery 패턴으로 나눌 수 있다.[1] <표 3>은 웹 어플리케이션의 세 가지 패턴 특징을 나타낸 것이다.
 초기의 쇼핑몰 시스템은 텍스트 기반으로 클라이언트 측에 제품의 목록을 나열하고 고객의 정보를 관리하는 단순한 형태의 것이 주를 이루었으나 현재의 쇼핑몰 시스템은 가상 현실, 멀티미디어 콘텐츠 제공, 컴포넌트 기술 등 웹 어플리케이션에서 수행할 수 있는 모든 기술을 채택하는 방향으로 발전하고 있다. 따라서 본 논문에서는 쇼핑몰 시스템의 모든 기능을 수용할 수 있는 Web Delivery 패턴을 사용하기로 한다.

<표 3> 웹 어플리케이션의 세 가지 패턴 [1]

Thin Web Client	클라이언트는 단지 표준 형식이 가능한 웹 브라우저를 가지고 있으면 되고 모든 비즈니스 로직은 서버에서 처리된다.
Thick Web Client	Thin Web Client에 비해 많은 양의 비즈니스 로직이 클라이언트에서 처리된다. 클라이언트는 비즈니스 로직을 처리하기 위해 Dynamic HTML, Java Applet, Active X 컨트롤 등을 사용한다. 서버와의 통신은 HTTP를 통해 수행된다.
Web Delivery	클라이언트와 서버간의 통신은 HTTP 외에도 IIOP나 DCOM과 같은 프로토콜이 사용된다. 웹 브라우저는 주로 분산 객체 시스템의 전달과 컨테이너로서 작용한다.

<그림 1>은 UML의 확장 메커니즘을 사용하여 제품 라인의 Web Delivery 패턴에 대한 템플릿을 설계한 것이다. 이 패턴은 스타일, 패턴, 프레임워크에 포함된다.

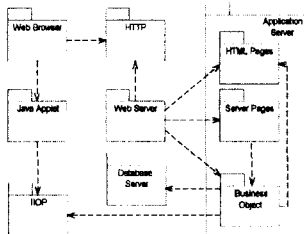


<그림 1> 쇼핑몰의 Web Delivery 패턴

UIClientPage 클래스는 입력폼을 화면에 출력하고 사용자의 입력을 받아 이를 서버측에 전송한다. ServerPage 클래스는 클라이언트 측의 요청을 받아 이를 BuildServerPage에 전달하며 BuildServerPage는 비즈니스 오브젝트를 나타내는 BusinessObject 클래스를 사용하여 ClientPage 클래스를 생성한 후 이를 사용자에게 전송한다.

UIClientPage는 입력폼을 구성하기 위해 Applet인 ClientForm을 포함하며 이는 서버 측의 오브젝트를 나타내는 ServerObject와 RMI를 통해 동적인 처리를 수행한다.

설계자는 <그림 1>의 패턴을 자신의 쇼핑몰 시스템에 맞도록 정제함으로써 Thin Web Client, Thick Web Client, Web Delivery의 세 가지 패턴 모두를 나타낼 수 있다.



<그림 2> B2C 쇼핑몰 시스템의 기본 아키텍처

쇼핑몰 시스템 아키텍처를 구축하기 위해서는 쇼핑몰 시스템에 대한 요구사항을 분석하고, 이를 바탕으로 아키텍처를 구성하는 컴포넌트들과 그들 사이의 관계를 정의해야 한다.[4] <그림 2>는 Web Delivery 패턴을 수용하는 쇼핑몰 시스템의 아키텍처를 제시한 것이다.

웹 브라우저는 고객과 쇼핑몰 점원이 사용하는 일반적인 사용자 인터페이스 역할을 수행한다. HTTP는 인터넷 상에서 클라이언트와 서버 사이의 가장 일반적인 연결로서, 비 지속성 연결 방식을 지원한다. HTML 페이지와 서버 페이지는 HTTP 연결을 통해 전달된다. 웹 서버는 브라우저를 통해 쇼핑몰 시스템에 연결하기 위한 연결점 역할을 수행하며 웹 브라우저의 요청을 어플리케이션 서버로 전달한다. 웹 브라우저의 요청이 서버 측의 특정 프로세스를 필요로 하는 경우 웹 서버는 웹 브라우저의 요청을 어플리케이션 서버로 전달한다. 어플리케이션 서버는 웹 브라우저의 요청을 처리할 수 있는 비즈니스 객체를 호출, 구동시킨다. 호출된 비즈니스 객체는 데이터베이스 서버에게 질의를 전달하고, 데이터베이스 서버는 이를 처리하여 질의를 전달했던 비즈니스 객체에게 결과를 전달한다. 처리 결과는 비즈니스 객체에 의해 서버 페이지로 구성되어 웹 서버로 전달되고, 웹 서버는 웹 브라우저로 처리 결과를 전송한다. 만일 웹 브라우저 측에서 그래프와 같은 동적인 요소의 표시가

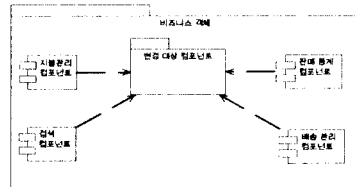
필요할 경우 자바 애플릿이 사용될 수 있다. 애플릿은 웹 서버 측에서 웹 브라우저로 필요에 따라 전송된다. 웹 브라우저로 전송된 애플릿은 JRMP(Java Remote Method Protocol) 또는 CORBA IIOP를 사용하여 서버 측의 다른 객체들과 통신할 수 있다. 어플리케이션 서버는 비즈니스 객체의 실행을 담당한다.

제품 라인 자체가 생산공정을 나타내기 때문에 추출한 쇼핑몰 시스템의 공통점과 차이점을 바탕으로 변화를 수용할 수 있는 아키텍처를 설계하는 것이 가장 중요하다.[5] 이에 따라 개발자는 2.1절의 제품 제약사항에서 정의한 쇼핑몰 시스템의 요구사항 중 공통점 뿐만 아니라 목적 시스템의 특징인 차이점을 아키텍처에 반영해야 한다. 이를 위해 아키텍처의 공유 가능한 공통부분은 핵심 구성품으로 구성하고 제품 개발 프로세스(Product Development Process)에서 변경되거나 대체될 가능성이 있는 부분은 <그림 3>과 같이 UML의 패키지 다이어그램에 아키텍처에 표현한다.

기본 아키텍처 중 변경이 빈번한 부분은 어플리케이션 서버의 HTML 페이지나 서버 페이지, 비즈니스 객체, Java 애플릿 부분이며 HTTP, 웹 서버, 데이터베이스 서버, IIOP의 경우 거의 변화가 일어나지 않는다. 따라서 Java 애플릿이나 HTML 페이지, 서버 페이지의 경우 아키텍처에 표현할 때 어떠한 핵심 구성품도 포함하지 않는 빈 패키지로 표현될 수 있다. 이것은 제품 개발 프로세스 시에 구축하려는 각 쇼핑몰 시스템에 적합하도록 설계와 된다.

비즈니스 객체 패키지의 경우 서버측의 비즈니스 로직을 처리할 비즈니스 객체를 포함하고 있다. 일반적인 쇼핑몰 시스템의 경우 고객 정보 및 상품 정보, 통계 및 검색에 대한 비즈니스 객체들이 필요하다. 이러한 객체들 중 구축하려는 제품 라인에서 재사용 가능한 객체들은 비즈니스 객체 패키지 내에 수록되며 변경되거나 대체 가능한 것들은 <그림 3>과 같이 패키지를 사용하여 우선 표시한 후 구축 시에 각 시스템에 적합한 객체들로 이를 재구성할 수 있다.

<그림 3>은 비즈니스 객체 패키지를 설계한 예이다. 제품 라인 구축 시 지불관리 컴포넌트와 판매통계 컴포넌트, 검색 컴포넌트, 배송관리 컴포넌트는 제품 라인 전체에 걸쳐 재사용 가능한 부분으로 UML의 컴포넌트 표기를 빌려 표현하였으며, 시스템의 사용자 요구사항에 따라 개발되거나 COTS 컴포넌트로 대체될 부분은 패키지로 표현하였다.



<그림 3> 비즈니스 객체 패키지의 설계 예

제품 라인을 통해 쇼핑몰 시스템을 구축할 경우 변경 대상 컴포넌트 패키지는 각 시스템의 차이점을 반영하게 될 컴포넌트로 대체될 것이다. 즉, <그림 3>의 쇼핑몰 시스템이 멀티미디어 콘텐츠를 제공한다면 이에 대한 새로운 컴포넌트가 변경 대상 컴포넌트 패키지에 포함되고 기존의 핵심 구성품으로 정의된 컴포넌트와 사용(usage) 관계가 형성될 것이다.

2.3 생산 제약사항

<표 4> 쇼핑몰 시스템 구축 시 사용 가능한 COTS 제품의 예

Categories	Products
Front Shop	Client & Clerk Internet Explorer, Navigator, Mosaic
Operating System	Windows NT, Windows 2000, UNIX, LINUX
Web Server	IIS, Apache
Application Server	Web BlueStone Sapphire/Web application server IBM WebSphere Netscape Application Server(Kiva) SilverStream SilverStream Application Server Sun/NetDynamics NetDynamics BEA Systems WebLogic Application Server(Tengah) TPM BEA Systems Tuxedo, Top End IBM CICS Microsoft MTS와 COM+ ORB Iona ORBIX Inprise Visibroker OTM (TPM+ORB) Oracle Oracle Application Server BEA Systems M3(Iceberg) IBM Component Broker Iona ORBIX/OTM
Database Server	Oracle, DB2, Sybase, Infomix, Ingres, MS SQL Server
ORB	Java IDL, OrbixWeb, Visibroker
DB Connectivity	ODBC, JDBC
Payment Gateway	XPAY Gateway 2.0, Credit Payment Gateway 2.0, Globeset Gateway 1.4, IBM Payment Gateway 1.2

생산 제약사항은 제품이 만들어질 하부조직과 사용할 COTS, 재사용할 기존 컴포넌트의 목록을 정의하며 이 목록의 내용 중 제품 라인에서 공유되는 컴포넌트는 핵심 구성품으로 포함된다.[2]
 <표 4>는 쇼핑몰 시스템 구축 시 사용 가능한 COTS 제품의 예를 나타낸 것이다.

2.4 생산 전략

생산 전략이란 핵심 구성품을 실제화하기 위한 종합적인 접근방법을 의미한다.[2] 생산 전략에는 크게 기존의 컴포넌트를 조립하여 제품을 생산하는 상향식 방식과 개발할 제품의 도메인을 분할하여 요구사항을 상세화하는 하향식 방식이 있다.

B2C 쇼핑몰 시스템의 경우 <그림 2>의 기본 아키텍처를 개발할 시스템의 사용자 요구사항에 맞추어 정제한 후, 사용할 COTS, 재사용할 기존 컴포넌트의 목록을 생산 제약사항으로 정의한다. 생산 제약사항에 정의된 목록 중에서 정제한 아키텍처의 요구사항에 부합되는 컴포넌트나 COTS를 파악하고 컴포넌트화할 기능을 분석하여 개발할 컴포넌트를 식별한다. 식별된 COTS나 컴포넌트는 기존 컴포넌트의 사용, COTS 획득, 신규 컴포넌트의 설계 및 구현 등의 방법을 사용하여 획득되어 진다. 또한 사용자 요구사항이 변경된 경우 변경된 도메인을 바탕으로 개선 또는 대체되어야 하는 컴포넌트를 식별해야 한다.

이처럼 B2C 쇼핑몰 시스템의 구축 과정은 개발하고자 하는 제품군의 특성을 파악하고 이에 따라 아키텍처를 정제한 후 필요한 컴포넌트에 대한 획득 단계들 거처므로 상향식 방식보다는 하향식 방식을 사용하는 것이 적절하다.

따라서 B2C 쇼핑몰 시스템의 생산 전략은 하향식 접근 방식을 전제로 하며 아키텍처에 대한 정제 방법 및 그에 대한 정책을 포함하는 것이 필수적이라 할 수 있다.

2.5 기존 구성품의 재고조사

기존 구성품의 재고조사는 제품 라인 구축 시에 이용 가능한 모든 소프트웨어와 제품 라인의 구성 요소를 의미한다.[2] 기존 구성품의 재고조사에 포함되는 항목은 단순히 소프트웨어 컴포넌트와 같은 개발 부품만을 의미하는 것이 아니라 모든 사용 가능한 라이브러리, 프레임워크, 알고리즘, 도구 등을 포함한다.

<표 5>는 B2C 쇼핑몰 시스템의 기존 구성품의 재고조사 중 개발 관련 도구의 예를 나타낸 것이다.

<표 5> B2C 쇼핑몰 시스템의 기존 구성품의 재고조사의 예

디자인 도구	Cool Joe, Rational Rose, Plastic Software
개발 언어	ASP, PHP, CGI, Java Script
개발 기술	VRML, Java Applet, Active-X
2D 모델링 도구	Photoshop, Paint Shop Pro
3D 모델링 도구	Pioneer Pro, True Space, Cosmo World, 3D Studio MAX, Website Builder, V-Realm Builder, Ray Dream Studio
웹 저작 도구	Hot Dog, Front Page, Web Edit Pro, 나뭇 웹 에디터

개발 도구 및 개발 언어의 선정은 시스템 구축 방법, 개발 방법, 의사소통의 수단을 정의하는 것이다. 따라서 표준적인 도구를 선정함으로써 생산 공정의 일관성을 유지할 수 있으며 발생할 수 있는 오류의 종류를 줄일 수 있다.

3. 제품 라인 범위

제품 라인 범위를 정의하기 위해서는 제품 라인을 구성할 제품의 목적을 명시하고 제품간의 공통점과 차이점을 식별하여야 한다[2]. 이를 사용하여 구축하게 될 쇼핑몰 시스템의 특성을 파악하고 이러한 특성을 기반으로 제품 제약사항에 정의된 구축 대상 시스템 간의 공통점과 차이점을 정제한다.

제품 라인 범위를 통해 추출된 공통점과 차이점을 바탕으로 제품 라인을 구성할 핵심 구성품과 대체 가능한 컴포넌트가 추출된다. 이를 사용하여 <그림 1>의 Web Delivery 패턴과 <그림 2>의 기본 아키텍처의 정제가 가능하다. 정제된 패턴과 아키텍처 역시 핵심 구성품의 일부로 포함되며 이들은 새로운 사용자 요구사항의 발생과 기존 사용자 요구사항의 변화할 수용할 수 있는 구조를 유지해야 한다. 또한 생산 제약사항과 기존 구성품의 재고조사 중 제품 라인 범위를 통해 공통점으로 추출된 사항과 일치하는 요소는 핵심 구성품으로 포함한다.

결국 제품 라인 범위란 제품 라인에 걸쳐 재사용될 핵심 구성품의 선별 기준을 결정하는 것이라 할 수 있다. 따라서 제품 라인 범위 정의에 오류가 있을 경우 제품 개발 프로세스 시에 시스템 구축이 어려워지며 제품 라인 구축을 통해 얻을 수 있는 생산성 및 경제성, 재사용성이 반감된다.

본 논문에서는 제품 라인의 구축 대상물 일반적인 B2C 쇼핑몰 시스템으로 전제하였기 때문에 특별히 제품 라인을 구성할 제품의 목록을 명시하지 않았으며 제품 제약사항에서 설명한 바와 같이 B2C 쇼핑몰 시스템의 네 가지 모델과 일반적인 사용자 요구사항을 바탕으로 공통점과 차이점을 추출하였다. 본 논문에서 정의한 B2C 쇼핑몰 시스템의 제품 라인 범위는 <표 6>과 같다. 제시한 제품 라인 범위는 일반적인 B2C 쇼핑몰 시스템 영역을 기반으로 공통점과 차이점을 추출한 것이므로 제품 라인화할 영역이 특정 컨텍스트에 의존적이거나 특정 모델에 특화된 경우 현재 제품 라인 범위를 수정 및 보완하여야 한다.

<표 6> B2C 쇼핑몰 시스템의 제품 라인 범위

공통점	웹 상에서 물건 판매 Transaction Processing Security Information Storage
	웹 상에서 물건의 카탈로그 나열 주문 정보 접수 상품 운송 고객 정보 관리
	사용자 인터페이스 데이터베이스 스키마 온라인 지불 여부 가격이 고정적인지의 여부 고객이 원하는 상품의 가격을 결정할 수 있는지의 여부 다양한 사이트의 검색 기능을 통해 원하는 가격이나 서비스를 선택할 수 있는지의 여부 하나의 상품에 대해 공동구매 가능 여부 고객이 판매되는 상품을 자신이 보유한 물품과 교환하거나 금전으로 지불할 수 있는지의 여부 항상 상품의 가격을 할인하는지의 여부 마일리지 적립 기능 지원 여부 멀티미디어 서비스 지원 여부 상품 검색 외에 광범위한 정보의 검색을 지원하는지의 여부 상품 검색 외에 한 분야에 대한 검색만을 지원하는지의 여부 상품에 대한 판매 시스템 지원 여부 이메일 및 홈페이지의 제공 여부

<표 6>에서 보는 바와 같이 일반적인 B2C 쇼핑몰 시스템이 제공하여야 하는 고객 정보 관리, 상품 정보 관리, 주문관리, 보안, 상품 운송 등의 기본 기능이 제품 라인 구축을 위한 공통점으로 도출되었으며, 가격의 고정 여부, 사용자 인터페이스, 멀티미디어 서비스 지원 여부, 포털 사이트의 검색 기능, 온라인 지불의 지원 등 위에서 제시한 네 가지 기본 모델에 의존적인 기능과 모든 웹 어플리케이션에서 항상 재작성되어야 하는 사용자 인터페이스 및 데이터베이스 스키마 등이 차이점으로 추출되었다. <표 6>에서 열거한 공통점과 차이점에 따라 입력물인 제품 제약사항, 스타일, 패턴 및 프레임워크, 생산 제약사항, 생산 전략, 기존 구성품의 재고조사 항목을 재배치함으로써 제품 라인 구축에 적합한 핵심 구성품을 추출할 수 있다.

<표 6>의 제품 라인 범위를 검증하기 위해 현재 운영되고 있는 대표적인 쇼핑몰 시스템의 목록을 만든 후 각 쇼핑몰 시스템의 기능 및 특징을 표시한 매트릭스를 작성하였다. 이 매트릭스에 나타난 기능을 제품 라인 범위에 나타난 공통점과 차이점으로 분류한 결과 공통점으로 추출된 항목은 모든 쇼핑몰 시스템이 지원하고 있었으며 차이점으로 추출된 항목은 몇몇 시스템만이 지원하고 있었다.

4. 결론

본 논문에서는 B2C 쇼핑몰 시스템을 위한 제품 라인 구축의 기반이 되는 제품 라인 범위를 제시하였다. 이를 위해 핵심 구성품 개발 프로세스의 다섯 가지 입력물인 제품 제약사항, 스타일, 패턴 및 프레임워크, 생산 제약사항, 생산 전략, 기존 구성품의 재고조사를 정의하고, 쇼핑몰 시스템을 Storefront Model, Auction Model, Portal Model, Dynamic Pricing Model의 네 가지 모델로 구분하였으며 B2C 쇼핑몰 시스템에 대한 기본적인 사용자 요구사항을 도출하고 각 모델의 공통점과 차이점을 식별하였다.

일반적인 B2C 쇼핑몰 시스템이 제공해야 하는 공통점으로는 고객 정보 관리, 상품 정보 관리, 주문관리, 보안, 상품 운송 등의 기본 기능이 추출되었으며 가격의 고정 여부, 사용자 인터페이스, 멀티미디어 서비스 지원 여부, 포털 사이트 기능 여부, 온라인 지불의 지원 등 위에서 제시한 네 가지 기본 모델에 의존적인 기능과 모든 웹 어플리케이션에서 항상 재작성되어야 하는 사용자 인터페이스 및 데이터베이스 스키마 등이 차이점으로 추출되었다.

위에서 제시한 다섯 가지 입력물 중 제품 라인 범위에서 공통점으로 추출된 사항은 핵심 구성품으로 포함시키고 차이점으로 추출된 사항은 변경 가능하거나 대체 가능한 컴포넌트로 포함시킴으로써 B2C 쇼핑몰 시스템의 핵심 구성품과 변경 부분을 식별해낼 수 있다.

향후에는 위의 다섯 가지 입력물과 추출된 제품 라인 범위를 바탕으로 핵심 구성품과 각 핵심 구성품에 대한 부차 프로세스의 정의에 관한 연구가 필요할 것이다. 이를 위해서는 현재 쇼핑몰 시스템 구축에 이용되는 핵심 구성품의 사용 사례와 웹 어플리케이션 개발에 관련된 응용기술에 대한 연구가 선행되어야 할 것이다.

5. 참고문헌

[1] Jim Conallen, "Building Web Applications with UML", Addison Wesley, 1999.
 [2] Carnegie Mellon Software Engineering Institute. "A Framework for Software Product Line Practice- Version 3.0", <http://www.sei.cmu.edu/plp/framework.html>
 [3] H.M.Deitel, P.J.Deitel, T.R.Nieto, "e-Business & e-Commerce HOW TO PROGRAM", Prentice Hall, 2001
 [4] 정기원, 최윤식, "아키텍처 정제에 기반한 쇼핑몰 시스템 개발 모델", 한국프로젝트관리기술회, 프로젝트 관리 기술 논문집, 제1권, 제1호, pp. 5-14, 2000, 10
 [5] 조영호, 윤병권, 최윤식, 정기원, "B2C 쇼핑몰 시스템을 위한 프로덕트 라인", 정보과학회, 제28회 임시총회 및 춘계학술발표회, 2001, 5