

經營學碩士 學位論文

XML기반의 상품정보 표준화를 위한 전자카탈로그 설계에 관한 연구

指導教授 金 河 均

이 論文을



提出함

2008年 8月

釜慶大學校 大學院

情報 시스템 學科

金 省 材

金省材의 經營學碩士 學位論文을 認准함

2003年 6月

主 審 經營學博士

李 賢 圭



委 員 經營學博士

金 河 均



委 員 工學博士

金 榮 鳳



[차 례]

표 차 례	3
그 림 차 례	4
Abstract	6
 제 I 장 서 론	1
1. 1 연구배경	1
1. 2 연구 목적 및 범위	3
 제 II 장 기반 기술	4
2. 1 XML의 등장배경 및 정의	4
2. 2 XML의 특징	7
2. 3 XML 관련 표준	11
 제 III 장 전자카탈로그의 표준화와 선행연구	14
3. 1 전자카탈로그 개념 및 표준화	14
3. 2 전자카탈로그 관련 선행연구	17
 제 IV 장 표준으로 지정된 전자카탈로그 비교	22
4. 1 cXML(commerce XML)	22
4. 1. 1 전자카탈로그 DTD	23
4. 1. 2 전자카탈로그 관리	31
4. 2 xCBL (xml Common Business Library)	32
4. 2. 1 전자카탈로그 DTD	32
4. 2. 2 전자카탈로그 관리	38
4. 3 RosettaNet PIP2A와 PIP2B	39

4. 3. 1 전자카탈로그 DTD	40
4. 3. 2 전자카탈로그 관리	47
4. 4 전자카탈로그 비교	48
 제 V 장 XBC 전자카탈로그 설계	52
5. 1 XBC 전자카탈로그 모델 원형	52
5. 2 XBC 전자카탈로그 모델	55
 제 VI 장 XBC 전자카탈로그 적용	61
 제 VII 장 결 론	77
 참 고 문 헌	80

[표 차 례]

[표 1] XML 문서의 구조	7
[표 2] HTML/SGML/XML의 비교분석	10
[표 3] 전자카탈로그 관련 표준	17
[표 4] 통신판매업에 대해 법률이 지정한 정보표시항목	20
[표 5] IndexItemAdd 엘리먼트 예	26
[표 6] IndexItemPunchout 엘리먼트 예	28
[표 7] Supplier 엘리먼트 예	29
[표 8] Index 엘리먼트 예	30
[표 9] Contract 엘리먼트 예	31
[표 10] CatalogHeader 엘리먼트 예	34
[표 11] CatalogSchema 엘리먼트 예	36
[표 12] CatalogData(Product) 엘리먼트 예	38
[표 13] 전자카탈로그 비교	49
[표 14] 한국전산원 “전자카탈로그의 게시항목 권고안”	54
[표 15] DTD파일의 내용	61
[표 16] XBC에 도서상품을 적용한 예	64
[표 17] XBC에 컴퓨터상품을 적용한 예	67
[표 18] cXML에 컴퓨터상품을 적용한 예	70
[표 19] xCBL에 컴퓨터상품을 적용한 예	73
[표 20] cXML · xCBL · XBC 게시 권고안 표	75
[표 21] cXML · xCBL · XBC 게시 권고안 비교표	76
[표 22] 중복적 표현 제거 CatalogHead	78

[그림 차례]

[그림 1]	전자카탈로그 표준의 효과	16
[그림 2]	cXML 카탈로그 DTD	24
[그림 3]	IndexItemAdd 엘리먼트	25
[그림 4]	IndexItemDelete 엘리먼트	26
[그림 5]	IndexItemPunchout 엘리먼트	27
[그림 6]	ProductCatalog DTD	33
[그림 7]	CatalogHeader 엘리먼트	33
[그림 8]	CatalogSchema 엘리먼트	35
[그림 9]	CatalogData 엘리먼트	37
[그림 10]	Pip2A1ProductInformationNotification DTD	41
[그림 11]	PartnerRoleDescription 엘리먼트	42
[그림 12]	Pip2A1ProductResourceUpdate 엘리먼트	42
[그림 13]	SalesCatalog 엘리먼트	43
[그림 14]	ProductLineItem	44
[그림 15]	ComponentTechnicalSpecification 엘리먼트	45
[그림 16]	SubComponent 엘리먼트	46
[그림 17]	Quantity와 Quality 엘리먼트	46
[그림 18]	XBC 모델 원형	53
[그림 19]	XBC(XML Based Catalog) DTD	55
[그림 20]	ClassifierIdentity 엘리먼트	56
[그림 21]	DefaultInfo 엘리먼트	57
[그림 22]	Characteristic 엘리먼트	58
[그림 23]	Component 엘리먼트	59
[그림 24]	SellingInfo 엘리먼트	60
[그림 25]	가장 최근에 전자상거래로 구매한 상품	62

[그림 26]	XBC에 도서상품을 브라우저로 파싱한 예	65
[그림 27]	XBC에 컴퓨터상품을 브라우저로 파싱한 예	68
[그림 28]	cXML에 컴퓨터상품을 브라우저로 파싱한 예	71
[그림 29]	xCBL에 컴퓨터상품을 브라우저에 파싱한 예	74

A Study on Designing the Electronic Catalog for Standardizing Goods Information Based on XML

Sung-Jae, Kim

*Department of Information System, Major in Management Information
Systems, Graduate School of Pukyong National University*

Abstract

World Wide Web, providing the hyper medium with its superb user interface function, has led the internet to the public, and E-commerce to a point of commercialization of the internet. In the type of E-commerce, one of the essential technologies is the capability of enabling users to pertinently represent their electronic catalog, and it is necessary to establish standardization to exchange and share the information in the catalogs.

In the catalogs that have presently been under research, the product information cannot be represented in a classified form but it also fails to be well defined, lacking the description of detail information for various products in a systematic fashion. And the product information is duplicated in its representation. In this study, an implementing model is to be proposed to analyze and classify the information that can be in use of describing various products, and to describe an integrated product information for various types of products while making up for

the weak points of presently available researches. Also, this study shows that it is possible for the proposed model to represent various units of product information with presently proposed models applied to computers and book products in high demand of transaction, and by implementing electronic catalogues in the existing researches to prove its efficiency.

The use of proposed model can make it possible to represent various units of product information as one electronic catalog model, with a result that it gives both convenience and efficiency to similar types of apparatus i.e., internet shopping malls where they deal in various kinds of product.

제 I 장 서론

1. 1 연구배경

인터넷 인구의 빠른 증가와 함께 인터넷을 이용한 상거래가 시간적, 공간적 제약을 극복한 새로운 경제 활동으로 부각되고 있다. 이러한 시대적 환경에 맞추어 전자상거래 시장의 규모도 이례 없이 증가하고 있다.

2003년 2월 통계청의 조사결과에 따르면 인터넷 쇼핑몰의 사업체 수는 2002년 12월말 현재 2,896개로 2001년 12월말(2,166개)에 비해 730개(33.7%) 증가한 것으로 조사되었고 인터넷 쇼핑몰의 거래규모는 2002년 1월~12월 중 총 6조 299억원으로 2001년 12월말(3조3,471억원)에 비해 2조 6,828억원(80.2%)이 증가한 것으로 집계되었다. 이중 B2C의 거래규모는 5조 433억원으로 전년도에 비해 2조 4,632억원이 증가하였다.

이와 같은 전자상거래는 기술적 기반구조의 뒷받침이 이루어져야 하며, 현재 전자상거래를 구축하기 위해 다양한 기술들이 사용되고 있다. 이러한 기술들로 만들어진 웹 어플리케이션이 최근 몇년 사이에 일반화되어 가고 있다. 기업간의 원활한 전자상거래를 위해서는 전자상거래 절차, 거래양식, 시스템 환경 등의 전자상거래 기반구조가 구축되어야 하며, 특히 상품을 거래하기 위하여 구축될 전자카탈로그는 풍부한 상품정보와 정보의 활용을 가능케 하는 서비스를 제공해야 한다. 즉, 전자카탈로그에 접근하는 각종 전자상거래 시스템 및 다른 전자카탈로그와 정보를 교환할 수 있어야 한다. 이렇게 정보교환과 공유를 위해서는 표준화의 필요성이 절실하다. 전자상거래의 경우 선도 기업들간의 자율적인 협의체를 통한 사실 표준이 보편적으로 사용되는 경향이 두드러지고 있다. 하지만 우리나라의 경우는 기업들간의 지나친 경쟁으로 인해 전략적 제휴나 자발적 표준 제정활동이 미흡하고, 관련 기업, 연구소, 기관간 수평적 협의가 원활하지 못하여 기술

개발 및 표준화사업을 추진함에 있어 시너지 효과를 낳지 못하고 있는 실정이다.

그래서 많은 기업과 단체에서 각자의 표준을 마련하고 이를 지원하기 위한 전자상거래 표준 프레임워크를 개발하고 있다. 현재 개발 중인 프레임워크로는 CommerceNet의 eCo, RosettaNet, Microsoft의 BizTalk, UN/CEFACT와 OASIS의 ebXML 등을 들 수 있다.

위의 프레임워크는 모두 XML(eXtensible Markup Language)에 기반하여 기업간의 모든 거래에 대한 표준 문서를 기술한다. 웹 상에서 데이터 교환을 목적으로 개발된 XML은 HTML(Hypertext Markup Language)과는 달리 사용자 정의 태그 제공, 구조적 문서 기술, 문서구조 검증 등의 기능을 제공함으로써, 전자상거래와 같은 데이터 처리 분야에서 적극 활용되고 있다. 하지만 XML의 이와 같은 장점에도 불구하고 태그의 의미에 대한 표준이 정의되어 있지 않으면, 서로 다른 조직과 사용자간에 의미 전달을 명확히 하는 것은 불가능하다. 따라서 공통적으로 사용할 수 있는 태그나 용어를 표준화하는 것이 필요하다. 현재 이러한 표준화 작업이 cXML (commerce XML), xCBL(xml Common Business Library), RosettaNet, ICE (Information and Exchange), OCF(Open Catalog Format), eCX(electronic Catalog XML) 등에서 이루어지고 있다.

본 논문에서는 전자상거래를 위한 전자카탈로그에 대해서 다룬다. 이에 관한 연구는 국내에서는 아직까지 미흡하며, 국외의 경우 cXML, xCBL, RosettaNet 등에서 기업간 전자상거래를 위한 연구가 이루어지고 있지만, 상품의 정보가 분류되어 표현되지 않고 구체적이지 못하여, 상품의 세부적인 정보를 체계적으로 상세 기술하는 것에는 문제점이 있다.

1. 2 연구 목적 및 범위

B2B 또는 B2C 전자상거래를 구축할 때 공급자, 중계자, 구매자가 주고 받는 정보의 핵심은 상품정보이다. 전자상거래에서 상품을 공급하려는 공급자, 중계자는 개별상품을 가장 잘 나타낼 수 있는 정보들을 결정하고, 이를 구매자가 받아볼 수 있는 전자적 형태의 데이터로 작성하여 전송하며, 구매자는 전송 받은 전자카탈로그로부터 상품의 상세한 정보를 얻어 구매를 결정할 수 있다. 전자카탈로그를 제작하는데 있어서 가장 중요한 것은 표준이다. 개별적인 상품DB 구축에 따른 시간, 비용의 중복투자를 방지하고 상품정보의 공동활용을 위해서는 특히 표준화가 중요하며, 전자카탈로그는 전자상거래 환경의 정보 인프라로서 중요성을 지니기 때문이다. 표준이 아닌 개별적인 방법으로 전자카탈로그를 개발한다면 전자상거래의 활성화에 장애가 됨은 물론 비용면에서도 큰 손실이 될 것이다.

이렇게 상품정보의 표준화를 위해 XML을 기반으로 상품정보를 효과적으로 기술할 수 있는 모델 XBC(XML Based Catalog)을 제안한다. XBC는 전자카탈로그의 구성요소 중 내용적 구성요소인 상품분류 및 속성표준을 중심으로 모델화 한다. 기술적 구성요소에 해당하는 표준들은 여러 방식 중에서 하나를 선정하면 되지만 제품분류 및 속성표준은 통일하기 어려운 부분이기 때문이다.

다음의 3가지는 모델 설계시 고려된 유형화의 기준이다.

첫째, 유형들간에는 상호 배타적이고 완전해야 한다.

둘째, 상품정보의 공통게시정보와 고유게시정보를 모두 고려하여 모델의 범용성을 높이는 것이다.

셋째 상품속성은 상품자체에 속하는 것으로 국한하였다. 즉, 배송, 지불처럼 판매업체와 관계된 속성들은 고려하지 않고 상품정보만을 고려하였다.

제 II 장 기반 기술

2. 1 XML의 등장배경 및 정의

모든 문서는 어떠한 형태로든 구조를 가지는데 논리적인 구조를 가지거나 물리적, 외형적인 구조를 가지게 마련이다. 이러한 구조적인 문서를 표준화된 공개 형식을 사용하여 작성하기 위한 노력이 다각적으로 연구되어 왔고, 그러한 노력의 결과로 1986년 SGML(Standard Generalized Markup Language)이 ISO의 표준으로 제정되었다. SGML은 문서의 유효성을 정형적으로 증명할 수 있고, 복잡한 문서를 다룰 수 있을 만큼 구조적이며, 방대한 정보의 관리를 지원할 수 있는 확장성을 가지도록 설계되었다.

SGML은 시스템 또는 플랫폼에 독립적으로 동작하고 문서의 구조를 저장할 수 있기 때문에 문서의 구조를 기반으로 한 다양한 응용에 사용할 수 있고, 업계의 표준으로 사용되는 경우도 많다(Connolly, Khare and Rifkin, 1997).

그러나 SGML 문서는 그 구조에 있어서 미리 정해진 것 이외에 사용자가 선택할 수 있는 부분이 많기 때문에 문서를 보내는 쪽과 받는 쪽이 해당 문서의 형식에 대하여 똑같은 정보를 가지고 있어야 한다. 또한 구조 자체가 복잡하기 때문에 SGML 전체를 지원하는 소프트웨어의 개발이 용이하지 않으며, 웹을 위한 목적으로 설계되지도 않았다. 이러한 문제로 현재까지 웹 문서를 SGML에 기반을 둔 HTML을 사용하여 작성되고 있다. HTML은 하이퍼텍스트, 하이퍼미디어의 기능을 지원하고, 누구나 사용할 수 있을 만큼 간단하며, 특별한 데이터 타입이 사용되지 않는 단순한 텍스트이기 때문에 이식성이 좋고 사용이 편리하다. 그러나 SGML의 극히 제한된 부분집합인 HTML은 전송의 효율을 위하여 그 내부적인 구조를 상

당 부분 제한하였기 때문에 구조가 복잡한 데이터를 표현하기 어렵다는 단점이 있다. 따라서 HTML과 마찬가지로 웹 상의 문서를 작성하기 위한 좀 더 구조적인 마크업 언어로서 XML이 등장하였다.

이렇듯 XML은 SGML과 HTML의 한계점을 극복하기 위해 탄생한 언어라고 할 수 있다. SGML의 한계점과 HTML의 한계점을 살펴보면 다음과 같다.

(1) HTML의 한계점

- HTML은 확장되지 않기 때문에 개발자들이 임의대로 태그를 확장할 수가 없다.
- HTML은 중첩 구조를 지원하지 못한다. 또한 문서의 구조가 단순하기 때문에 전문 검색이 용이하지 못하다.
- 독특한 형식의 웹 문서나 기존의 스타일과 다른 포맷을 요구하는 클라이언트가 있을 수 있다. 이때 레이아웃 정보와 같은 물리적인 구조와 논리적인 구조를 분리하지 않고 있는 HTML 마크업 특성은 이러한 요구를 만족시킬 수 없다.
- HTML에서 지원되는 단일 링크형은 실제로 페이지들 사이에서 발생하는 다양한 관계를 지원하지 못한다.
- 웹 페이지는 끊임없이 이동하고 있으며, 웹 이용자들이 변화하는 URL을 찾아 다녀야 한다는 문제점이 있다.
- 수학 공식이나 특정 문자에 대한 지원이 부족하며, 이러한 문자를 지원한다고 해도 플랫폼이 변경되는 경우에는 문제를 일으킬 수 있다.
- HTML은 정보를 재사용하기가 어렵다. 동일한 데이터를 웹에서 발행하기 위해서는 카탈로그 형태로 제작하고 이를 데이터베이스의 내부에 유지해야 할 뿐만 아니라 변환을 해주거나 경우에 따라서는 직접 재포맷을 해주어야 한다.

(2) SGML의 한계점

- SGML은 어렵고 복잡하기 때문에 어플리케이션 개발이 용이하지 않다. 현재까지 SGML을 완벽히 지원하는 브라우저가 없는 상황이므로 SGML문서를 HTML문서로 변환시켜 HTML브라우저를 통해 웹 상에 보여준다. 그러나 100% 변환시켜 주는 프로그램은 없다.
- 웹 상에서 SGML의 사용이 보급되지 못하는 또 다른 이유는 SGML이 단지 논리 구조에 대한 표준 마크업 언어이므로 포맷과 관련된 어떠한 정보도 지원하지 않기 때문이다. 이러한 문제점을 해결하기 위해 DSSSL(Document Style Semantics and Specification Language)과 같은 스타일시트를 개발하여 사용하려고 시도하였지만, 현재까지 이를 지원하는 어플리케이션이 없다.
- SGML은 웹 상에서 사용되고 있는 하이퍼링크 기능 및 웹 상에서의 사용을 전혀 고려하지 않고 개발되었다.

따라서 현 시점에서 SGML을 그대로 웹 상에 적용하기는 힘들 것으로 판단하고 새로운 대안을 마련키로 합의하였다. 그 결과 Sun Microsystem의 Jon Bosak을 의장으로 한 W3C(World Wide Web Consortium)를 발족하게 되었고, HTML과 SGML의 단점은 제거하고, 둘의 장점만을 모아 놓은 새로운 표준안인 XML을 제안하였다. XML은 1996년 W3C의 후원으로 형성된 XML Working Group에 의해 개발되었으며, 1998년 2월 10일 W3C의 권고에 따르는 XML 1.0 사양이 발표되면서 표준으로 채택되었다.

XML의 정의는 한 마디로 내릴 수 없지만 W3C의 의장인 Jon Bosak은 XML을 다음과 같이 설명하고 있다. “향후 XML은 웹 기술상에 있어서 가장 핵심적인 진보를 가져 올 것이며, 웹의 근본을 송두리째 바꿀 것이다. XML은 안전한 전자상거래 구축을 가능하게 하고, 새로운 분산 어플리케이션 시대를 이끌어 나갈 것이다. 또한 XML은 소프트웨어 개발자와 고객의 관계를 새롭게 변화시킬 것이다. 다시 말해서 XML은 어떤 플랫폼에서나 읽을 수 있는 포맷을 제공하기 때문에 특정 회사의 제품과 관련된

특정 환경에 읽매이지 않아도 된다.” (Bosak, 1997)

2. 2 XML의 특징

(1) XML 문서의 구조

[표 1]은 XML 문서는 그 논리적 구조상 프롤로그와 문서 엘리먼트로 구성되고, 프롤로그는 XML 선언과 문서 형식 선언으로 이루어진다. XML 선언부에는 XML의 버전과 인코딩을 기술한다. 그리고 문서 형식 선언부에서 DTD(Document Type Definition)를 선언하는데, 선언부 내에 직접 DTD를 기술할 수도 있고, 외부의 DTD를 참조하도록 선언하고 별도로 그 문서에 한정된 DTD를 따로 선언하여 사용한다. 이는 모든 문서의 큰 형식을 유지하면서 각 문서의 고유한 형식을 개별적으로 설정할 수 있다. XML 문서의 논리적 구조와 물리적 구조는 DTD를 사용하여 정의할 수 있도록 한다.

XML 문서	문서 구조
<?XML version=1.0 encoding=UTF-8?>	선언부
<!DOCTYPE mydoc [<!ELEMENT mydoc (header, body)> <!ELEMENT header (#PCDATA)> <!ELEMENT body (chapter*)> <!ELEMENT chapter (#PCDATA)>]>	DTD (내부참조)
<!DOCTYPE mydoc SYSTEM doctype.dtd >	(외부참조)
<mydoc> <header>Xml의 소개</header> <body> <chapter>1. Xml의 개요</chapter> </body> </mydoc>	

[표 1] XML 문서의 구조

(2) XML의 장점

① 인터넷 환경 고려

XML은 웹 상에 정보를 전달하기 위하여 제안된 메타언어이다. 따라서 XML 인스턴스(Instance) 작성시 필요하다면 URL을 자유로이 사용할 수 있으며, 또한 XML을 위한 소프트웨어는 이러한 URL을 마치 HTML의 링크처럼 접근할 수 있는 기능이 내재되어 있다는 것을 의미한다.

② Well-formed 문서 가능

일반적으로 XML을 보다 효율적으로 사용하기 위해서는 DTD를 작성하고 DTD에 의하여 인스턴스를 생성하여야 한다. 그러나 이와 같은 일련의 과정은 구조화 문서에 대한 어느 정도의 이해가 필요하게 되어 초보자들은 문서 작성에 어려움을 겪게 된다. 따라서 사용자 중에 필요에 의하여 DTD를 만들 수 있으며, 필요치 않으면 만들지 않게 함으로써 유연하게 문서 작성을 할 수 있게 한다.

③ 스타일 시트

XML의 스타일시트인 XSL은 XML의 인스턴스에 사용자가 정의하는 문서 스타일을 적용하는 메커니즘을 제공한다. XSL은 DSSSL를 기반으로 하고 있으며 CSS와 상호 운영이 가능하다. 또한 CSS가 단지 문서의 디스플레이에 대한 기능을 제공하고 있는 반면에 XSL은 문서의 프린팅을 위한 기능도 제공한다.

④ 링크 및 포인트

XML의 표준 셋은 HTML의 링크를 확장한 XLink와 문서의 전체 또는 일부분을 지정하기 위한 XPointer를 포함한다. 따라서 XML을 지원할 때에는 이러한 XLink와 XPointer를 모두 지원하는 것을 의미한다.

⑤ 메타 데이터 관리

메타 데이터란 일종의 카탈로그 정보로서 문서 이외의 정보들(문서의 저자, 작성일시, 최종 수정일 등)을 일컫는 말이다. 현재 XML은 이러한 메타 데이터 관리를 위한 표준안(Resource Definition Framework : RDF)이 동시에 개발되고 있으며 SGML에서와 달리 RDF를 이용한 다양한 응용들이 개발되고 있다.

⑥ 표준 API

XML은 SGML과는 달리 인스턴스와 DTD에 대한 표준 인터페이스를 지원한다. 표준 인터페이스인 DOM(Document Object Model)은 W3C에서 XML 문서와 HTML 문서에 대한 API(Application Programming Interface)를 제정한 것으로 DOM을 지원하는 모든 프로그램들은 XML문서에 대한 접근, 추출, 저장이 가능하게 되며 문서를 문서 자체가 아닌 객체들로 나누어 처리하게 된다. 현재 DOM의 표준안 개발은 활발히 진행되고 있으며, 업계에서도 DOM을 지원하는 소프트웨어(파서, 문서 관리 시스템)들이 지속적으로 출시되고 있다.

⑦ 사용 가능한 소프트웨어

현재 XML은 표준의 발표와 함께 파서가 개발되었으며 현재까지 다양한 소프트웨어가 개발되고 있다. 물론 SGML도 전용 소프트웨어가 다양하지만, XML의 소프트웨어와는 다음과 같은 차이가 있다.

첫째, SGML의 경우는 전용 소프트웨어가 필요에 의해 개별적으로 개발되는 상황이다. 같은 SGML 기반이라고 하더라도 이들은 API나 데이터 교환에 있어 통일성을 제공하지 못한다. 반면 XML은 웹을 기반으로 하여 웹 브라우저만 있으면 어느 플랫폼에서나 똑같은 출력물을 얻을 수 있어서 데이터의 표준성을 제공한다(Bosak, 1997). 둘째, 마크업 언어의 소프트웨어 중에서는 가장 중요한 소프트웨어 컴포넌트는 파서이다. 모든 소프트웨어의 개발에 파서는 필수적인 중요한 요소이며, 이 파서의 역할에 따라 응용 프로그램이 제대로 기능을 나타낼 수 있는지가 결정된다. 물론 파서

의 역할이 어디까지냐에 따라 다르겠지만, 일반적으로 파서란 문서를 객체화시켜 메모리에 트리를 구성하고, 그 트리에 대한 다양한 API까지 제공함으로써 사용자는 그 API만을 이용하여 프로그래밍할 수 있도록 되어있는 것이다. SGML의 경우에는 범용으로 자유로이 사용 가능한 파서가 존재하지 않으며, XML의 경우에는 소스 코드까지 공개하는 파서가 일반적이다(Simon, 1998).

이러한 XML의 특징을 바탕으로 SGML, HTML의 특징과의 차이점을 하나의 표로 정리하면 [표 2]와 같다.

비교항목	HTML	SGML	XML
태그사용	사용자 정의 불가능 제한적	사용자 정의 가능 무제한	사용자 정의 가능 SGML보다 제한적
문서의 재사용	불가능	가능	가능
응용분야	홈페이지 작성용 간단한 구조의 문서	기술문서용, 전자도서관용 대용량의 복잡한 구조의 문서	SGML과 동일 웹상의 전자상거래용문서
문서작성	간단하고 용이함	매우 복잡함	SGML보다 편리함
문서검색	효과적 검색 어려움	보다 정확한 검색가능 문서구조에 대한 검색가능	SGML과 동일
링크	HTML(단순링크)	HyTime	XLL
구축 비용	낮다	높다	낮다
구축 소요시간	적다	많다	적다
구축 용이성	쉽다	어렵다	쉽다
출력형식	CSS	DSSSL	XSL

[표 2] HTML/SGML/XML의 비교분석

2. 3 XML 관련 표준

현재 XML은 표준으로서 확립된 상태는 아니지만 W3C는 XML을 표준으로 채택하기 위한 권고안과 초안들을 내놓은 상태이다. XML과 관련된 핵심 표준안들은 다음과 같고 이 자료는 W3C의 권고안으로부터 발췌한 것이다.

(1) XML

XML은 1996년 W3C(World Wide Web Consortium)의 원조하에 만들어진 XML 워킹 그룹(XML Working Group)이 개발한 것이다. XML 워킹 그룹은 Sun Microsystems사의 Jon Bosak이 주도했으며, 역시 W3C가 조직한 XML Special Interest Group이 적극적으로 참가했다.

XML의 설계 목표는 SGML과 호환되는 인터넷상에서 수월하게 사용될 수 있는 문서를 만드는 것이었고, 다양한 어플리케이션들을 지원하기 위함이었다. 또한 작성하기 쉽고, 논리적으로 명확한 문서를 만들기 위해 설계되었다. 뿐만 아니라 이질적인 시스템과 다른 기종의 시스템간에도 원활한 데이터의 교환을 할 수 있도록 설계하였다. XML은 데이터의 구조와 내용까지 담고 있는 문서이다. XML은 웹 브라우저만 있으면 어떤 플랫폼과 어떤 기종에서나 똑같은 출력물을 주고받을 수 있도록 설계함으로써 데이터 교환 독립성(Data Interchange Independence)을 실현하였다.

XML은 단지 웹에서만 사용 가능한 자료 형태가 아니고 프리젠테이션 로직과 프로세싱 로직이 분리되어 있기 때문에 웹 이외의 일반적인 응용 프로그램간의 자료 교환에도 충분히 적용할 수 있으며 그 적용분야는 무궁무진하다. 일반적인 웹 환경에서는 XSL과 같은 변환 툴만 있다면 특정 응용프로그램의 용도에 맞는 형태로 표현하는 것이 가능하다. W3C가 이 권고안을 만든 목적은 XML 사양에 대한 관심을 끌고 XML 사양이 널리 적용되도록 하는 것이다. W3C의 그러한 노력은 웹의 기능과 상호 운용성

을 증진시킬 것이다.

(2) DOM

DOM(Document Object Model)은 XML 문서를 처리할 때 사용하는 공통적인 단어집을 XML 응용 프로그램 개발자들에게 제공한다. 응용 프로그램이 C로 쓰여졌든, 아니면 델파이나 자바, 펄, 심지어 비주얼 베이직으로 쓰여졌든 DOM을 따른다면 동일한 메소드를 사용하게 된다. 즉, 내부에서 벌어지는 일은 서로 상당히 다르다고 해도 어떠한 작업을 수행하기 위한 명령 자체는 모두 동일하다.

그리고 클라이언트 쪽에서는 브라우저가 넷스케이프이전 익스플로러이전 문제가 되지 않는다. 그 브라우저들이 모두 동일한 DOM을 구현하고 있는 한 하나의 스크립트는 모두 동일한 방식으로 브라우저에서 실행한다. 즉, DOM은 플랫폼과 프로그래밍 언어에 중립적인 인터페이스를 제공하고, XML을 이용한 개발을 보다 쉽게 해 줄 수 있다.

또한 서로 다른 플랫폼과 언어를 이용하여 작성한 XML 문서의 구조와 내용을 쉽게 이해할 수 있게 해줌으로써 XML의 문서를 재사용하는 것을 돕는다.

(3) XML Namespace

이름공간(Namespace)은 URI(Uniform Resources Identifier)로 식별되는 이름들의 집합이며, 이름이란 XML 문서의 엘리먼트나 어트리뷰트 이름으로 쓰이는 식별자들을 말한다. 하나의 URI는 고유한 값이므로 이름공간 역시 고유하며, 결과적으로 이름공간의 한 이름 역시 고유하다. 이름공간의 사용은 같은 이름의 태그가 있을 때 이를 구별하기 위해서 사용하며, 이름이 정의된 곳의 URI를 XML 문서에 선언함으로써 이용할 수 있다.

(4) XSL

XSL(eXtension Stylesheet Language)은 XML 문서의 글꼴, 크기, 색, 배치 등의 외양을 지정하는 기능을 제공할 뿐만 아니라, CSS같은 기존 스타일시트에는 없는 확장성, 즉 사용자가 직접 자신만의 포매팅 태그와 특성들을 만들어서 사용할 수 있는 능력을 제공한다. 또한 XSL은 문서의 표현 방식을 좀 더 정교하게 제어할 수 있도록 해준다. 예를 들어 요소들의 표시 순서를 지정하는 것도 가능하기 때문에 문서 자체에 담긴 데이터 순서는 중요하지 않다.

(5) XLink(XLL)

XLink(XML Linking Language; XLL)는 XML 문서의 연결을 위해 사용되는 언어이다. XLL은 XLink와 XPointer로 이루어진다. XLink는 서로 개별적인 두 XML 문서 사이의 링크 연결에 관련된 것이다. XML 문서는 XLink를 이용하여 다양한 Link기능을 수행할 수 있다. XPointer는 URI와 함께 한 문서의 특정 부분을 지정하는데 사용된다. 즉, 문서 전체가 아니라 문서의 특정한 부분만을 지정할 수가 있다. 문서의 부분을 언급하는 링크는 모두 XPointer의 형태를 가져야 한다. 그러나 HTML에서와는 달리 XPointer 언어에서는 명시적인 ID특성을 포함시키지 않고도 XML 문서의 특정 요소나 문자열, 기타 여러 부분들에 대한 참조를 기술할 수 있다. 이러한 XLink를 이용하여 XML 문서는 일대 다, 다대 일 관계를 가지는 양방향 링크를 만들 수 있다. 뿐만 아니라 링크 요소를 직접 만들 수 있으며, 사용자가 링크들 사이를 돌아다니는 방식을 지정할 수 있으며 Link를 그룹화할 수도 있다. 또한 XLink 문서를 외부에 두면 XML 문서들이 지정한 링크가 깨졌을 때 XLink 문서만 수정해주면 되는 편리한 점도 있다.

제 III 장 전자카탈로그의 표준화와 선행연구

3. 1 전자카탈로그 개념 및 표준화

최근 들어 온라인 쇼핑이 인터넷의 주요 상업적 응용영역이 됨에 따라 이들 사이트들의 사용성(Usability)도 심각한 문제가 되고 있다. 조사에 따라 차이는 있지만 사용자의 1/2~2/3 정도가 자신들이 찾고자 하는 것을 웹 상에서 찾지 못한다고 한다. 따라서 이러한 문제를 해결하기 위해 인간·컴퓨터 인터페이스(Human Computer Interface : HCI)에 대한 기법, 도구 및 방법이 새로운 상호작용 형태를 설계, 구현, 평가하기 위해 사용되고 있으며, 전자카탈로그 역시 흥미 있는 HCI 영역중 하나이다(Koenemann & Stolze, 1999).

전자카탈로그(Electronic Catalogues)란 “상품에 대한 정보를 담은 전자 문서”로 정의할 수 있다(전자상거래 표준화 통합 포럼, 2001). 여기서 상품이란 상거래의 대상이 되는 유무형의 재화와 서비스를 뜻하며, 정보란 상품자체에 대한 묘사와 상거래를 위해 필요한 부가적인 데이터, 또는 다른 문서들과의 관련성 등을 포괄적으로 포함한다. 전자문서란 컴퓨터에 저장되는 데이터의 묶음으로서 특정 단위를 명시하지 않는 추상적 개념이다.

전자카탈로그는 상품에 대한 간략한 소개, 사진(동영상), 제작업체 URL, 연락처 등 기존의 인쇄물 형태의 카탈로그에 비하여 많은 내용으로 구성된다. 인터넷상에서의 구현은 정보검색 편의성, 효율성 등을 위하여 많은 상품광고를 Mall 형태로 구축하는 것이 바람직하며 그 만큼 광고의 효과를 높일 수 있다.

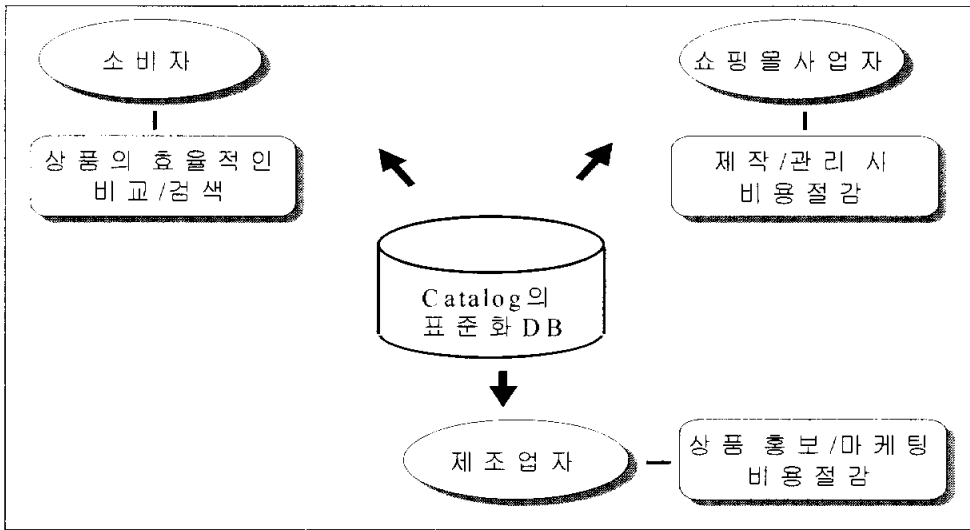
전자카탈로그 관련 주체로는 공급자, 구매자, 그리고 마켓플레이스가 있다. 공급자는 상품이나 서비스를 가진 주체로서, 전자카탈로그를 정의 및 구현해야 한다. 구매자는 마켓플레이스를 통해 상품이나 서비스에 대한

전자카탈로그를 검색해서 거래할 공급자를 찾는 주체이다. 그리고, 마켓플레이스는 수많은 공급자들의 비즈니스 정보와 전자카탈로그 정보 등을 정보저장소에 등록해 놓은 주체이다.

전자카탈로그의 특징은 종이 등의 인쇄물 형태의 카탈로그보다 저렴한 비용으로 제작할 수 있으며 판매자가 직접 수정/편집이 가능한 DIY(Do It Yourself) 방식이 가능하다. 또한 시간적, 공간적 제약을 받지 않고 세계적으로 BBS, WWW 등을 통한 홍보 및 구매가 가능하다. 즉, 전자카탈로그는 인터넷을 통한 상거래가 즉시 이루어질 수 있게 하며, 상품 또는 기업에 대한 광고를 홍보하기 위한 가장 효과적인 방법이다(홍언주, 1999).

현재 많은 쇼핑몰 관련 업체들이 전자카탈로그를 구축하고 있으나, 전자카탈로그에 대한 명확한 표준이 없어 서로 상이한 내용, 포맷, 교환방법 등으로 구축하고 있다. 이렇게 표준이 없는 상태에서 구축된 전자카탈로그는 상품을 제조, 판매하는 업체에서 동일 상품에 대해 각기 다른 속성과 포맷으로 정의하여 중복 구축함으로써 연속된 정보의 흐름을 어렵게 하고, 수작업에 따른 부대비용이 많이 들고, 상품정보의 교환 및 검색 등에서 전자카탈로그의 상호 호환 및 연계가 곤란한 문제를 가지고 있다. 이러한 문제점을 해결하기 위해서는 전자카탈로그가 표준화되어야 한다.

[그림 1]은 전자카탈로그 표준의 의의로는 1) 소비자 입장에서는 상품의 효율적인 비교·검색이 가능 2) 쇼핑몰 사업자에 있어서는 제작·관리 시 비용 절감 3) 제조업자는 상품홍보 및 마케팅 비용절감 등을 들 수 있다(홍언주, 1999).



[그림 1] 전자카탈로그 표준의 효과

전자카탈로그 관련 표준은 아래의 [표 3]에서와 같이 크게 내용적인 구성요소와 기술적인 구성요소로 나눌 수 있다(전자상거래 표준화 통합 포럼, 2001). 내용적 구성요소는 제품의 내용을 나타내는 정보로서 상품분류 및 식별 표준, 속성 표준으로 다시 구분된다. 기술적 구성요소는 그 내용을 어떠한 형식으로 구축하여 구매자에게 어떻게 보내야 하는지에 대한 기술적 방법을 정의하는 부분으로 전송 표준, 포맷 표준, 표현 표준, 디렉토리서비스 표준 등 네 가지로 구성된다. 전송 표준은 상품정보를 전송하기 위한 프로토콜에 관한 표준으로서 HTTP, X.400, FTP등이 있다. 포맷 표준은 상품정보를 전자문서로 저장하는 방식에 관한 표준으로서 텍스트, HTML, XML, SGML, PDF등의 방식이 존재한다. 표현 표준은 전송된 상품정보를 사용자가 볼 수 있는 형태로 나타내기 위한 스타일시트에 관한 표준으로서 CSS, XSL, DSSSL 등의 방식이 존재한다. 마지막으로 디렉토리서비스 표준은 X.500, LDAP 등이 있다.

구 분	세부 표준	내 용
내용적 구성요소	상품 분류체계	상품 분류체계의 구축
	게시표준	상품 분류체계에 따른 상품 속성정보 정의(Content)
기술적 구성요소	전송표준	통신망상으로 교환하기 위한 프로토콜 - HTTP, X.400, X.435, FTP, SMTP 등 (SCTP(Standard of Catalog Transfer Protocol) 제정)
	포맷표준	상품정보를 담고 있는 문서형태에 관한 표준안 - 텍스트, HTML, XML, SGML, PDF, SMTP 등 (XML(eXtensible Markup Language) 예정)
	표현표준	상품정보를 화면에 출력하기 위한 표준안 - CSS, XSL, DSSSL 등 (XSL(eXtensible Stylesheet Language) 예정)
	디렉토리 서비스표준	카탈로그 및 목록 등에 사용되는 디렉토리 표준 - X.500, LDAP

[표 3] 전자카탈로그 관련 표준

3. 2 전자카탈로그 관련 선행연구

(1) 웹 콘텐츠에 관한 선행연구

W3C의 등장 후, 불과 몇년 만인 1996년 6월 기준으로 230,000개의 웹 사이트가 생겨났으며, 이중 50%가 인터넷 쇼핑몰 사이트이다(Vadapalli & Ramamurthy, 1997~8). 즉 W3C의 등장으로 인해 인터넷의 상업적 이용이 전통적인 인터넷의 비상업적 이용을 추월하게 된 것이다. 이로 인해 인터넷은 여러 산업에 걸쳐 많은 변화를 주고 있으며, 관련 기술 역시 급속도로 변화하고 있다. 이러한 시대적 변화에 따라 많은 기업들이 웹사이트를 개설하기 시작하였다. 그러나 실제 여러 기업들이 웹사이트 개발시 어떤 콘텐츠로 웹사이트를 구성하여야 하는가에 대한 명확한 가이드라인은

없는 실정이다. 특히, 최근에는 웹 상에서 직접적으로 판매하는 전자상거래가 일반화되어 많은 산업에서 정부, 기업, 소비자 등 여러 주체간에 거래가 이루어지고 있다. 그러나 아직까지 웹을 상업적 목적으로 보다 잘 활용하기 위해 웹 콘텐츠를 어떻게 구성하여야 하는가에 대한 연구나 가이드라인은 부족한 실정이다. 이는 아직 쇼핑몰 구축시 필요한 요소 기술에 대한 표준안이 마련되지 못했기 때문이다. 따라서 현재는 쇼핑몰마다 각 기능별 제공 정보 및 디자인의 형태가 서로 달라 쇼핑몰 방문자들이 혼동 혹은 이질감(Unfriendliness)을 겪게 됨에 따라 웹사이트 방문율과 구매율 간의 차이가 많은 실정이다.

최근 들어 이러한 문제를 해결하기 위해 웹 콘텐츠에 대한 몇몇 연구가 시도되고 있다. 이러한 연구들 중 하나는 컴퓨터에 대한 기초지식이 없는 사람도 쉽게 웹에 접속할 수 있도록 하기 위해 W3C(World Wide Web Consortium)에서 제시한 “Web Content Accessibility Guidelines”이며, 이는 일반 홈페이지에 대한 권고안의 성격을 갖는 것이다.

Liu 등(1997)은 Fortune 500대 기업을 대상으로 미국 대기업이 새로운 기술에 대한 반응 방법과 비즈니스를 하기 위해 웹사이트를 사용하는 방법을 알아보는 과정에서 일반 홈페이지를 대상으로 웹사이트의 주요 콘텐츠에 대한 조사를 실시하였다.

또한 Poon & Swatman (1998)은 전자상거래를 하는 소규모업체를 대상으로 인터넷에 대한 기대 및 현실간의 차이를 20개월의 시차를 두고 조사한 추적조사(Longitudinal survey)를 통해 기업별로 웹에 대한 기대 차이로 인해 웹 콘텐츠가 다를 수 있다는 가능성을 제시하였다.

Huizingh(2000)는 북미와 네덜란드의 사이트를 대상으로 웹의 콘텐츠와 디자인이 산업 및 웹사이트 규모에 따라 차이가 존재하는가를 조사하였다. 조사 결과에 의하면, 웹사이트 규모가 산업 기준보다는 콘텐츠와 디자인에 대한 중요한 서술자임이 밝혀졌다.

Wan(2000)은 특정 비즈니스의 성공을 위한 웹 특성과 비즈니스 웹사이트의 개발 방향간의 이론적 관계를 표시하는 격자 구조의 프레임워크를 제시하였다. Liu & Arnett(2000)은 성공적인 전자상거래를 지원하기 위해

웹사이트가 갖추어야 될 요인을 마케팅과 정보시스템 영역에서 조사해서 이를 토대로 Fortune 1,000대 기업의 웹마스터에게 설문조사를 실시한 결과, 정보의 질(Information Quality), 학습 능력(Learning Capability), 유희성(Playfulness), 시스템의 질(System Quality), 시스템 사용(System Use), 서비스의 질(Service Quality) 등의 6가지 구성 개념이 웹 디자인시 중요하다고 하였다.

(2) 상품 카탈로그에 관한 선행연구

전자카탈로그는 상거래와 관련된 정보를 제공하는 수단으로서 오프라인 상에서의 상품 전시 및 점원의 설명기능이나, 통신판매에서의 종이카탈로그(Paper Catalog)에 해당하는 것이다. 먼저 전자카탈로그의 특성들을 알아보기 전에 전통적인 종이카탈로그에 대한 연구를 살펴보면, 종이카탈로그는 전통적인 직접 마케팅(Direct Marketing)의 수단으로서 많은 사람들에게 의해 연구가 되었으나, 이들 연구들은 주로 페이지 배치(Layout) 문제에 집중되어 있다(Lewis, 1992 ; Schmid, 1993, 1996 ; Hayes, 1993 ; ShashoJones, 1991). 따라서 주로 판매카탈로그 상에서 판매상품의 배치 요인만을 강조하고 있다.

전자카탈로그에 대한 선행연구 중에서 먼저 Spiller & Lohse(1997~8)의 연구를 살펴보면, 이들은 전자 상점의 분류를 위해 의류업을 대상으로 온라인 상점의 속성을 조사하였다. 이들은 전자카탈로그가 전자 상점을 위해 전자적인 형태로 된 상품속성 정보의 전달 문서라면, 종이카탈로그는 기존의 오프라인 거래를 지원하기 위한 종이 형태로 된 상품속성 정보의 전달문서라는 전제를 하고, Hoffman 등(1996)의 주장에 따라 소매점, 종이카탈로그, 전자카탈로그간의 유추를 통해 전자카탈로그의 특성 조사 변수들을 정의하였다. Hoffman 등은 온라인 소매상점을 전자카탈로그를 이용해서 전자적 경로를 통해 직접 판매를 하는 곳이라 보고 물리적 상점, 종이카탈로그, 그리고 온라인 상점의 차이를 제품의 다양성과 유형, 가격, 광고, 판촉노력, 서비스, 카탈로그 및 상점의 디자인과 배치 등의 측면에서

유추(Analogies)를 통해 제시하였다. 그리고 Keller(1997)는 여러 기업들이 상호작용이 가능하도록 하는 전자카탈로그의 아키텍처에 대한 연구를 통해, 기업은 검색 및 기계관독용(Machine Sensible)스마트 카탈로그를 만들고, 소매상이나 유통업자는 고객에게 제품정보를 제공하기 위해 가상카탈로그(Virtual Catalog)를 만드는 방법을 제안하였다.

내 용
통신판매 사업자의 상호 · 주소 · 전화번호
상품의 종류 또는 용역의 내용
상품의 판매 가격 또는 용역의 대가
상품 대금 또는 용역 대가의 지급 시기 및 방법
상품의 인도 시기 또는 용역의 제공 시기
상품의 제조원
청약의 기간 또는 기한
상품의 판매 가격 또는 용역의 대가에 송료가 포함되지 않을 경우에는 그 송료 부담에 관한 내용
상품의 판매 가격 또는 용역의 대가 외에 소비자가 추가로 부담하여야 할 비용이 있는 경우에는 그 내용 및 금액

[표 4] 통신판매업에 대해 법률이 지정한 정보표시항목

국내에서 행한 전자카탈로그에 관한 선행연구로는 한국소비자보호원(1998)에서 행한 OECD의 가이드라인 및 현행 방문판매 등에 관한 법률의 관련 규정에 비추어 107개 인터넷 쇼핑몰(종합쇼핑몰 21개, 전문쇼핑몰 86개)의 정보표시 실태를 조사한 결과가 있다. 조사결과에 의하면 전반적으로 쇼핑몰이 소비자에게 제공해야 할 정보의 표시 실태가 미흡한 것으로 밝혀졌으며, 특히 상품의 사양 및 배달방법이 많이 미흡한 것으로 분석되었다. 그리고 문희철(2001)은 전자상거래 표준 동향에 관한 연구에서 전자카탈로그의 속성 정보를 제품관련 정보, 거래관련 정보, 배송관련 정보, 업체관련 정보, 소비자관련 정보 등으로 세분화해서 제시하였다. 홍언주(1999)에 의하면 “방문판매 등에 관한 법률”의 제18조, 시행령 제8조, 시

행규칙 제12조에 의해 통신판매업자는 상품의 판매 또는 용역의 제공에 관하여 표시, 광고를 할 때 위의 [표 4]와 같은 사항을 명시해야 한다고 한다.

제 IV 장 표준으로 지정된 전자카탈로그 비교

4. 1 cXML(commerce XML)

cXML은 Ariba를 주축으로 한 40개 이상의 전자상거래 분야의 선도적 업체들이 공동으로 개발한 것으로, 인터넷에서 공급자와 구매자간에 컨텐츠와 거래 자료를 주고받기 위한 XML기반 표준 인터넷 언어이다.

cXML은 XML 기반의 언어로서 구매자와 공급자 사이의 전자상거래 시 필요한 트랜잭션들을 위한 DTD의 집합이다. cXML 문서는 크게 공급자의 서비스와 물품들의 목록을 구성하는 카탈로그와 공급자의 웹사이트에서 상호간에 확인할 수 있는 Punchout 그리고 주문 관련 문서로 나눌 수 있다. 이러한 구매자 쪽의 구매 어플리케이션은 Ariba net을 통해 공급자들의 카탈로그를 검색하여 자신의 데이터베이스에 저장해서 필요한 물품을 찾아내거나 구매하거나 공급자 쪽의 Punchout 문서를 통해 직접 검색하거나 구매할 수 있다.

이것은 일반거래에서 쓰이는 여러 트랜잭션들의 폼을 표준화시킨 것이다. 상품의 분류면에서 본다면, cXML을 통해서 구매자는 상품들 사이의 일련의 계층관계까지는 알 수 없다. 공급자가 가지고 있는 상품의 아이템과 세부적인 정보까지만 알아차릴 수 있도록 표준을 정해놓은 것이다.

cXML은 전자상거래부분에서 구매자 관리, 공급자 관리, 컨텐츠 관리 서비스, 전자 시장 및 웹 기반 자원구입 조직을 포함한 모든 공급자의 컨텐츠와 상품목록 모델을 지원한다. 또한 구매요청서, 교환 인증서, 인증, 상태 갱신, 선적 고지, 지불 및 기타 거래 등 거래 관련 정보의 교환에 관한 요청 및 응답절차를 정의하고 있으며 보안 기능도 제공하고 있다.

cXML은 요청/응답(Request/Response) 모델에 기반하고 있으며 헤더(Header), 메시지(Message), 요청/응답(Request/Response) 메시지의 구조

를 갖는다.

4. 1. 1 전자카탈로그 DTD

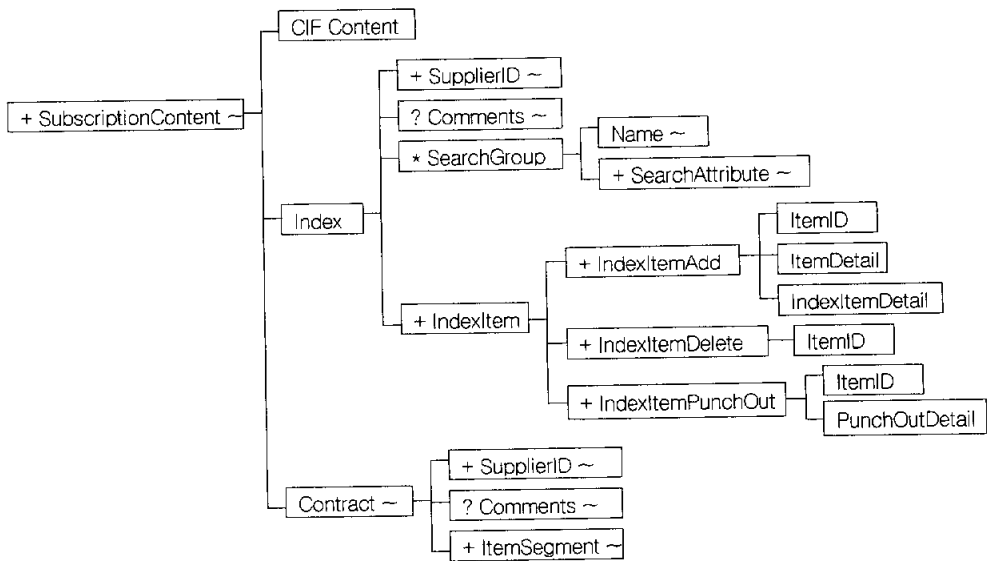
cXML의 카탈로그는 Supplier, Index, Contract 엘리먼트로 구성되어 있다.

Supplier는 주소, 연락 방법, 주문 정보 등 공급자의 기초적인 정보를 포함한다.

Index 엘리먼트는 설명, 파트 번호, 분류코드와 같은 상품과 서비스의 공급자 목록에 대한 데이터를 말한다. Index 엘리먼트는 SupplierID, SearchGroup, IndexItem으로 구성되고, IndexItem 엘리먼트는 IndexItemAdd, IndexItemDelete, IndexItemPunchout을 가질 수 있다.

IndexItemAdd 엘리먼트는 인덱스기존의 상품정보를 갱신하거나 새로운 상품정보를 추가한다. 이때 정보는 ItemID, ItemDetail, IndexItemDetail 엘리먼트를 포함한다. IndexItemDelete 엘리먼트는 인덱스에서 상품을 제거한다. 이 때는 유일한 ItemID 엘리먼트를 포함한다. IndexItemPunchout은 공급자의 웹사이트로 Punchout 세션을 초기화 할 상품을 추가한다. 이것은 ItemID와 PunchoutDetail 엘리먼트를 포함한다. 이는 구매자가 공급자의 웹사이트로부터 실시간으로 상품의 상세 정보를 획득할 수 있다는 것을 의미한다.

Contract 엘리먼트는 공급자와 구매자 사이의 유연한 협상을 지원하기 위한 정보를 포함한다. Contract는 OVERLAY 어트리뷰트를 이용하여 상품의 가격을 협상할 수 있도록 한다. 다음의 [그림 2]는 cXML DTD 중에서 카탈로그에 해당하는 부분을 확대하여 보여준다.



[그림 2] cXML 카탈로그 DTD¹⁾

상품정보를 담은 Index 엘리먼트를 자세히 살펴보겠다. IndexItemAdd는 ItemID, ItemDetail, IndexDtail의 세가지 엘리먼트로 구성되어 있다. 다음의 [그림 3]은 IndexItemAdd가 포함하는 자식 엘리먼트를 보여준다.

ItemID는 원격 웹사이트에서 상품을 가리키는 유일한 키이며, SupplierPartID와 SupplierPartAuxiliaryID로 구성되어 있다. SupplierPart-ID는 상품을 나타내는 유일한 ID일수 있지만, 예외 경우로 같은 상품이지만 “날개”, “박스”와 같이 단위에 따라 다른 가격을 가지는 상품에 대해서는 SupplierPartAuxiliaryID를 부가함으로써 유일한 ID가 될 수 있다.

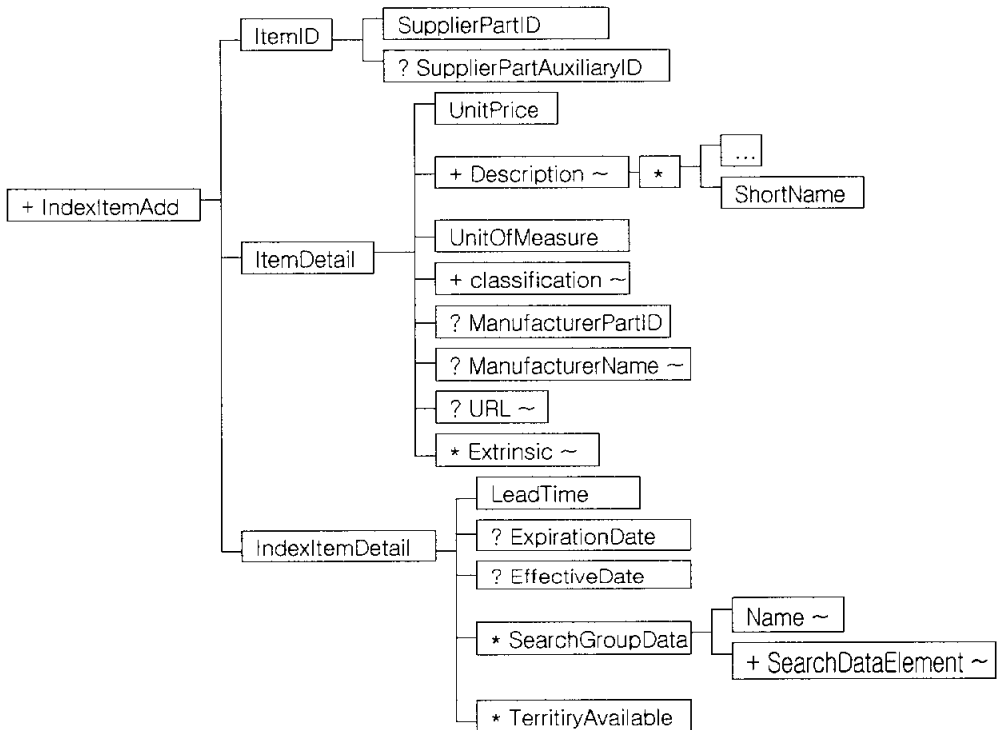
ItemDetail은 상품의 상세 정보를 가지며, 꼭 포함해야 하는 UnitPrice와 UnitOfMeasure, 부가적인 정보인 ManufacturePartID, Manufacture

1) 기호 설명

- ? : 해당 엘리먼트를 0 또는 1번 포함할 수 있음.
- + : 해당 엘리먼트를 1번 이상 꼭 포함하여야 함.
- * : 해당 엘리먼트를 0 또는 1번 이상 포함할 수 있음.
- ~ : 해당 엘리먼트에 어트리뷰트가 정의되어 있음.

-Name, Description, URL, Classification, Extrinsic 엘리먼트를 포함한다. UnitPrice 엘리먼트는 단위별 가격, UnitOfMeasure 엘리먼트는 파운드 또는 그램 등과 같은 측량 단위를 나타낸다.

IndexItemDetail은 상품의 추가 정보를 정의하기 위하여 필수 엘리먼트인 LeadTime를 가지고 그 외에 ExpirationDate, EffectiveDate, SearchGroupData, TerritoryAvailable 엘리먼트들을 포함할 수 있다. 다음의 [표 5]는 위에서 설명한 IndexItemAdd 엘리먼트 내에서 DressShoe라는 이름의 상품의 속성을 표현하는 예를 보여준다.



[그림 3] IndexItemAdd 엘리먼트

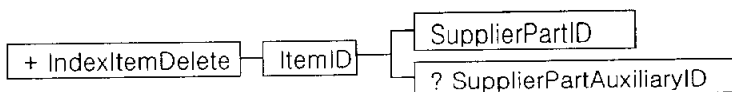
```

<IndexItemAdd>
  <ItemID><SupplierPartID>pn12345</SupplierPartID></ItemID>
  <ItemDetail>
    <UnitPrice><Money currency= "USD" >60</Money></UnitPrice>
    <Description xml:lang= "en" >Men's black shoes</Description>
    <UnitOfMeasure>EA</UnitOfMeasure>
    <Classification domain= "SPSC" >5136030000</Classification>
    <ManufacturerPartID>MBS3.12</ManufacturerPartID>
    <ManufacturerName>Florsheim</ManufacturerName>
    <URL>http://www.florsheim.com</URL>
    <Extrinsic name= "ManufacturerURL" >http://www.shoemaker.com
    </Extrinsic>
  </ItemDetail>
  <IndexItemDetail>
    <LeadTime>10</LeadTime>
    <ExpirationDate>2000-06-01</ExpirationDate>
    <EffectiveDate>1999-01-01</EffectiveDate>
    <SearchGroupData>
      <Name xml:lang= "en" >DressShoe</Name>
      <SearchDataElement name= "Size" value= "12" />
      <SearchDataElement name= "Color" value= "Black" />
      <SearchDataElement name= "Gender" value= "Men's" />
    </SearchGroupData>
    <TerritoryAvailable>USA</TerritoryAvailable>
  </IndexItemDetail>
</IndexItemAdd>

```

[표 5] IndexItemAdd 엘리먼트 예

IndexItemDelete는 상품정보를 삭제하는 엘리먼트이며, ItemID를 포함한다. ItemID는 IndexItemAdd 엘리먼트에서 설명한 ItemID와 동일하다. 아래의 [그림 4]는 IndexItemDelete 엘리먼트의 자식 엘리먼트를 나타낸다.

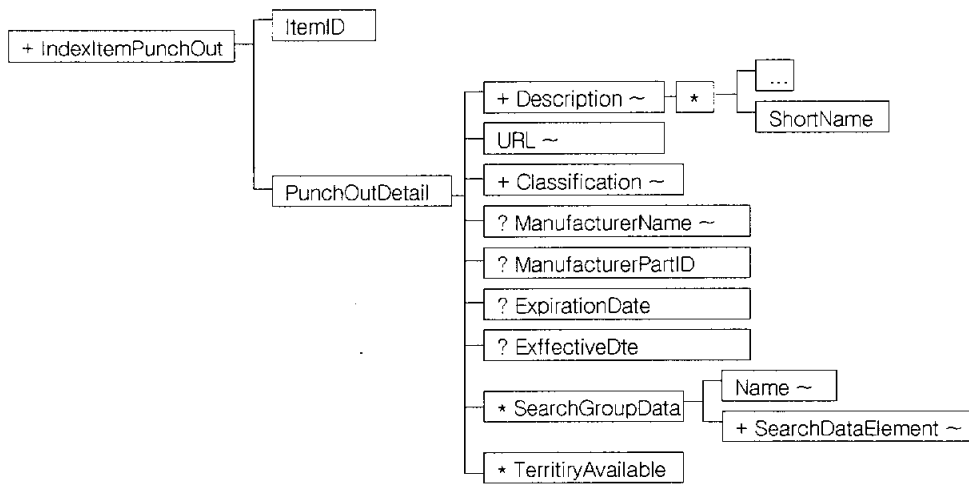


[그림 4] IndexItemDelete 엘리먼트

IndexItemPunchout로 지정한 상품에 관심이 있는 구매자는 공급자의 웹사이트에서 실시간 카탈로그 정보를 조회하는 Punchout 방식을 통해 가격을 비롯한 상세 내역을 얻는다. PunchoutDetail과 ItemID 엘리먼트로 구성되는데,

이것은 가격 정보가 필요하지 않은 것만 제외하면 IndexItemAdd 엘리먼트와 유사하다. PunchoutDetail은 하나 이상의 Description과 Classification이 필요하다는 것을 제외하면 ItemDetail과 유사하다. PunchoutDetail은 가격, lead time 또는 측량 단위 정보를 포함하지 않는다. URL은 꼭 포함해야 하고, 그 외에 ManufacturerName, ManufacturerPartID, ExpirationDate, EffectiveDate, SearchGroupData, TerritoryAvailable, Extrinsic 엘리먼트를 포함할 수도 있다.

ItemID 엘리먼트는 IndexItemAdd에서 설명한 것과 같다. 아래의 [그림 5]는 IndexItemPunchout 엘리먼트의 자식 엘리먼트를 보여준다.



[그림 5] IndexItemPunchout 엘리먼트

다음의 [표 6] IndexItemPunchout 엘리먼트 예는 Punchout 대상인 상품 pn12399의 상품 설명, URL, 제조업자 명, 판매 구역, 검색을 위한 데이터로 크기, 칼라, 성별 등을 명시하고 있다.

```

<IndexItemPunchout>
  <ItemID><SupplierPartID>pn12399</SupplierPartID></ItemID>
  <PunchoutDetail>
    <Description xml:lang= "en-US" >Ruby slippers</Description>
    <URL>http://oz.com/Dorothy/shoes/red.htm</URL>
    <Classification domain= "SPSC" >5136030000</Classification>
    <ManufacturerName>Wizard Shoes</ManufacturerName>
    <ManufacturerPartID>WSRS1</ManufacturerPartID>
    <ExpirationDate>2010-01-01</ExpirationDate>
    <EffectiveDate>1999-11-24</EffectiveDate>
    <SearchGroupData>
      <Name xml:lang= "en" >DressShoe</Name>
      <SearchDataElement name= "Size" value= "4" />
      <SearchDataElement name= "Color" value= "Ruby Red" />
      <SearchDataElement name= "Gender" value= "Women's" />
    </SearchGroupData>
    <TerritoryAvailable>US</TerritoryAvailable>
    <TerritoryAvailable>UK</TerritoryAvailable>
  </PunchoutDetail>
</IndexItemPunchout>

```

[표 6] IndexItemPunchout 엘리먼트 예

Index 엘리먼트 내의 SearchGroup 엘리먼트는 상품의 검색을 위해 사용된다. Index 엘리먼트 내 앞에 오는 SearchGroup은 검색 항목들을 정의한다. 정의된 검색 항목들에 대한 개별 상품의 속성 값은 그 상품의 상세 정보를 담은 IndexItemDetail 엘리먼트 내에서 SearchData 엘리먼트로 부여한다. Search-Data 엘리먼트의 자식 엘리먼트인 SearchDataElement가 개별적인 속성 값을 명시하여 이 상품이 검색될 수 있도록 한다.

상품이 소속된 분류 카테고리에 해당하는 Classification 엘리먼트는 Item-Detail 엘리먼트 내에 포함된다. 이때 Classification은 Domain 어트리뷰트에 "SPSC" 와 같은 표준 분류체계를 명시하고 그 코드 값을 시작 태그와 끝 태그 사이에 쓴다.

곧 이은 [표 7], [표 8], [표 9]은 카탈로그를 구성하는 Supplier, Index, Contract의 예이다. 먼저 [표 7]의 Supplier 엘리먼트 예는 Workchairs라는 회사의 정보를 담고 있다.

```

<?xml version= "1.0" encoding= "UTF-8" ?>
<Supplier corporateURL= "http://www.workchairs.com" storeFrontURL=
"http://buy.workchairs.com" >
  <Name xml:lang= "en-US" >Workchairs</Name>
  <Comments xml:lang= "en-US" >this is a cool company</Comments>
  <SupplierID domain= "DUNS" >942888711</SupplierID>
  <SupplierLocation>
    <Address>
      <Name xml:lang= "en-US" >Main Office</Name>
      <PostalAddress>.....</PostalAddress>
      <Email>bobw@workchairs.com</Email>
      <Phone name= "Office" >.....</Phone>
      <Fax name= "Order" >.....</Fax>
      <URL>http://www.workchairs.com/Support.htm</URL>
    </Address>
    <OrderMethods>
      <OrderMethod>
        <OrderTarget><URL>http://www.workchairs.com/cxmlorders
          </URL>
        </OrderTarget>
      </OrderMethod>
    </Contact>
      <Name xml:lang= "en-US" >Mr. Smart E. Pants</Name>
      <PostalAddress>.....</PostalAddress>
      <Email>sepants@workchairs.com</Email>
      <Phone name= "Office" >.....</Phone>
    </Contact>
  </OrderMethods>
</SupplierLocation>
</Supplier>

```

[표 7] Supplier 엘리먼트 예

아래의 [표 8]의 Index 엘리먼트의 예는 예복용 구두의 상품정보를 등록하고, 검색될 속성들을 정의하는 것을 보여준다.

```

<?xml version= "1.0" encoding= "UTF-8" ?>
<!DOCTYPE Index SYSTEM
"http://xml.cXML.org/schemas/cXML/1.1.009/cXML.dtd" >
<Index>
  <SupplierID domain= "InternalSupplierID" >29</SupplierID>
  <Comments xml:lang= "en-US" >Sample cXML/Index</Comments>
  <SearchGroup>
    <Name xml:lang= "en" >DressShoe</Name>
    <SearchAttribute name= "Size" type= "number" />
    <SearchAttribute name= "Color" type= "string" />
    <SearchAttribute name= "Gender" type= "string" />
  </SearchGroup>

```

```

<IndexItem>
  <IndexItemAdd>
    <ItemID><SupplierPartID>pn12345</SupplierPartID></ItemID>
    <ItemDetail>
      <UnitPrice><Money currency= "USD" >60</Money></UnitPrice>
      <Description xml:lang= "en" >Men's black shoes</Description>
      <UnitOfMeasure>EA</UnitOfMeasure>
      <Classification domain= "SPSC" >5136030000</Classification>
      <ManufacturerPartID>MBS3.12</ManufacturerPartID>
      <ManufacturerName>Florsheim</ManufacturerName>
      <URL>http://www.florsheim.com</URL>
      <Extrinsic name= "ManufacturerURL" >http://www.shoemaker.com
      </Extrinsic>
    </ItemDetail>
    <IndexItemDetail>
      <LeadTime>10</LeadTime>
      <ExpirationDate>2000-06-01</ExpirationDate>
      <EffectiveDate>1999-01-01</EffectiveDate>
      <SearchGroupData>
        <Name xml:lang= "en" >DressShoe</Name>
        <SearchDataElement name= "Size" value= "12" />
        <SearchDataElement name= "Color" value= "Black" />
        <SearchDataElement name= "Gender" value= "Men's" />
      </SearchGroupData>
      <TerritoryAvailable>USA</TerritoryAvailable>
    </IndexItemDetail>
  </IndexItemAdd>
</IndexItem>
</Index>

```

[표 8] Index 엘리먼트 예

아래의 [표 9]는 Contract 엘리먼트에 의해 공급자의 상품번호 12345, 12347에 대해 카탈로그 상품정보가 협상된 정보로 대체되는 예이다.

```

<?xml version= "1.0" encoding= "UTF-8" ?>
<!DOCTYPE Contract SYSTEM
"http://xml.cXML.org/schemas/cXML/1.1.009/cXML.dtd" >
<Contract effectiveDate= "1999-01-01T14:32:20-08:00"
expirationDate= "2000-06-01T12:00:00-08:00" >
  <SupplierID domain= "DUNS" >123456</SupplierID>
  <Comments xml:lang= "en-US" >Sample cXML contract
  </Comments>
  <ItemSegment segmentKey= "Detroit" >
    <ContractItem>
      <ItemID><SupplierPartID>12345</SupplierPartID></ItemID>
      <UnitPrice><Money currency= "USD" >1.50</Money></UnitPrice>
    </ContractItem>
  </ItemSegment>
</Contract>

```

```

    <Extrinsic name= "URL" >http://www.workchairs.com/CoolProducts
  </Extrinsic>
</ContractItem>
<ContractItem>
  <ItemID><SupplierPartID>12347</SupplierPartID>/ItemID>
  <UnitPrice><Money currency= "USD" >111.50</Money></UnitPrice>
</ContractItem>
</ItemSegment>
</Contract>

```

[표 9] Contract 엘리먼트 예

4. 1. 2 전자카탈로그 관리

cXML은 검색 방법으로 SearchGroup을 제공한다. SearchGroup의 자식 엘리먼트인 SearchAttribute는 name과 type 어트리뷰트를 사용하여 사용자 중심의 검색 속성들을 정의한다. 그리고 IndexItemDetail이나 PunchoutDetail 엘리먼트 내에 SearchGroupData 엘리먼트를 포함시킨다. SearchGroupData의 자식 엘리먼트인 SearchDataElement는 name과 value 어트리뷰트로 SearchAttribute에서 정의한 항목에 대한 값을 표기함으로써 질의가 가능하게 된다. 각 속성 값인 value 어트리뷰트의 값은 SearchAttribute의 type 어트리뷰트로 정의한 데이터 형식을 따라 정의되어야 한다. SearchGroup 엘리먼트에서 예비용 구두에 대해 검색 속성으로 숫자 형식의 크기(Size), 문자열로 색상(Color), 문자열 형식의 성별(Gender) 속성을 정의하였다. 그리고 SearchGroupData 엘리먼트에서는 해당 상품의 검색 속성 값을 지정한다.

데이터의 수집과 배포는 카탈로그 정보가 필요한 사용자의 요청과 공급자의 응답인 SubscriptionContentResponse 엘리먼트에 버전이나 카탈로그의 형식 등의 정보를 포함한 Subscription, Contract, 상품정보인 Index나 CIF Content를 실어 보내는 방법을 제공한다. 이때 CIF Content는 CIF (Content Interchange Format) 포맷의 카탈로그의 파일을 첨부하며, cXML의 포맷의 카탈로그를 보내려면 Index 엘리먼트 내에 IndexItem

-Add, IndexItemDelete, IndexItemPunchout 엘리먼트로 카탈로그 정보를 담는다. 수집되는 새 상품이나 기존 상품의 변경 정보는 IndexItemAdd 엘리먼트나 IndexItemPunchout 엘리먼트로 전달되고, 삭제할 상품은 IndexItemDelete로 명시한다.

4. 2 xCBL (xml Common Business Library)

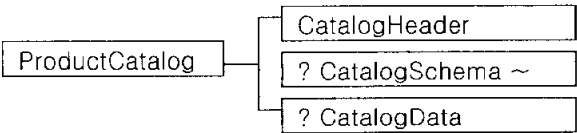
xCBL은 커머스 원(Commerce One)사가 전 업종에서 전자상거래시 교환되는 문서들로부터 추출된 공통적인 요소들과 XML 기술을 결합하여 고안해 낸 규격이다. xCBL 3.0은 대부분 카탈로그 데이터 형식들이 지원하는 상품 속성들의 공통집합을 이용해 속성을 정의할 수 있는 방법을 제시하였다. 또한, xCBL 카탈로그는 상품의 카테고리에 의존적인 많은 속성을 표현할 수 있어서, 분류체계별 검색을 하는 커머스원사의 Content Engine이나 레퀴지트사의 BugsEye와 같은 카탈로그 검색 기술들이 카탈로그 정보들을 쉽게 탐색할 수 있도록 지원한다. 즉, xCBL 3.0은 상품을 표현하기 위한 추가적인 속성을 정의할 수 있게 하며, 상품의 카테고리를 계층별로 분류하고 각 카테고리별로 사용 가능한 속성들을 기술할 수 있도록 하였다.

4. 2. 1 전자카탈로그 DTD

(1) ProductCatalog

루트 엘리먼트는 CatalogHeader, CatalogSchema, CatalogData의 3부분으로 구성된다. DTD의 구성은 다음의 [그림 6]과 같다. CatalogHeader 엘리먼트에는 관리 정보가 기술되고, CatalogSchema 엘리먼트에는 이 카탈로그의

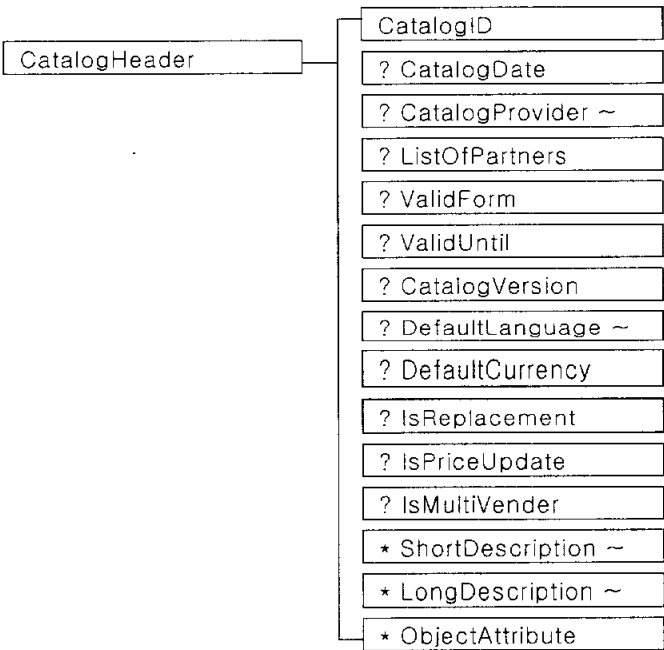
분류체계가 포함되며, CatalogData 엘리먼트가 상품정보를 담는다.



[그림 6] ProductCatalog DTD

(2) CatalogHeader

카탈로그에 대한 관리 정보를 기술한 부분으로 카탈로그 공급업자, 협력사 (Buyer, Manufacturer, Supplier), 유효날짜, 문자 코드, 국가 코드, 소개하는 글 등을 포함하고 있다. DTD의 구성은 아래의 [그림 7]과 같다.



[그림 7] CatalogHeader 엘리먼트

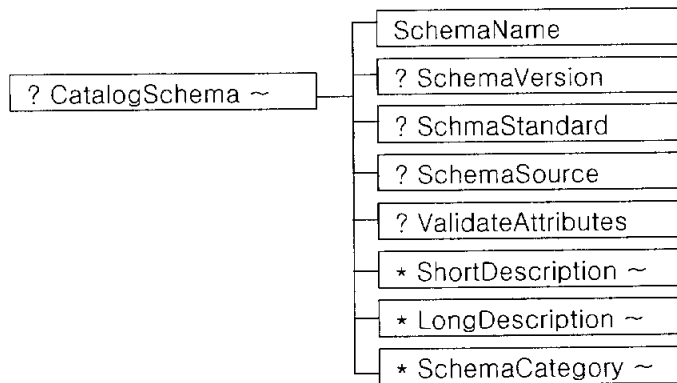
아래의 [표 10]은 랩탑 컴퓨터에 관한 CatalogHeader 부분을 기술한 예제이다. 카탈로그의 ID와 카탈로그가 작성된 날짜, 에이전시 이름, 협력사 이름, 유효일, 카탈로그 버전, 작성 언어, 통화 코드와 짧은 카탈로그 설명을 나타낸다.

```
<CatalogHeader>
  <CatalogID>Acme Laptops Catalog</CatalogID>
  <CatalogDate>20000816</CatalogDate>
  <CatalogProvider>
    <Party>
      <PartyID>
        <Identifier>
          <Agency>
            <AgencyCoded>Other</AgencyCoded>
            <AgencyCodedOther>CommerceOne</AgencyCodedOther>
          </Agency>
          <Ident>Acme_Laptops</Ident>
        </Identifier>
      </PartyID>
    </Party>
  </CatalogProvider>
  <ListOfPartners>
    <Partner PartnerID= "General_Motors" Relationship= "Buyer" />
  </ListOfPartners>
  <ValidFrom>20000816</ValidFrom>
  <ValidUntil>20001231</ValidUntil>
  <CatalogVersion>1.0</CatalogVersion>
  <DefaultLanguage xml:lang= "en" />
  <DefaultCurrency>
    <Currency><CurrencyCoded>USD</CurrencyCoded></Currency>
  </DefaultCurrency>
  <IsMultiVendor/>
  <ShortDescription xml:lang= "en" >This is Acme Laptops's
    catalog of Compaq Laptops, for General Motors.
  </ShortDescription>
</CatalogHeader>
```

[표 10] CatalogHeader 엘리먼트 예

(3) CatalogSchema

상품의 분류체계를 기술하기 위한 엘리먼트로서 상품이 소속되는 카테고리의 ID와 이름, 속성과 설명, 부모 카테고리의 ID 등을 기술할 수 있다. CatalogSchema의 DTD 구성은 다음의 [그림 8]과 같다.



[그림 8] CatalogSchema 엘리먼트

아래의 [표 11]는 위에서 기술한 랩탑 컴퓨터에 관한 CatalogHeader에 이
은 CatalogSchema 엘리먼트의 예제이다. UN/SPSC 코드를 사용하여 부모 카
테고리를 참조하는 자식 카테고리들이 계층별 분류체계로 표현된다.

아래의 [표 11]에서 카테고리 C43171801의 부모 카테고리는 C43171800임을
ParentCategoryRef 어트리뷰트 값으로 명시하고 있다. 카테고리 C43171800은
다시 부모 카테고리로 C43170000을 참조하고, 카테고리 C43170000은 카테고
리 C43000000을 부모로 명시하고 있다.

```

<CatalogSchema>
  <SchemaName>Acme's UNSPSC Subset for Computers
</SchemaName>
  <SchemaVersion>1.0</SchemaVersion>
  <SchemaStandard>UNSPSC</SchemaStandard>
  <SchemaCategory CategoryID= "C43000000" >
    <CategoryName>Communications, Computer Equipment, Peripherals,
      Components and Supplies
    </CategoryName>
  </SchemaCategory>
  <SchemaCategory CategoryID= "C43170000"
    ParentCategoryRef= "C43000000" >
    <CategoryName>Hardware and accessories
    </CategoryName>
  </SchemaCategory>

```

```

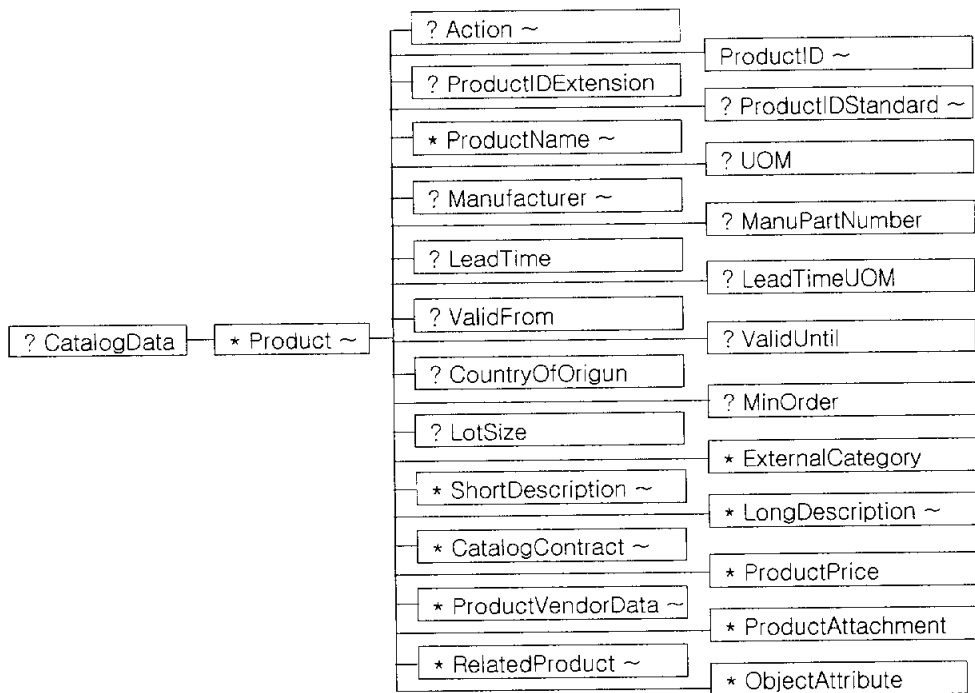
<SchemaCategory CategoryID= "C43171800"
                    ParentCategoryRef= "C43170000" >
  <CategoryName>Computers</CategoryName>
  <CategoryAttribute>
    <AttributeID>Processor Speed</AttributeID>
    <AttributeName xml:lang= "en" >Processor Speed</AttributeName>
    <AttributeType ScalarType= "String" ></AttributeType>
  </CategoryAttribute>
</SchemaCategory>
<SchemaCategory CategoryID= "C43171801"
                    ParentCategoryRef= "C43171800" >
  <CategoryName>Notebook computers</CategoryName>
  <CategoryAttribute>
    <AttributeID>Battery Life</AttributeID>
    <AttributeName xml:lang= "en" >Battery Life</AttributeName>
    <AttributeType ScalarType= "String" ></AttributeType>
  </CategoryAttribute>
</SchemaCategory>
</CatalogSchema>

```

[표 11] CatalogSchema 엘리먼트 예

4) CatalogData

CatalogData는 CatalogSchema에서 분류된 카테고리 ID를 참조하여 상품별로 상세 내역을 기술하는 엘리먼트이다. CatalogData 엘리먼트는 한 개 이상의 Product 엘리먼트를 가질 수 있는데, Product 엘리먼트는 상품의 특성을 잘 설명할 수 있도록 상품 타입, 상품 ID, 단위, 제조업체, 원산지, 제품 설명 등 상세한 항목으로 구성되어 있다. CatalogData의 DTD 구성은 다음의 [그림 9]와 같다.



[그림 9] CatalogData 엘리먼트

다음의 [표 12]은 위에서 기술한 랩탑 컴퓨터에 관한 CatalogHeader와 CatalogSchema에 이은 CatalogData 부분을 기술한 예제이다.

CatalogSchema에서 정의된 카테고리 “C43171801”을 참조한 상품에 대한 정보를 상세히 기록하고 있음을 알 수 있다. 상품의 ID는 “140141-002”이고, 이 상품의 단위는 “EA”를 사용하며, 제조업체는 “Compaq”, 첨부된 그림 파일을 볼 수 있으며, 프로세스의 속도 등을 나열했음을 알 수 있다. 여기에서 프로세스 속도나 배터리 수명을 나타내기 위해 “ObjectAttribute” 엘리먼트를 사용했는데, 이 엘리먼트는 xCBL 3.0 DTD에 명시적인 엘리먼트로 존재하지 않는 상품의 속성을 추가적으로 기술할 수 있도록 제공된 엘리먼트이다.

```

<Product Type= "Good" SchemaCategoryRef= "C43171801" >
  <ProductID>140141-002</ProductID>
  <UOM><UOMCoded>EA</UOMCoded></UOM>
  <Manufacturer>Compaq</Manufacturer>
  <LeadTime>2</LeadTime>
  <CountryOfOrigin>
    <Country><CountryCoded>US</CountryCoded></Country>
  </CountryOfOrigin>
  <ShortDescription xml:lang= "en" >Armada M700 PIII 500 12GB
  </ShortDescription>
  <ProductAttachment>
    <AttachmentURL>file:\5931.jpg</AttachmentURL>
    <AttachmentPurpose>PicName</AttachmentPurpose>
    <AttachmentMIMEType>jpg</AttachmentMIMEType>
  </ProductAttachment>
  <ObjectAttribute>
    <AttributeID>Processor Speed</AttributeID>
    <AttributeValue>500MHZ</AttributeValue>
  </ObjectAttribute>
  <ObjectAttribute>
    <AttributeID>Battery Life</AttributeID>
    <AttributeUnit>
      <UOM><UOMCoded>EA</UOMCoded></UOM>
    </AttributeUnit>
    <AttributeValue>6 hours</AttributeValue>
  </ObjectAttribute>
</Product>

```

[표 12] CatalogData(Product) 엘리먼트 예

4. 2. 2 전자카탈로그 관리

xCBL 3.0은 카탈로그 버전 변경에 따른 카탈로그의 일괄 갱신과 개별 상품에 대한 추가, 삭제 등 운용을 위한 엘리먼트를 제공한다. 먼저 카탈로그 버전 변경에 따른 카탈로그의 일괄 갱신은 카탈로그에 대한 관리 사항을 다루고 있는 CatalogHeader에 주어진다. CatalogHeader의 자식 엘리먼트 중 IsReplacement 엘리먼트가 명시되면, 이는 현재 카탈로그 내용 전체를 CatalogData 엘리먼트의 내용으로 교체하라는 의미이다. 두번째, 개별 상품의 추가, 삭제 등은 상품별로 상세 정보를 포함하는 CatalogData

엘리먼트 내에서 지정한다. CatalogData 엘리먼트의 하위에 개별 상품을 기술하는 Product 엘리먼트는 Action이라는 엘리먼트를 포함하고 있는데 이는 Product 엘리먼트 내에 기술된 데이터를 가지고 어떤 조작을 할지 지시하는 value 어트리뷰트를 가진다. 선택할 수 있는 값은 Add, Update, Delete이며 각각에 대하여 취해져야 할 동작은 아래와 같다. 동작은 CatalogHeader 엘리먼트에 나올 수 있는 IsPriceUpdate 엘리먼트와의 조합에 의해 결정되는데, IsPriceUpdate 엘리먼트가 존재하면 CatalogData 엘리먼트에 담은 정보는 오직 가격 정보만을 갱신하기 위한 것임을 뜻한다.

4. 3 RosettaNet PIP2A와 PIP2B

전자상거래의 각종 업무를 표준화한 RosettaNet은 업무를 크게 6개의 클러스터로 나누고 각 클러스터를 세그먼트별로 세분하여 PIP(Partner Interface Process)를 정의한다. 이 중 카탈로그와 관련된 PIP는 PIP2A1 (Partner Interface Process cluster 2 segment A1), PIP2A2, PIP2B1와 PIP2B4가 있다.

PIP2A1은 신상품에 대한 정보를 구매자에게 배포하여 기업의 시스템이 주문을 처리할 수 있도록 준비시키고 구매자가 가진 카탈로그에 신상품 정보를 추가하는 업무를 규정한다. PIP2A2는 기술적 사항을 제외한 상품정보를 질의하는 업무를 규정하는데 응답으로 반환된 상품에 대한 정보는 온라인 전자카탈로그를 구축하거나 주문처리 시스템을 갱신하는데 이용된다. PIP2B1은 상품 판매자가 구매자에게 상품정보의 변경을 알리고 그에 대한 수신 확인을 받는 업무의 규정이다. PIP2B4는 상품 판매자가 구매자에게 상품의 기술적 사항에 생긴 변경을 알리고 그에 대한 수신 확인을 받는 업무의 규정이다.

PIP2A1, PIP2A2, PIP2B1과 PIP2B4는 업무 수행 주체의 역할을 상품정보 배포자(Product Information Distributor), 구매자(Buyer), 상품정보 사

용자(Product Information User)로 구분하고 이 주체들 사이에서 주거나 받는 메시지를 정의한다. 상품정보 배포자란 상품을 팔거나 상품정보를 제공하는 파트너에 의해 수행되는 역할이며, 구매자는 상품을 구매하는 파트너에 의해 수행되는 역할이다. 이 두개의 역할은 전자상거래 참가자를 조직으로 나눈 역할이다. 상품정보 사용자는 기업 시스템과 전자카탈로그 같은 온라인 홍보 시스템을 갱신하는 활동을 하는 역할로 기업내 담당자일 수도 있고 조직으로 볼 수도 있다.

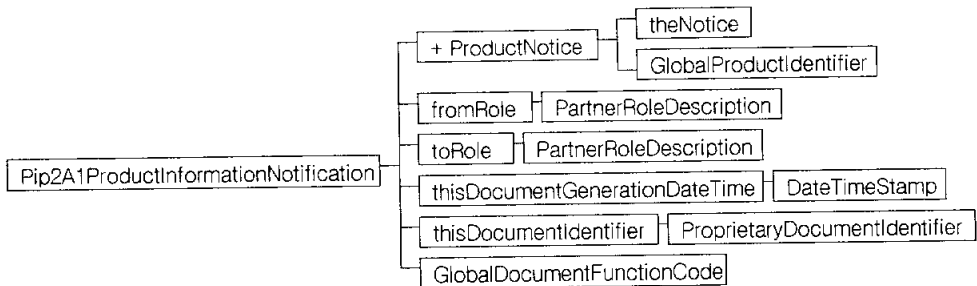
4. 3. 1 전자카탈로그 DTD

(1) Pip2A1ProductInforamtionNotification

Pip2A1ProductInforamtionNotification는 비구조화된 정보를 상품정보 배포자가 구매자에게 전달하는 메시지이다. 자식 엘리먼트로 ProductNotice, from-Role, toRole, thisDocumentGenerationDateTime, thisDocumentIdentifier, GlobalDocumentFunctionCode 엘리먼트를 가진다.

ProductNotice 엘리먼트는 상품에 대한 추가 정보를 통지한다. fromRole 엘리먼트는 이 문서를 보내는 자의 역할을 표시하고, toRole 엘리먼트는 문서를 받는 쪽의 역할을 표시한다.

ThisDocumentGenerationDateTime 엘리먼트는 문서 생성일을 기록한다. ThisDocumentIdentifier 엘리먼트는 전송된 문서를 확인하는 유일한 추적 확인 번호이다. GlobalDocumentFunctionCode 엘리먼트는 요청인지 응답인지 문서의 기능을 확인하는 코드이다. 다음의 [그림 10]은 Pip2A1Product-InformationNotification의 DTD를 나타낸다.

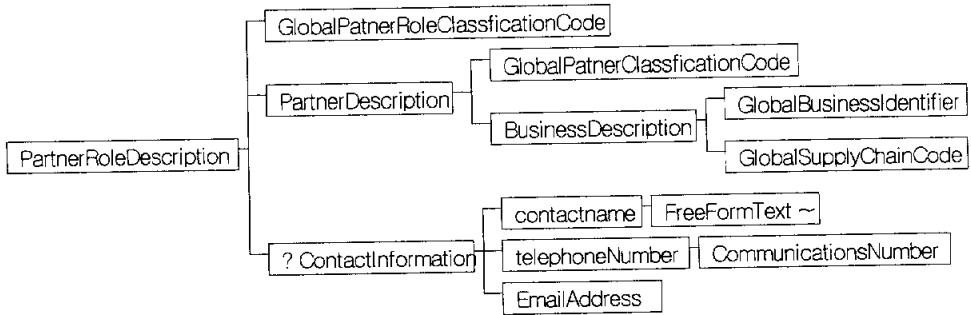


[그림 10] Pip2A1ProductInformationNotification DTD

fromRole과 toRole의 하위 엘리먼트인 PartnerRoleDescription은 PIP에서 비즈니스 파트너의 역할을 기술하는 비즈니스 속성들의 집합이다.

GlobalPartnerRoleClassificationCode는 공급망에서 상대방의 역할을 확인하는 코드이다. PartnerDescription 엘리먼트는 공급망에서 비즈니스 파트너의 ID와 그들의 기능을 기술하는 비즈니스 속성의 집합을 의미하며, 파트너의 기능을 확인할 수 있는 GlobalPartnerClassificationCode와 비즈니스 ID와 위치를 기술하는 비즈니스 속성의 집합인 BusinessDescription을 포함한다.

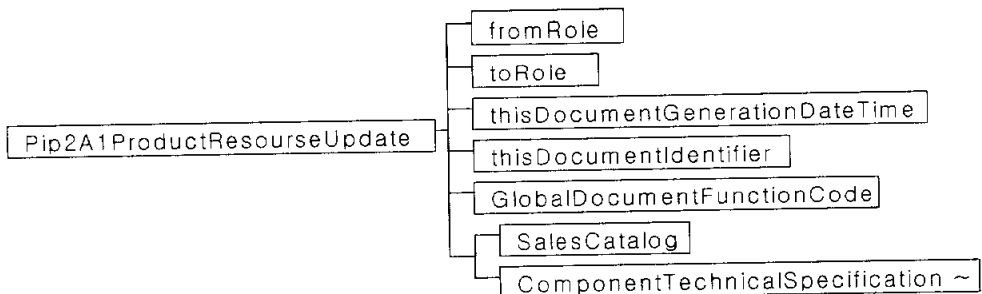
GlobalBusinessIdentifier는 DUNS 번호를, GlobalSupplyChainCode는 정보 기술(Information Technology)과 전자부품(Electronic components)을 값으로 가진다. ContactInformation은 개인, 조직, 협회에 대한 통신과 주소 정보를 제공하는 비즈니스 속성의 집합이다. 다음의 [그림 11]은 PartnerRole-Description 엘리먼트를 상세하게 보여준다.



[그림 11] PartnerRoleDescription 엘리먼트

(2) Pip2A1ProductResourceUpdate

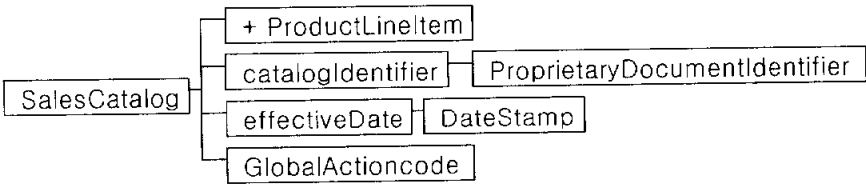
Pip2A1ProductResourceUpdate는 상품정보 배포자가 구조화된 신상품정보를 상품정보 사용자에게 보내는 메시지이다. 위에서 설명한 fromRole, toRole, thisDocumentGenerationDateTime, thisDocumentIdentifier, GlobalDocumentFunctionCode 부분은 동일하다. 단, 공통 엘리먼트들 다음에 SalesCatalog 또는 ComponentTechnicalSpecification 엘리먼트가 온다. 아래의 [그림 12]는 Pip2A1ProductResourceUpdate의 DTD이다.



[그림 12] Pip2A1ProductResourceUpdate 엘리먼트

SalesCatalog는 판매자의 전자카탈로그 정보를 기술하는 비즈니스 속성 집합이며, ProductLineItem, catalogIdentifier, effectiveDate, GlobalActionCode 엘리먼트로 구성된다. ProductLineItem은 상품정보를 기술하는 엘리먼트이고, catalogIdentifier는 전자카탈로그의 독자적인 ID이며, effectiveDate는 카탈로그의 유효기간을 표시한다.

GlobalActionCode 엘리먼트는 이 메시지를 받고 취해야 할 동작의 종류를 의미하는 코드인데, Pip2A1ProductResourceUpdate에서는 ADD가 명시된다. 아래의 [그림 13]은 SalesCatalog의 자식 엘리먼트를 나타낸다.

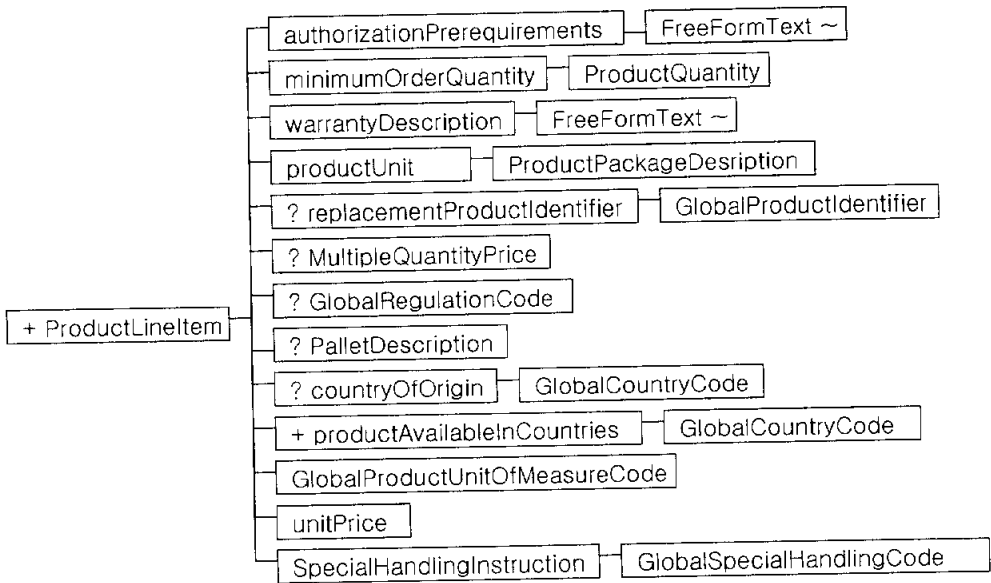


[그림 13] SalesCatalog 엘리먼트

ProductLineItem은 상품의 속성을 기술하는 엘리먼트이다. 자식 엘리먼트인 AuthorizationPrerequisites은 상품 판매 권한을 얻기 위한 요구사항들을 기술한다. MinimumOrderQuantity는 주문의 최소량을 나타내고 warranty-Description는 상품의 보증기간을 문자열로 기술한다. ProductUnit은 상품의 단위이며 ReplacementProductIdentifier는 상품이 대체되었음을 표시하고, MultipleQuantityPrice는 주어진 가격에 대한 단위의 양을 말한다.

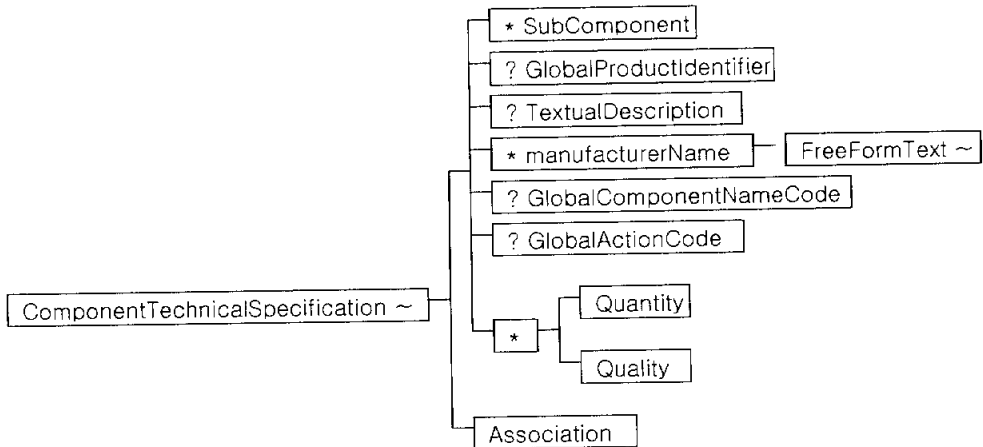
GlobalRegulationCode는 상품이나 서비스의 판매 또는 사용을 지배하는 규칙들이다. PalletDescription는 팔레트와 관련된 순번을, CountryOfOrigin는 상품의 원산지 국가를 나타낸다. ProductAvailableInCountries는 판매가 가능한 상품의 국가들을 나타낸다. GlobalProductUnitOfMeasureCode는 측량 단위를 나타내며 UnitPrice는 단위별 가격, SpecialHandlingInstruction는 상품의 포장

이나 배송시의 지시사항을 기술한다. 다음의 [그림 14]는 ProductLineItem의 자식 엘리먼트를 나타낸다.



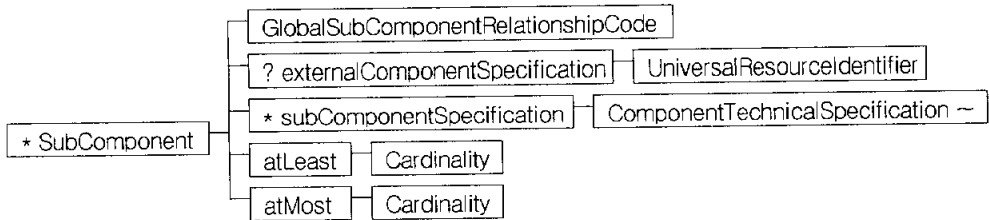
[그림 14] ProductLineItem

ComponentTechnicalSpecification는 부품의 기술적인 명세서를 나타내는 엘리먼트이다. GlobalProductIdentifier, TextualDescription, GlobalActionCode는 위에서 설명되었다. manufactureName은 제조업자의 역할로서 비즈니스의 이름을 가진다. GlobalComponentNameCode는 부품의 이름을 나타낸다. Association은 reference라는 어트리뷰트를 가지는데, 이것은 다른 Component-TechnicalSpecification의 ID를 참조한다. 다음의 [그림 15]는 Component-TechnicalSpecification의 자식 엘리먼트를 나타낸다.



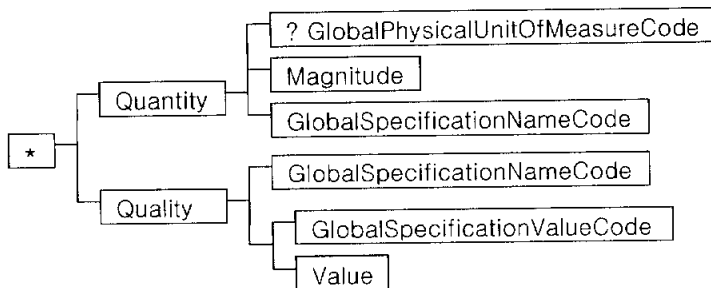
[그림 15] ComponentTechnicalSpecification 엘리먼트

SubComponent는 부품의 부분을 말하며, GlobalSubComponentRelationship-Code, atLeast, atMost, externalComponetSpecification, subComponentSpecifi-cation 엘리먼트를 포함한다. GlobalSubComponentRelationshipCode는 부품사이의 관계를 표시하는데, 예를 들면, Exclusivelyrequires는 메모리 모듈과 메모리 슬롯의 관계로, 한 개의 메모리 모듈이 메모리 슬롯을 독점적으로 필요로하며 슬롯은 공유될 수 없음을 의미한다. ExternalComponentSpecification는 부품 기술 명세서 바깥쪽에 위치한 부품 명세서로 링크를 가지는 엘리먼트이다. SubComponentSpeicification은 부품 명세서가 다른 부품의 명세서를 가질 수 있음을 의미한다. 예를 들면, 컴퓨터 시스템은 하드웨어와 키보드 같은 다른 부품으로 구성할 수 있는 부품이다. atLeast는 부품 명세서에서 연관된 부품의 최소 수이며, atMost는 최대 수이다. 다음의 [그림 16]은 SubComponent의 자식 엘리먼트를 나타낸다.



[그림 16] SubComponent 엘리먼트

Quantity는 측량되는 특징을, Quality는 질적 특징을 기술한다. Quantity는 GlobalPhysicalUnitOfMeasureCode, GlobalSpecificationNameCode, Magnitud 엘리먼트를 포함한다. GlobalPhysicalUnitOfMeasureCode는 물리적 양의 측량 단위를 말하며, 예로는 Byte, Bit 등이 있다. GlobalSpecificationNameCode는 기술명세서의 이름을 나타낸다. Magnitude는 물리적 측량의 양을 나타낸다. Quality의 자식 엘리먼트인 GlobalSpecificationValueCode는 기술명세서에서 값의 이름을 말하는데, 예를 들면, odd, OK, No signal 등이 있다. value는 속 성값을 가지는 엘리먼트이다. 아래의 [그림 17]에서 Quantity와 Quality의 자 식 엘리먼트를 볼 수 있다.



[그림 17] Quantity와 Quality 엘리먼트

4. 3. 2 전자카탈로그 관리

PIP2A1에서 정의된 메시지에는 ProductInforamtionNotification와 ProductResourceUpdate가 있다. 상품에 대한 비구조화된 정보는 ProductInformationNotification에 기술되고, 구조화된 정보는 ProductResourceUpdate에 기술되어 전달된다. 구조화된 상품정보는 판매 카탈로그 정보와 기술정보로 나뉜다. 판매 카탈로그 정보는 양, 가격, 마케팅정보를 포함하여 비전문가인 고객에게 상품정보를 홍보하는 데 사용된다. 기술 정보는 전문적인 상품을 위하여 상품을 향상시키고, 카탈로그를 구성하기 위하여 내용을 생성하는데 사용된다.

PIP2A2에서는 ProductInformationQuery 메시지를 상품정보 배포자에게 보낸다. ProductInformationQuery는 상품의 정보에 대한 질의를 요청하는 메시지로서 자식 엘리먼트인 QueryConstraint 엘리먼트에 하나 이상의 매개변수를 준다. QueryConstraint 다음에 SalesCatalog 엘리먼트를 포함한다. 질의에 응답하여 2A2ProductInformationResponse가 상품정보 사용자에게 반환되는데 이 메시지가 PIP2A1의 SalesCatalog와 다른점은 메시지 부분에 Association 엘리먼트가 없다는 것이다. 아래의 보기는 상품 중에 현재 통화 코드가 USD이고, 통화량이 1000보다 작은 상품의 내역을 요청하는 메시지이다. 질의 결과 응답 메시지는 상품정보 배포자가 요청한 상품 내역이다.

PIP2B1에 의해 상품정보 배포자는 상품정보의 변경을 구매자에게 통보하기 위하여 BasicProductInformationChange 메시지를 전송한다. 메시지는 PIP2A1에서 사용되는 SalesCatalog 엘리먼트를 포함하여 변경된 내역을 전달한다. PIP2B1은 구매자 시스템에 변경 통보가 도착했음을 확실시키기 위하여 변경 통보 절차로서 확인메시지인 ChangeBasicProductInformationConfirmation를 보내도록 규정한다.

부품기술 명세서 변경에 관한 통보는 PIP2B1과 비슷하다. PIP2B4에 규정된 대로 ProductTechnicalInformationChange 메시지를 구매자에게 보내고, 확인서로서 ChangeProductTechnicalInformationConfirmation이 상품정보 배포자에게 반환되어야 한다.

4. 4 전자카탈로그 비교

앞에서 설명한 cXML, xCBL, RosettaNet 국제 전자카탈로그 규격은 XML을 포맷으로 채택하여, 전자카탈로그의 내용을 담은 엘리먼트와 어트리뷰트, 또한 구조를 DTD에 정의한다. 따라서 각 DTD에 할당된 엘리먼트와 어트리뷰트의 이름과 위치가 해당 규격이 허용하는 전자카탈로그의 내용을 암시한다. 3개의 DTD에서 표현하고자 하는 카탈로그의 내용을 분석하여 보면 다음과 같다.

내용은 크게 운영을 위한 정보와 상품정보로 나뉜다. 먼저 운영을 위한 정보는 카탈로그 관리 정보와 동작 정보로 나눌 수 있다.

카탈로그 관리 정보는 카탈로그의 버전, 수정 내역 등과 같이 카탈로그 자체에 대한 정보이고, 동작 정보는 추가, 갱신, 삭제와 같은 동작을 지시한다. 카탈로그 관리 정보는 3장에서 살펴 본 모든 DTD에 엘리먼트와 어트리뷰트가 정의되어 있으며, 수정 날짜, 유효 날짜, 버전, 부가 설명, 사용 언어 등이 해당된다. 특히 xCBL는 CatalogHeader 엘리먼트로 명시적으로 관리 정보를 기술하게 한다.

동작 정보는 DTD 자체에 엘리먼트나 어트리뷰트로 존재하지는 않지만 프레임워크에 의해 지시될 수도 있다. RosettaNet의 경우 PIP가 업무를 종류별로 나눈 것이기 때문에 PIP가 카탈로그 추가 업무인지 갱신인지 통보인지를 구분하여 준다. 이와 같이 프레임워크에 의존적인 규격과 달리 cXML에서는 IndexItemAdd, IndexItemDelete 엘리먼트가 동작의 종류를 암시하며, 동작의 대상이 되는 상품정보는 자식 엘리먼트 내에서 기술된다. xCBL은 카탈로그 내에서 상품정보와 함께 동작이 표시되는데, action 어트리뷰트를 사용한다.

RosettaNet는 프레임워크에 의존하므로 카탈로그 규격이 독립적으로 존재할 수 없다는 단점이 있다면, 엘리먼트나 어트리뷰트로 정의하는 경우 카탈로그가 프레임워크로부터 자유롭다는 장점이 있다. 카탈로그 DTD로 동작까지 표현하는 방법에는 엘리먼트를 이용하는 방법과 어트리뷰트를

이용하는 방법 두 가지가 있다. 이 중 cXML과 같이 엘리먼트를 이용하는 경우 엘리먼트 이름이 동작의 종류를 암시하면 다른 동작을 추가할 때 DTD를 수정해야 하는 단점이 있다. xCBL과 같이 어트리뷰트를 이용할 때는 동작의 종류가 어트리뷰트 값이므로 동작이 추가되었다고 할지라도 DTD를 수정할 필요는 없다는 장점이 있다.

	cXML	xCBL	Rosettanet
동작 정보	엘리먼트 이용, 엘리먼트 이름이 동작의 종류를 암시하면 다른 동작을 추가할 때 DTD를 수정해야 하는 단점	어트리뷰트 이용, 동작의 종류가 어트리뷰트 값이므로 동작이 추가되었다고 할지라도 DTD를 수정할 필요가 없는 장점	프레임워크에 의존, 카탈로그에 독립적으로 존재할 수 없다는 단점
식별 코드	특정 식별 코드 표준안에 한정되지 않는다.		
분류 코드	특정 분류 코드 표준안에 한정되지 않는다.(UN/SPSC, SPSC코드)		
상품 속성	상품 공통 속성에 대한 부분을 상세히 정의	속성 이름, 값을 지정할 수 있는 요소를 제공한다.	Dictionary 제공
전자카탈로그 스키마	XML 스키마와 DTD 형태로 상세히 기술	XML 스키마 형태로 자세히 제공한다.	XML 문서로 정의
비즈니스 문서 스키마	주문, 전송, 교환 등등에 대한 DTD가 정해져 있다.	XDR, DTD, SOX 형태로 제공된다.	XML 문서로 정의

[표 13] 전자카탈로그 비교

상품정보는 상품 속성, 주문 정보, 배송 정보, 거래처 정보로 분류할 수 있다. 이 중 전자카탈로그 표준화에서 업종별 업체별 국가별 차이가 가장 큰 분야는 분류 및 식별 코드의 표준인데, 앞에서 살펴본 XML 기반의 카

탈로그 규격(cXML, xCBL, RosettaNet)은 특정 분류체계나 식별 코드 표준안에 의존적이지 않다는 공통적인 장점이 있다.

사용된 식별 코드 표준은 어트리뷰트 또는 엘리먼트로 명시하므로 다른 식별 코드의 표준으로 이전할 때 카탈로그의 구조를 변경할 필요 없이 값만 변경하여 주면된다.

분류체계 역시 특정 체계 표준안에 한정되지 않는다. xCBL은 Schema-Category 엘리먼트로 각 카테고리의 이름과 속성, 그리고 상위 카테고리의 ID를 기술한다. 이렇게 기술된 각 카테고리들은 SchemaCatalog 엘리먼트에 자식으로 위치하고 SchemaCatalog는 그 외에 분류체계에 대한 간단한 설명과 관리 정보를 포함한다.

기본적인 상품 속성, 즉 ID나 이름, 간단한 설명, 가격 등은 대응 엘리먼트가 존재하지만, XML 기반의 전자카탈로그 규격은 일반적으로 각 속성에 한 엘리먼트를 할당하는 방법 대신 속성을 카탈로그에서 자유롭게 정의하도록 보편적인 엘리먼트를 제공한다. xCBL의 ObjectAttribute 엘리먼트가 속성 정의에 사용된다.

위에서 살펴본 장단점 이외에 cXML은 인터넷에서 공급자와 구매자간에 콘텐츠와 거래 자료를 주고받기 위한 언어로 이것은 일반거래에서 쓰이는 여러 트랜잭션들의 품을 표준화시킨 것이다. 상품의 분류면에서 본다면, cXML을 통해서 구매자는 상품들 사이의 일련의 계층관계를 알 수 없고, 공급자가 가지고 있는 상품의 세부적인 아이템과 세부적인 정보까지만 알아 차릴 수 있도록 표준으로 정해 놓은 것이다. 그래서 상품정보의 중복적인 표현이 이루어지고 있다.

xCBL은 표준화된 XML문서 포맷을 제공하여 기존 시스템을 효율적으로 e-마켓플레이스에 통합할 수 있게 해주기 때문에 기존에 만들어진 시스템을 통합할 수 있도록 다양한 상품을 표현하는 것에 중점을 두었다. 그래서 다양한 상품의 표현은 가능하지만 하나의 상품의 세부적인 표현이 어렵다는 문제점이 있다.

RosettaNet은 전자부품의 전자상거래 업무와 교환 메시지, 그리고 공통 용어를 상세히 기술했다는 장점이 있으나, 전자 업종의 상품 표현에 특화

되어 있고 카탈로그에 관한 업무가 새 상품 통보, 새 상품정보 전송, 상품 정보 변경 통보, 기술 정보 변경 통보로 정해져 있다는 단점이 있다. 따라서 해외 기업들의 정형화된 업무와 국내 기업의 업무 사이에 존재하는 차이를 반영하기 어려울 수 있다.

제 V 장 XBC 전자카탈로그 설계

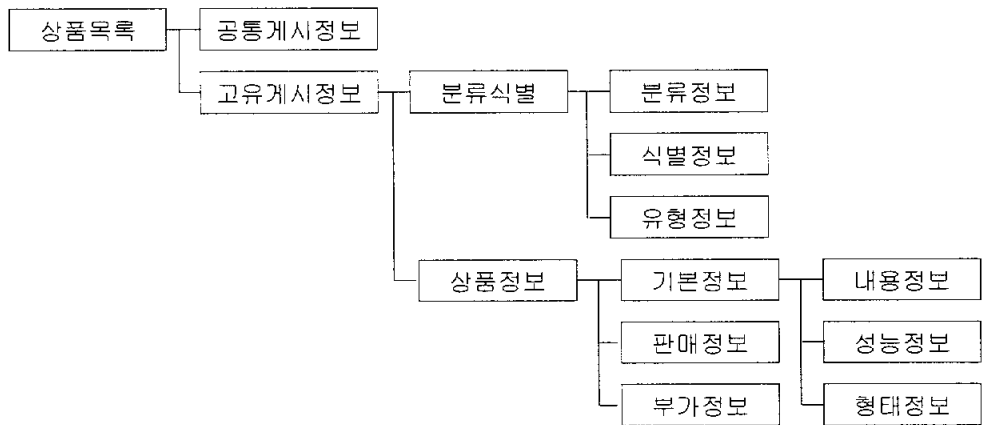
cXML, xCBL, RosettaNet 연구를 기초로 하여 기존 표준들의 문제점을 발견하고 문제점을 보완할 수 있는 다양한 상품의 정보를 자세하게 표현할 수 있는 전자상거래를 위한 XBC(XML Based Catalog) 전자카탈로그 모델을 제시한다. 이 모델에 대한 가정은 기본적으로 B2C환경에 기반하고 있으며, 상품자체의 정보를 기술하는데 주안점을 두어 배송이나 지불은 고려하지 않는다는 것이다. 그러나 이 모델은 범용성이 높아 B2B환경이나 배송, 지불 요소 등 다른 요소를 손쉽게 표현할 수 있도록 확장 가능하다.

5. 1 XBC 전자카탈로그 모델 원형

모델 원형 설계시 가장 큰 주안점을 둔 것은, 전자상거래에서 통용 가능한 모든 상품에 대한 정보를 한개의 모델로 통합하여 기술한다는 것이다. 이를 위해서, 먼저 다양한 상품들을 일반적인 모델로 표현할 수 있도록 모든 상품이 갖는 속성들을 유형화하였다. 본 모델 설계시 고려된 유형화의 기준은 다음과 같다. 첫째, 유형들 간에는 상호 배타적이고 완전해야 하고 둘째, 상품정보의 공통게시정보와 고유게시정보를 모두 고려하여 모델의 범용성을 높이는 것이며, 셋째 상품속성은 상품자체에 속하는 것으로 국한하였다. 즉, 배송, 지불처럼 판매 업체와 관계된 속성들은 고려하지 않았다.

위의 상품속성 유형화 기준에 기반하여 모델은 상품속성들을 공통게시정보와 고유게시정보로 유형화한다. 공통게시정보는 상품목록 표현시 중복적으로 표현되는 부분을 추출하였다. 고유게시정보는 분류식별정보와 상

품정보로 나누어 설명한다. 분류식별 속성은 이질적인 상품군들간에서 동일 상품군으로 분류를 가능하게 하며, 동일 상품군내에서 특정 상품을 식별하기 위한 속성 정보를 포함한다. 이러한 속성은 분류정보, 식별정보, 유형 정보로 세분화된다. 상품정보는 상품을 제공하는 주체, 즉 정보 제공자로부터 제공되는 속성이다. 이러한 속성은 누가 정보를 제공하느냐에 따라서 다음과 같이 세분화된다. 이는 생산자(공급자)가 제공하는 기본 정보와 판매자가 제공하는 판매 정보, 그리고 고객 등 제3자 외부인이 제공하는 부가가치 정보이다. 상품본질은 모든 상품들이 갖는 본원적 가치를 제공하는 속성이다. 이러한 속성은 상품의 본원적 가치에 따라 내용 정보, 성능 정보, 형태 정보로 세분화될 수 있다.



[그림 18] XBC 모델 원형

[그림 18]은 위의 상품속성 유형화에 따른 모델의 원형을 보여준다. 이 원형에는 상품정보의 생산 정보가 기본정보로 표현되고 있는데, 이것은 생산자가 제공하는 상품목록이 상품의 기본정보를 형성하기 때문이다. 또한, 상품본질이 정보 원천의 기본정보의 요소로 표현되고 있다. 이것은 분류식별과 상품정보는 독립적이지만, 상품본질은 기본정보의 일부로 기술될 수 있기 때문이다.

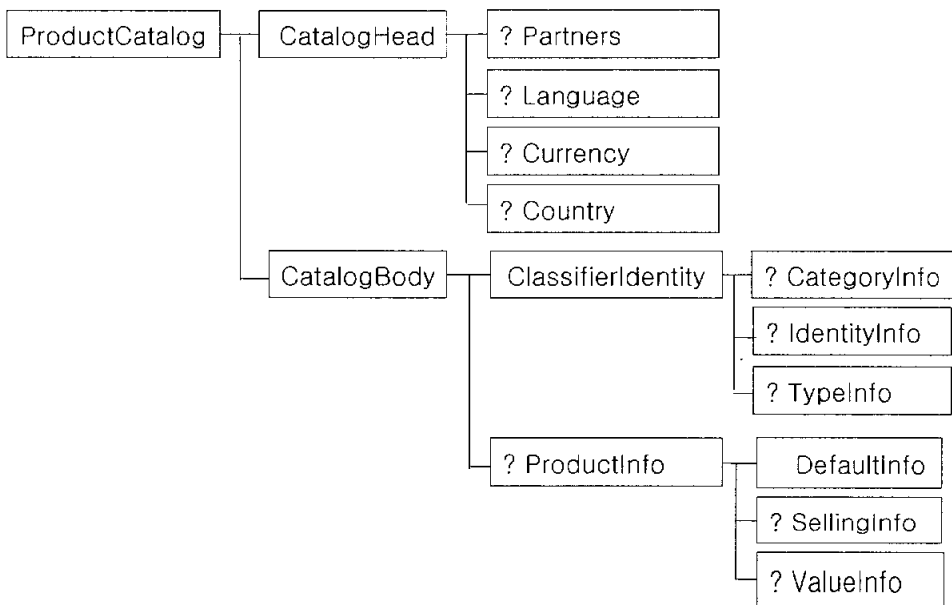
설계된 전자카탈로그 모델 원형은 한국전산원에서 발표한 “전자카탈로그의 게시항목 권고안”을 모두 만족하도록 설계되었다.

구분	필수게시항목	부가게시항목
제품 정보	제품명, 모델명, 제품생산일, 제품의 단위명칭, 제품수량, 제품제조원, 판 매원, 원산지, 제품의관정보(색상, 재 질, 용량, 무게, 규격) 제품내용물정 보, 제품기능정보(핵심특성, 주요사 양 등), 세트상품 또는 별도구매 표 시, 이미지 파일 및 설명, 취급주의 사항	제품코드, 제품약칭, 제품 그룹/ 종류, 제품관련 기술적 규정
가격 정보	제품가격(총 구매가격), 가격표시통 화, 세금내역(소비세, 관세 등), 가격 책정날짜, 가격책정수량, 대금지급 조건 및 방법(결제방법), 대금 지급 시기, 제품의 판매가격 외에 소비자의 추가부담 비용 내용 및 금액(별 도요금, 보험료, 수수료, 기타 부대 금액 등)	할인정보
업체 정보	사업자명칭, 상호, 점포명, 업체의 국 적, 소재지, 성명, 우편주소, 전화번호, E-mail 주소	도시명, 팩스번호, 법인의 관할, 사 업자등록번호, 법령 규정의 상업 적 등록정보, 업법관련 자격 내역
판매 정보	거래제한, 교환 조건, 환불/반품 조 건 및 내용(방법, 기간, 연락처), 취 소 조건 및 방법(계약철회권존재유 무, 해약조건, 방법, 피해구제지), 품 질보증마크, 주문방법, 영업일시, 판 매지역, 판매수량의 제한, 사은품유 무	한정판매유무, 재고보유여부, 사 용설명서, 품질보증(AS/보증)기 간, 청약기간, 유효기간, 제품의 인도 시기(기한), 기타 특별한 판 매조건이 있을 때는 그 내용

[표 14] 한국전산원 “전자카탈로그의 게시항목 권고안”

5. 2 XBC 전자카탈로그 모델

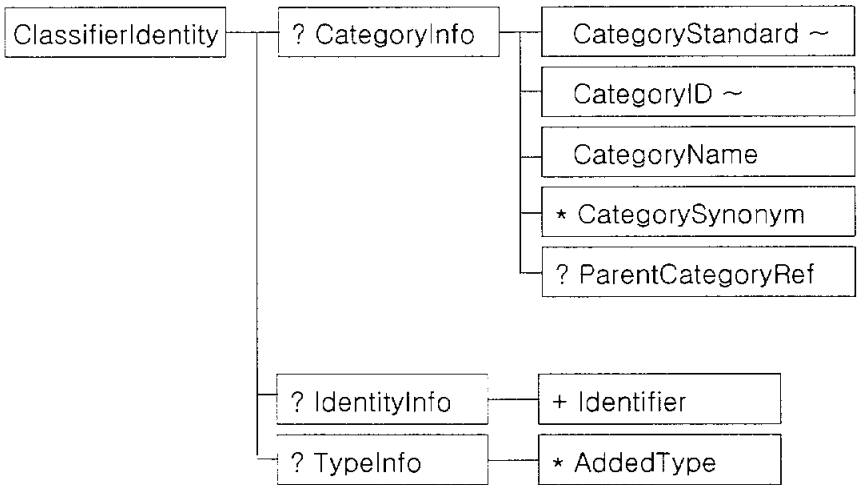
위의 [그림 18]의 상품표현 모델의 원형에 기반하여 전자카탈로그를 제안하였다. 아래의 [그림 19]은 본 논문에서 제안한 전자카탈로그를 나타낸다. 카탈로그 명세에서 박스 상에 기술된 요소는 XML의 엘리먼트에 대응될 수 있다. 실제로, 카탈로그 모델은 XML로 기술된다는 것을 생각하고서 설계되었다.



[그림 19] XBC(XML Based Catalog) DTD

XBC는 CatalogHead와 CatalogBody로 크게 구분된다. CatalogHead에는 전자카탈로그 전체에 공통된 요소와 CatalogBody부분에서 중복 가능성이 높은 요소가 위치한다. Language는 카탈로그 기술에 사용되는 기본적인

인 언어를 나타내고, Currency는 상품가격에 대한 기본 통화를 나타내고, Country는 상품의 원산지를 나타낸다. 물론, CatalogBody부분에서 상품정보 기술시 다른 언어와 다른 통화가 사용될 수 있다.



[그림 20] ClassifierIdentity 엘리먼트

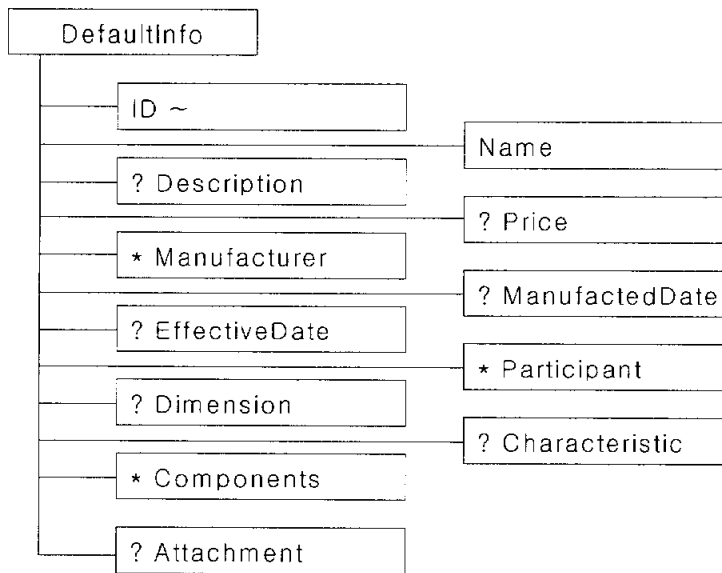
CatalogBody상에는 상품의 정보가 기술된다. [그림 20]은 CatalogBody의 분류, 식별, 유형 정보를 기술하는 ClassifierIdentity에 대한 상세 구조를 보여준다. CategoryInfo를 통해서 상품의 분류체계가 표현된다.

CategoryStandard은 분류체계를 나타내고, CategoryID는 분류코드를 나타낸다. CategoryName은 분류이름이 기술되고 CategorySynonym에는 이 분류 이름과 유사한 분류들이 기술될 수 있다. 이것은 사용자가 상품에 대한 정확한 분류를 모를 경우에도 검색을 지원하기 위함이다.

ParentCategoryRef는 부모 카테고리를 참조하여 부모 카테고리와 자식 카테고리의 계층적인 분류체계를 표현한다.

TypeInfo는 특정 대표 상품 내에서 원하는 상품에 대한 검색시 소비자의 편의를 지원하는 부가유형을 기술하는 AddedType으로 구분된다. 부가유형은 제품 특성을 반영하는 내용, 용도, 형태, 사용자, 사용시기, 사용 장

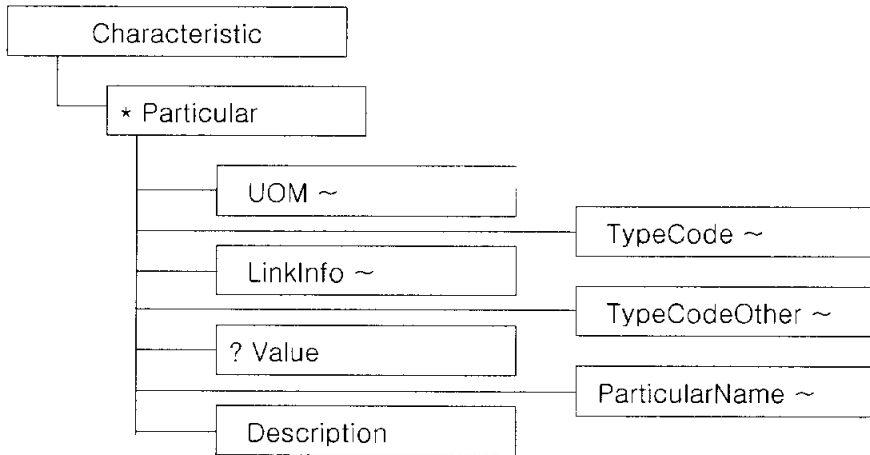
소 등에 따른 구분과 상품관측을 위한, 신상품, 추천상품, 인기상품, 수상 상품 등의 구분을 제공한다. 이러한 다양한 유형정보는 사용자에게 유용한 검색 패턴을 제공한다.



[그림 21] DefaultInfo 엘리먼트

ProductInfo는 기본정보를 기술하는 DefaultInfo, 판매정보를 기술하는 SellingInfo, 부가가치정보를 기술하는 ValueInfo로 구분된다. 위의 [그림 21]는 DefaultInfo에 대한 상세 정보를 보여준다. ID는 해당 상품을 식별할 수 있는 식별자이고, Manufacture와 Participant의 세부적인 정보는 CatalogHead상에 각각의 제조사와 상품컨텐츠 제공자에 대한 정보를 기술하며, Dimension은 상품의 규격/무게에 대한 정보가 기술된다.

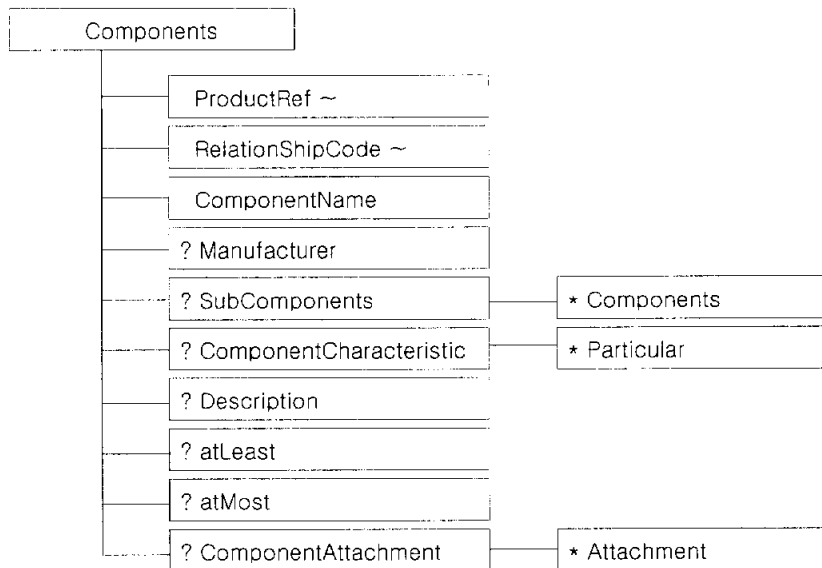
또한, Characteristic에는 상품의 전반적인 속성 정보가 기술되며, Components에는 상품의 부품에 대한 상세 정보가 기술된다. Attachment에는 그래픽이나 동영상 등 상품정보를 추가적으로 기술하는 파일들이 기술된다.



[그림 22] Characteristic 엘리먼트

Characteristic은 [그림 18]의 상품 본질의 속성을 기술하는 부분으로, 위의 [그림 22]를 통해서 좀더 상세히 기술한다. 상품 본질 속성은 여러개의 항목으로 기술된다. TypeCode속성은 명세의 유형을 나타낸다. 이러한 유형은 앞서 언급하였듯이, 내용, 성능, 형태 등의 3가지 유형으로 구분되고 이 유형들은 다음과 같이 규격, 중량, 재질, 성분, 색상, 모양, 포장, 성능, 내용으로 세분화된다. 다음은 본 논문에서 제시한 유형의 코드이다. Size, Weight, Texture, Ingredient, Color, Shape, Packing, Performance, Content, Others. 이 외의 다른 유형은 TypeCodeOthers상에 표현될 수 있다. 그러나 대부분의 상품이 위의 3가지 속성 유형에 속한다. Particular Name은 특정 유형에 속한 상품속성의 명칭이 기술되고, 해당 속성 값은 Value상에 표현된다.

가령, 컴퓨터 상품의 속성 기술시, 성능의 유형으로 CPU, RAM, 보조 기억장치 등이 ParticularName상에 기술될 수 있다. RAM의 경우 그 용량은 Value상에 기술되고, UOM에는 그 측정 단위가 기술된다.



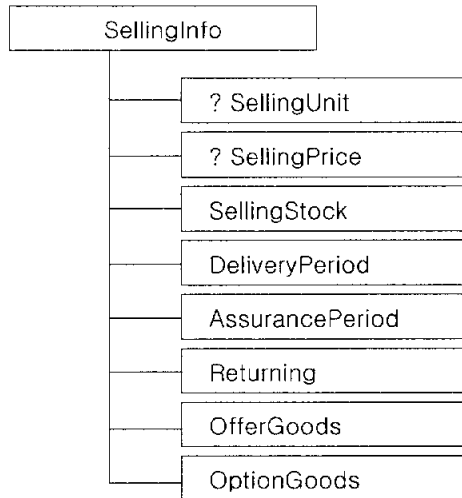
[그림 23] Component 엘리먼트

위의 [그림 23]에서 보여주는 Component는 상품의 부품을 상세히 기술한다. ProductRef는 부품이 이미 기술된 상품일 경우에 그 상품을 참조할 수 있는 식별자를 가리키며, RelationshipCode는 부품과 상품간의 관계를 나타낸다. 부품에 대한 특성은 ComponentCharacteristic에서 상품 특성을 기술과 동일한 방식으로 Particular을 통해서 기술된다.

atLeast와 atMost는 각각 부품의 최대 개수와 최소 개수를 나타낸다. SubComponent는 병시되고 있는 부품이 여러 부품들로 구성될 경우에, 그 구성 부품들을 또다시 Component를 사용하여 보다 상세히 기술할 수 있게 한다. 예를 들어, 컴퓨터는 마더보드 부품을 포함하며, 이 마더보드는 또한 메모리 슬롯의 부품을 갖는다. 이와 같은 방식은 상품특성을 단계적으로 세분화시키면서, 각 부품의 특성과 부품간의 연관관계를 상세히 기술할 수 있도록 해준다.

다음의 [그림 24]의 SellingInfo 엘리먼트에는 상품의 판매와 관련된 모든 정보가 기술된다. 판매 단위, 판매가, 재고상태, 주문배송기간, 보증기간, 반품 가능성, 제공품, 옵션 상품 등이 구체적으로 기술된다. 이러한 정

보는 판매업체에 따라 달라질 수 있으므로, 각 판매업체별로 각각 기술할 수 있다.



[그림 24] SellingInfo 엘리먼트

ValueAddedInfo에는 소비자의 제품 선택 편의와 합리적 선택을 도와주는 부가가치 정보가 기술된다. 여기에는 상품과 관련된 유사제품이나 보완 제품에 관한 정보가 기술되며, 상품에 대한 소비자, 언론, 기관 등 제3자의 평가 정보가 기술된다.

제 VI 장 XBC 전자카탈로그 적용

이번 장에서는 5장에서 제안된 XBC(XML Based Catalog) 전자카탈로그 모델을 웹 상의 다양한 상품에 적용한다. 먼저, 전자카탈로그를 XML DTD로 기술한다. DTD는 xCBL에서처럼 기본 블록에 기반하여 표준 문서를 작성할 수 있도록 작성되었다. 결과적으로 총 11개의 DTD로 작성되었으며, 각 DTD는 재사용성이 높은 기본 블록을 나타낸다.

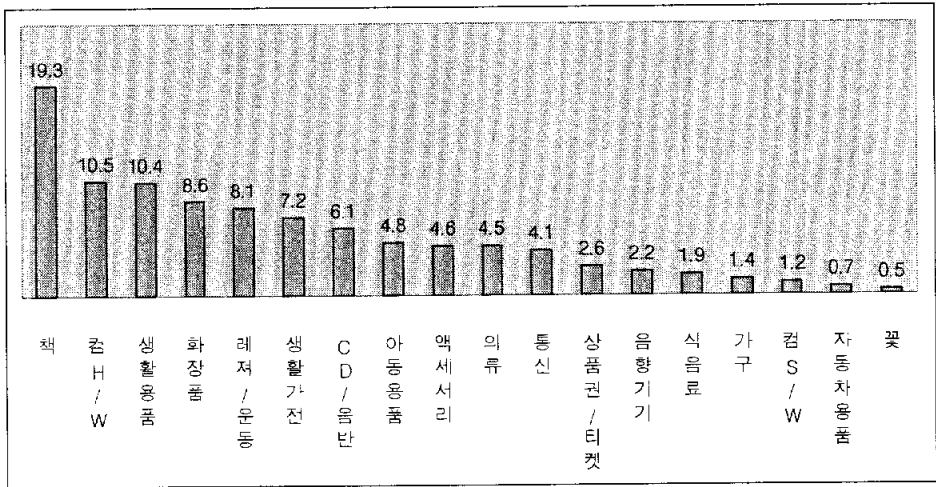
DTD 파일 이름	내 용
Catalog.DTD	카탈로그의 최상위 요소
CatalogHead.DTD	CatalogHeader요소의 하위 요소들을 표현한다.
CatalogBody.DTD	CatalogBody요소의 하위요소들을 표현한다.
ClassifierIdentity.DTD	ClassifierIdentity요소로 상품의 분류, 식별정보를 표현한다.
ProductInfo.DTD	ProductInfo요소의 하위요소들을 표현한다.
DefaultInfo.DTD	DefaultInfo요소의 하위요소를 표현한다.
Component.DTD	Component요소로 상품의 부품을 계층적으로 표현한다.
Specification.DTD	Specification요소의 하위요소로서 상품의 특징을 표현한다.
SellingInfo.DTD	SellingInfo요소로 상품의 판매에 관한 정보를 표현한다.
ValueAddedInfo.DTD	ValueAddedInfo요소로 상품의 부가가치정보를 표현한다.
Description.DTD	Description 요소로 서술적인 설명을 위한 클래스이다.

[표 15] DTD파일의 내용

[표 15]는 11개의 DTD파일의 구체적인 내용을 나타낸다. 웹 상에 표현되어 있는 상품을 임의로 추려 제안된 전자카탈로그를 적용하여 기술하였다. 본 논문에서는 2001년 통계청에서 조사한 “가장 최근에 전자상거래로 구매한 상품”에서 나타난 항목인 책, 컴퓨터H/W, 생활용품, 화장품, 레저/운동, 생활가전, CD/음반, 아동용품, 액세서리, 의류, 통신, 상품권/티켓,

음향기기, 식음료, 가구, 컴퓨터S/W, 자동차용품, 꽃 중 상위에 랭크된 상품을 선정하여 인터넷 웹사이트에서 정보를 수집하였다.

수집된 상품정보들을 구현된 DTD에 적용하여 본 결과 본 상품모델은 단일한 상품모델로 2개의 상품을 기술하는데 어려움이 없었으며, 각 상품의 속성 모두 효과적으로 기술되었다.



[그림 25] 가장 최근에 전자상거래로 구매한 상품²⁾

적용된 상품중 도서의 경우 상품에 대한 정보로 책의 출판사 및 저자, 해당 책의 판매사에 대한 정보가 표현되었음을 확인했다. 또한 그 외에 도서/소설/국외소설이라는 분류 정보와 베스트셀러라는 부가유형정보, 책의 용지, 장수, 목차, 요약, 표지이미지, 판매가, 할인가 등의 정보가 표현되었다.

2) 통계청, 5, 2001 (단위:%, 만19세 이상 최근6개월 내 EC경험자)

```

<?xml version= "1.0" encoding= "euc-kr" ?>
<?xml:stylesheet type= "text/xsl" href= "book_old.xsl" ?>
<!--File Name : book_01.xml-->
<ProductCatalog>
  <CatalogHead>
    <Manufacturer>
      <Name>인디북</Name>
      <Address>주소 생략</Address>
      <E-mail>이메일 생략</E-mail>
      <Phone>전화번호 생략</Phone>
      <Fax>팩스 생략</Fax>
    </Manufacturer>
    <Language>Korean</Language>
    <Currency>원</Currency>
    <Country>Korea</Country>
  </CatalogHead>
  <CatalogBody>
    <ClassifierIdentity>
      <CategoryInfo CategoryStandard="분류 체계">
        <CategoryID>분류ID</CategoryID>
        <ParentCategoryRef>부모분류ID</ParentCategoryRef>
        <CategoryName>국외소설</CategoryName>
        <CategorySynonym>해외소설</CategorySynonym>
      </CategoryInfo>
    </ClassifierInfo>
    <IdentityInfo><Identifier>상품코드:9788989258384</Identifier>
    </IdentityInfo>
    <TypeInfo><AddedType>베스트 셀러</AddedType></TypeInfo>
  </ClassifierIdentity>
  <ProductInfo>
    <DefaultInfo>
      <ID>9788989258384</ID>
      <Name>톨스토이 단편선</Name>
      <Price>8,500</Price>
      <Manufacturer>인디북</Manufacturer>
      <ManufacturedDate>2003-03-01</ManufacturedDate>
      <EffectiveDate>2003-12-31</EffectiveDate>
      <Dimension UOM= "g" >490</Dimension>
      <Attachment>book_01.JPG</Attachment>
      <Characteristic>
        <Particular TypeCode= "Size" ParticularName= "Paper" >
          <Value>A6</Value>
        </Particular>
        <Particular TypeCode= "Size" UOM= "Page" ParticularName= "Page" >
          <Value>294</Value>
        </Particular>
        <Particular TypeCode= "Content" ParticularName= "Contents" >
          <Value>책목차 생략</Value>
        </Particular>
      </Characteristic>
    </DefaultInfo>
  </ProductInfo>
</CatalogBody>
</Catalog>

```



```

    <Particular TypeCode= "Content" ParticularName= "Summary" >
      <Value>책소개 생략</Value>
    </Particular>
  </Characteristic>
</DefaultInfo>
<SellingInfo>
  <SellingUnit>1권</SellingUnit>
  <SellingPrice>7,650</SellingPrice>
  <SellingStock>재고없음</SellingStock>
  <DeliveryPeriod UOM= "day" >2</DeliveryPeriod>
  <AssurancePeriod UOM= "day" >보증기간</AssurancePeriod>
  <Returning>반품가능</Returning>
  <OfferGoods>없음</OfferGoods>
  <OptionGoods>없음</OptionGoods>
</SellingInfo>
<ValueInfo>
  <Value Name= "the book-review" >
    <Description><LongDescription>서평 생략</LongDescription>
  </Description>
  </Value>
</ValueInfo>
</ProductInfo>
</CatalogBody>
</ProductCatalog>

```

[표 16] XBC에 도서상품을 적용한 예

book - Microsoft Internet Explorer

파일(F) 편집(E) 보기(V) 즐겨찾기(A) 도구(T) 도움말(H)

뒤로 - - x - - 검색 ☆ 즐겨찾기 미디어

주소(D) D:\₩공유폴더\₩xml_ppt\₩book_01.xml

제조사	인디북
주소	주소 생략
이메일	이메일생략
전화	전화번호 생략
팩스	팩스 생략
사용언어	Korean
사용통화	원
원산지	Korea

분류이름	국외소설
유사분류이름	해외소설
식별이름	상품코드:9788989258384
부가정보	베스트셀러

상품ID	9788989258384
상품명	톨스토이 단편선
가격	8,500
제조사	인디북
제조일	2003-03-01
유효일	2003-12-31
무게	490 (g)



[그림 26] XBC에 도서상품을 브라우저로 파싱한 예

컴퓨터의 경우 상품에 대한 정보로 제조사에 대한 정보가 자세하게 표현되었고, 컴퓨터&인터넷/컴퓨터/데스크탑의 분류정보가 표현되었고 생산 가격과 판매가격, 컴퓨터의 기본사양 및 설명, CPU, RAM, HDD, VGA, CD-ROM, Sound 등의 부품에 대한 정보가 표현되어 있다. 또한 컴퓨터의 외관사진이 포함되어 있다.

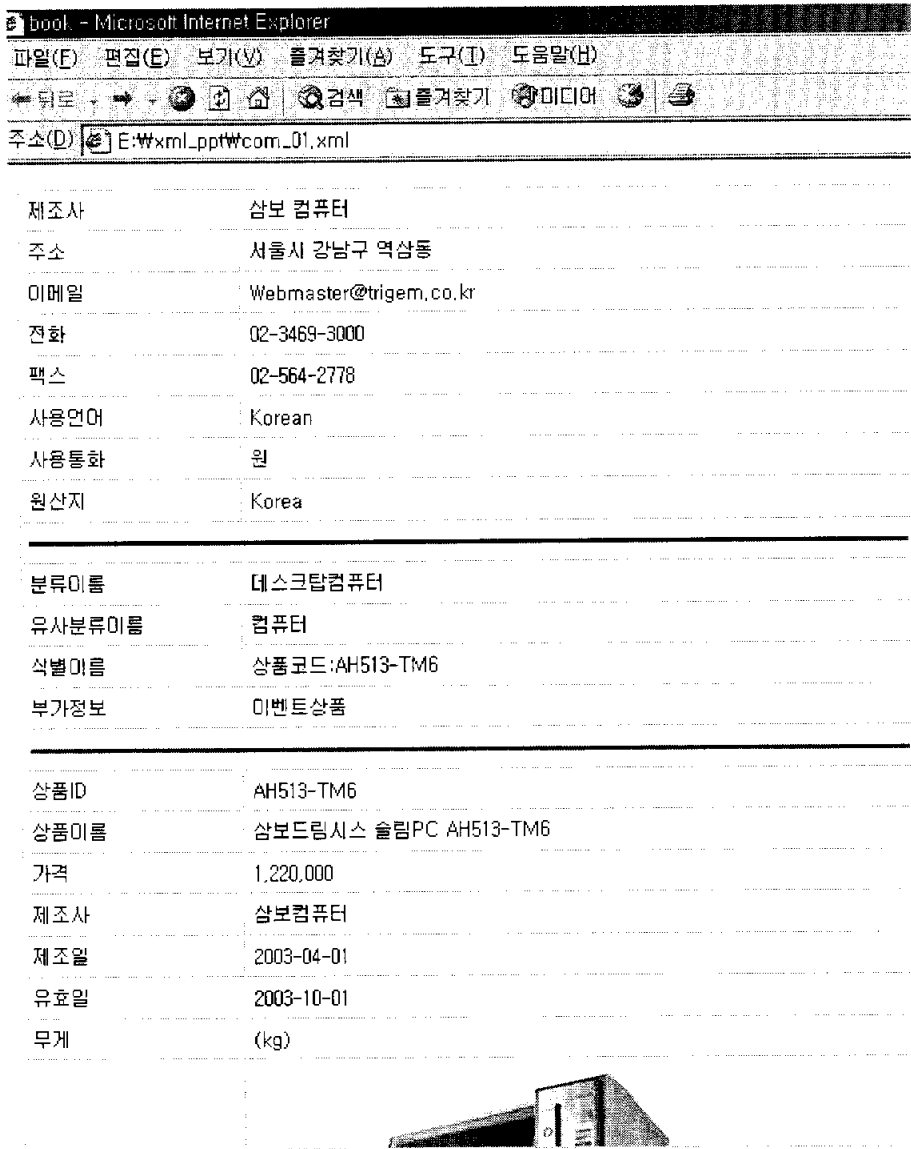
```
<?xml version= "1.0" encoding= "euc-kr" ?>
<?xml:stylesheet type= "text/xsl" href= "book_old.xsl" ?>
<ProductCatalog>
  <CatalogHead>
    <Manufacturer>
      <Name>삼보컴퓨터</Name>
      <Address>주소 생략</Address>
      <E-mail>이메일 생략</E-mail>
      <Phone>전화번호 생략</Phone>
      <Fax>팩스 생략</Fax>
    </Manufacturer>
    <Language>Korean</Language>
    <Currency>원</Currency>
    <Country>Korea</Country>
  </CatalogHead>
  <CatalogBody>
    <ClassifierIdentity>
      <CategoryInfo CategoryStandard="분류체계">
        <CategoryID>분류ID</CategoryID>
        <ParentCategoryRef>부모분류ID</ParentCategoryRef>
        <CategoryName>데스크탑컴퓨터</CategoryName>
        <CategorySynonym>컴퓨터</CategorySynonym>
      </CategoryInfo>
      <IdentityInfo>
        <Identifier>상품코드:AH513-TM6</Identifier>
      </IdentityInfo>
      <TypeInfo>
        <AddedType>이벤트상품</AddedType>
      </TypeInfo>
    </ClassifierIdentity>
    <ProductInfo>
      <DefaultInfo>
        <ID>AH513-TM6</ProductID>
        <Name>삼보드림시스 슬립PC AH513-TM6</ProductName>
        <Price>1,220,000</ProductPrice>
        <Manufacturer>삼보컴퓨터</Manufacturer>
        <ManufacturedDate></ManufacturedDate>
        <EffectiveDate></EffectiveDate>
```

```

<Dimension UOM= "kg" ></PhysicalDimension>
<Attachment>com_01.JPG</ProductAttachment>
<Characteristic>
<Particular TypeCode= "Performance" ParticularName= "CPU" >
  <Value>Intel Pentium IV 2.4Ghz</Value></Particular>
<Particular TypeCode= "Performance" ParticularName= "RAM" >
  <Value>256MB DDRRAM</Value></Particular>
<Particular TypeCode= "Performance" ParticularName= "Cache" >
  <Value>512KB</Value></Particular>
<Particular TypeCode= "Performance" ParticularName= "HDD" >
  <Value>60GB HDD(7200rpm)</Value></Particular>
<Particular TypeCode= "Performance" ParticularName= "VGA" >
  <Value>AGP 4x Max 64MB</Value></Particular>
<Particular TypeCode= "Performance" ParticularName= "ODD" >
  <Value>48배속 Combo</Value></Particular>
<Particular TypeCode= "Performance" ParticularName= "Sound" >
  <Value>PCI 32bit 3D On board</Value></Particular>
<Particular TypeCode= "Performance" ParticularName= "LAN" >
  <Value>10/100MB LAN</Value></Particular>
<Particular TypeCode= "Performance" ParticularName= "Mouse" >
  <Value>Optical Wheel</Value></Particular>
<Particular TypeCode= "Performance" ParticularName= "Keyboard" >
  <Value>멀티 컴팩트 키보드</Value></Particular>
<Particular TypeCode= "Performance" ParticularName= "OS" >
  <Value>Windows XP</Value></Particular>
</Characteristic>
</DefaultInfo>
<SellingInfo>
  <SellingUnit>1대</SellingUnit>
  <SellingPrice>1,098,000</SellingPrice>
  <SellingStock>재고없음</SellingStock>
  <DeliveryPeriod UOM= "day" >5</DeliveryPeriod>
  <AssurancePeriod UOM= "Year" >2</AssurancePeriod>
  <Returning>반품가능</Returning>
  <OfferGoods>없음</OfferGoods>
  <OptionGoods>없음</OptionGoods>
</SellingInfo>
</ProductInfo>
</CatalogBody>
</ProductCatalog>

```

[표 17] XBC에 컴퓨터상품을 적용한 예



[그림 27] XBC에 컴퓨터상품을 브라우저로 파싱한 예

적용된 결과는 기존의 방법들과는 달리 각 정보들을 분류하여 표현할 수 있었다. 각 정보는 성능중심, 형태중심, 내용중심의 상품들로 구분되었고, 상품의 분류정보는 계층적 구조를 가지며 표현되었다. 또한 제조사나 기획사, 도서의 저자, 출판사 등의 정보들에 대한 중복적 표현이 제거되었

음을 확인할 수 있었다.

실제로, 제안된 모델은 여타의 대부분의 모든 상품에 대해서 적용될 수 있다. 상품정보는 공통게시정보와 고유게시정보로 구분할 수 있는데, [그림 26]의 전자카탈로그에서 DefaultInfo의 Characteristic이 고유게시정보를 기술하고, 나머지 다른 모든 요소는 공통게시정보를 기술한다. 제안된 모델이 범용성을 가지려면, 모든 상품의 고유게시정보인 Characteristic상에 기술되어야 한다. 이것은 가능하다고 본다. 왜냐하면, 상품의 고유게시정보는 내용정보, 성능정보, 형태정보의 3가지 유형으로 구분될 수 있으며, 이러한 유형에 속한 속성은 Specification을 통해서 기술될 수 있기 때문이다. 또한 그 3가지 유형에 속하지 않은 속성을 갖는 상품이 추후에 만들어진다면, TypeCodeOther의 요소를 통해서 확장 기술될 수 있다.

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE "http://xml.cXML.org/schemas/cXML/1.1.009/cXML.dtd" >
<Supplier corporateURL="http://www.trigem.co.kr" storeFrontURL=
"http://shop.trigem.co.kr/tgshop/wwbs-bin/ip.front.top.FP_ShowMain" >
  <Name xml:lang="en-US">SAMBO Computer</Name>
  <Comments xml:lang="en-US">this is a cool company</Comments>
  <SupplierID domain="DUNS">000000001</SupplierID>
  <SupplierLocation>
    <Address>
      <Name xml:lang="KR">서울사무소</Name>
      <PostalAddress>서울시 강남구 역삼동</PostalAddress>
      <Email>Webmaster@trigem.co.kr</Email>
      <Phone name="Office">
        <TelephoneNumber>
          <CountryCode isoCountryCode="KR">82</CountryCode>
          <AreaOrCityCode>02</AreaOrCityCode>
          <Number>3469-3000</Number>
        </TelephoneNumber>
      </Phone>
      <Fax name="Order"><TelephoneNumber>
        <CountryCode isoCountryCode="KR">82</CountryCode>
        <AreaOrCityCode>02</AreaOrCityCode>
        <Number>564-2778</Number>
      </TelephoneNumber></Fax>
      <URL>http://www.trigem.co.kr</URL>
    </Address>
    <OrderMethods><OrderMethod><OrderTarget>
```

```

        <URL>http://shop.trigem.co.kr/tgshop/wwbs-bin/
        ip.front.cart.FP_OrderList</URL>
    </OrderTarget></OrderMethod><Contact>
        <Name xml:lang= "KR" >이지영</Name>
        <PostalAddress>서울시 강남구 역삼동</PostalAddress>
        <Email>Leejy@trigem.co.kr</Email>
        <Phone name= "Office" >02-3469-3000</Phone>
    </Contact></OrderMethods>
</SupplierLocation>
</Supplier>
<Index>
    <SupplierID domain= "InternalSupplierID" >01</SupplierID>
    <Comments xml:lang= "KR" >삼보 컴퓨터 상품입니다</Comments>
    <SearchGroup>
        <Name xml:lang= "en" >Desktop Computers</Name>
        <SearchAttribute name= "Size" type= "string" />
        <SearchAttribute name= "CPU" type= "string" />
        <SearchAttribute name= "RAM" type= "number" />
    </SearchGroup>
    <IndexItem>
        <IndexItemAdd>
            <ItemID><SupplierPartID>AH513-TM6</SupplierPartID></ItemID>
            <ItemDetail>
                <UnitPrice><Money currency=KR>1,220,000</Money></UnitPrice>
                <Description xml:lang= "KR" >삼보 드림시스 Pentium IV 2.4G 60GB
                </Description>
                <UnitOfMeasure>EA</UnitOfMeasure>
                <Classification domain= "UNSPSC" >C43171803</Classification>
                <ManufacturerPartID>COM_SAMBO</ManufacturerPartID>
                <ManufacturerName>삼보 컴퓨터</ManufacturerName>
                <URL>http://www.trigem.co.kr</URL>
            </ItemDetail>
            <IndexItemDetail>
                <LeadTime>10</LeadTime>
                <ExpirationDate>2003-04-01</ExpirationDate>
                <EffectiveDate>2003-10-01</EffectiveDate>
                <SearchGroupData>
                    <Name xml:lang= "en" >Desktop Computers</Name>
                    <SearchDataElement name= "Size" type= "SLIM" />
                    <SearchDataElement name= "CPU" type= "Pentium 4 " />
                    <SearchDataElement name= "RAM" type= "256" />
                </SearchGroupData>
                <TerritoryAvailable>KOREA</TerritoryAvailable>
            </IndexItemDetail>
        </IndexItemAdd>
    </IndexItem>
</Index>

```

[표 18] cXML에 컴퓨터 상품을 적용한 예

book - Microsoft Internet Explorer

파일(F) 편집(E) 보기(V) 즐겨찾기(A) 도구(T) 도움말(H)

뒤로 → → → 검색 즐겨찾기 미디어

주소(D) E:\xml_ppt\wcXML_com_01.xml

공급사(SupplierID)	SAMBO Computer
Comment	this is a cool company
업체번호	000000001
공급사 지사명	서울사무소
주소	서울시 강남구 역삼동
메일	Webmaster@trigem.co.kr
전화	82 - 02 - 3469 - 3000
팩스	82 - - 564 - 2778
홈페이지	http://www.trigem.co.kr
주문방법	http://shop.trigem.co.kr/tgshop/wwwbs-bin/ip.front.cart.FP_OrderList
담당자	이지영
주소	서울시 강남구 역삼동
메일	Leejy@trigem.co.kr
전화	02-3469-3000

공급사(InternalSupplierID)	01
Comments	삼보 컴퓨터 상품입니다
제품명(모델명)	AH513-TM6
제품가격	1,220,000
제품설명	삼보 드림시스 Intel Pentium IV 2.4Ghz 60GB
제품단위	EA
제품식별	C43171803
제조사ID	COM_SAMBO

[그림 28] cXML에 컴퓨터상품을 브라우저로 파싱한 예


```

<?xml version= "1.0" encoding= "euc-kr" ?>
<!--File Name : xCBL_컴퓨터.xml-->
<ProductCatalog>
<CatalogHeader>
  <CatalogID>SAMBO Computer Catalog</CatalogID>
  <CatalogDate>20000816</CatalogDate>
  <CatalogProvider>
    <Party>
      <PartyID>
        <Identifier>
          <Agency>
            <AgencyCoded>Other</AgencyCoded>
            <AgencyCodedOther>CommerceOne</AgencyCodedOther>
          </Agency>
          <Ident>SAMBO Computer</Ident>
        </Identifier>
      </PartyID>
    </Party>
  </CatalogProvider>
  <ListOfPartners>
    <Partner PartnerID= "SAMBO_Computer" Relationship= "Buyer" />
  </ListOfPartners>
  <ValidFrom>2003-04-01</ValidFrom>
  <ValidUntil>2003-10-01</ValidUntil>
  <CatalogVersion>1.0</CatalogVersion>
  <DefaultLanguage xml:lang= "KR" />
  <DefaultCurrency>
    <Currency>
      <CurrencyCoded>WON</CurrencyCoded>
    </Currency>
  </DefaultCurrency>
  <IsMultiVendor/>
  <ShortDescription xml:lang= "en" >SAMBO Computer Catalog
  </ShortDescription>
</CatalogHeader>
<CatalogSchema>
  <SchemaName>UNSPSC Subset for Computers</SchemaName>
  <SchemaVersion>1.0</SchemaVersion>
  <SchemaStandard>UNSPSC</SchemaStandard>
  <SchemaCategory CategoryID= "C43000000" >
    <CategoryName>Communications, Computer Equipment, Peripherals,
      Components and Supplies</CategoryName>
  </SchemaCategory>
  <SchemaCategory CategoryID=C43170000 ParentCategoryRef=C43000000>
    <CategoryName>Hardware and accessories</CategoryName>
  </SchemaCategory>

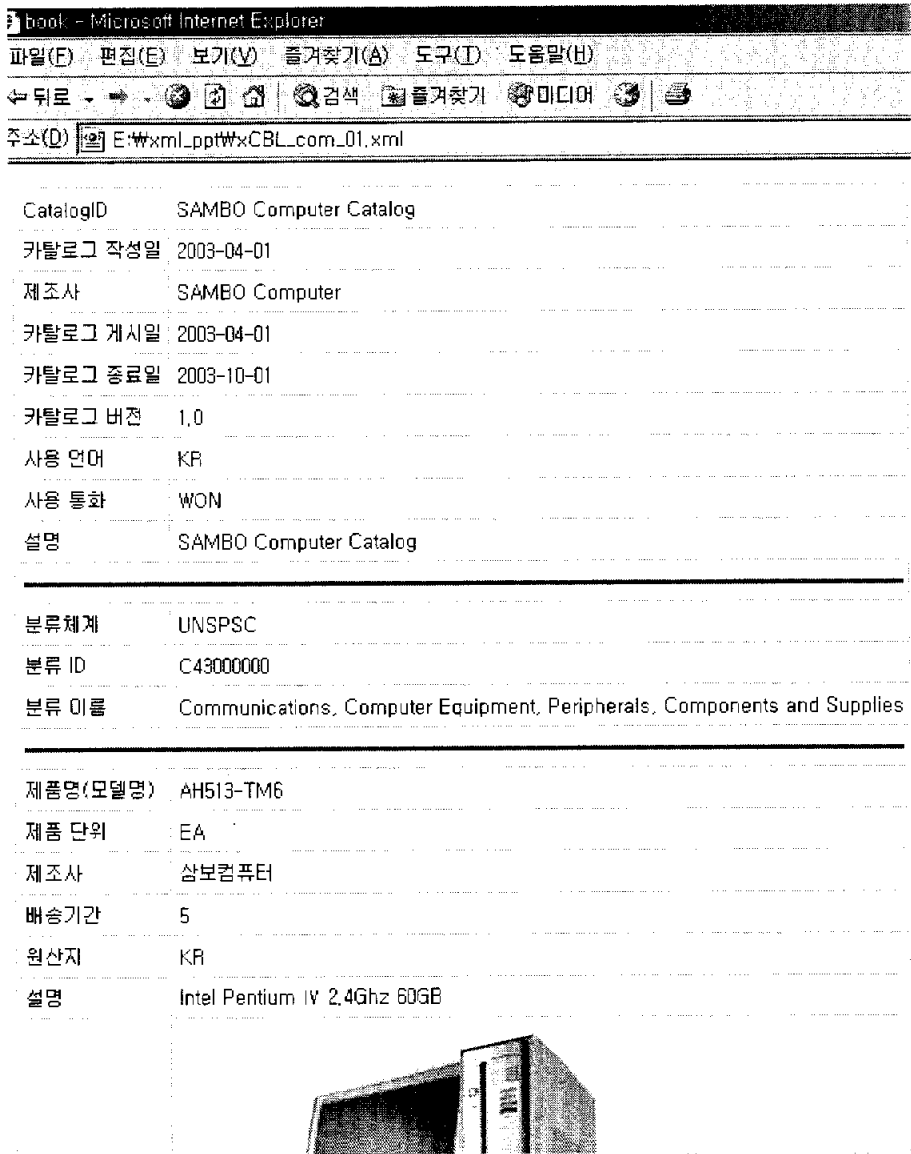
```

```

<SchemaCategory CategoryID=C43171800 ParentCategoryRef=C43170000>
  <CategoryName>Computers</CategoryName>
</SchemaCategory>
<SchemaCategory CategoryID=C43171803 ParentCategoryRef=C43171800>
  <CategoryName>Workstations or desktop computers</CategoryName>
</SchemaCategory>
</CatalogSchema>
<CatalogData>
<Product Type= "Good" SchemaCategoryRef= "C43171803" >
  <ProductID>AH513-TM6</ProductID>
  <UOM><UOMCoded>EA</UOMCoded></UOM>
  <Manufacturer>삼보 컴퓨터</Manufacturer>
  <LeadTime>5</LeadTime>
  <CountryOfOrigin>
    <Country><CountryCoded>KR</CountryCoded></Country>
  </CountryOfOrigin>
  <ShortDescription xml:lang= "KR" >Pentium IV 2.4G 60GB
</ShortDescription>
  <ProductAttachment>
    <AttachmentURL>com_01.JPG</AttachmentURL>
    <AttachmentPurpose>PicName</AttachmentPurpose>
    <AttachmentMIMEType>jpg</AttachmentMIMEType>
  </ProductAttachment>
  <ObjectAttribute><AttributeID>CPU</AttributeID>
    <AttributeValue>Pentium IV 2.4Ghz</AttributeValue></ObjectAttribute>
  <ObjectAttribute><AttributeID>RAM</AttributeID>
    <AttributeValue>256MB DDRAM</AttributeValue></ObjectAttribute>
  <ObjectAttribute><AttributeID>HDD</AttributeID>
    <AttributeValue>60GB HDD</AttributeValue></ObjectAttribute>
  <ObjectAttribute><AttributeID>VIDEO</AttributeID>
    <AttributeValue>AGP 4xMax 64MB</AttributeValue></ObjectAttribute>
  <ObjectAttribute><AttributeID>ODD</AttributeID>
    <AttributeValue>48배속 Combo</AttributeValue></ObjectAttribute>
  <ObjectAttribute><AttributeID>SOUND</AttributeID>
    <AttributeValue>PCI 32bit On board</AttributeValue></ObjectAttribute>
  <ObjectAttribute><AttributeID>LAN</AttributeID>
    <AttributeValue>10/100MB LAN</AttributeValue></ObjectAttribute>
  <ObjectAttribute><AttributeID>Mouse</AttributeID>
    <AttributeValue>Optical Wheel</AttributeValue></ObjectAttribute>
  <ObjectAttribute><AttributeID>Keyboard</AttributeID>
    <AttributeValue>멀티 컴팩트 키보드</AttributeValue></ObjectAttribute>
  <ObjectAttribute><AttributeID>OS</AttributeID>
    <AttributeValue>Windows XP</AttributeValue></ObjectAttribute>
</Product>
</CatalogData>
</ProductCatalog>

```

[표 19] xCBL에 컴퓨터상품을 적용한 예



[그림 29] xCBL에 컴퓨터상품을 브라우저에 파싱한 예

		cXML	xCBL	XBC
제 품 정 보	제품명	○	○	○
	모델명	○	○	○
	제품생산일			○
	제품의 단위명칭	○	○	○
	제품수량			○
	제품제조원	○	○	○
	판매원	○	○	○
	원산지		○	○
	제품 외관정보		○	○
	제품 기능정보		○	○
	세트상품, 별도구매			○
	이미지파일 및 설명		○	○
	취급주의 사항			
가 격 정 보	제품가격	○	○	○
	가격표시통화	○	○	○
	세금내역			
	가격책정수량	○	○	○
	대금지급조건 및 방법			
	대금지급시기			
업 체 정 보	소비자의 추가비용			
	사업자 명칭	○		○
	점포명			
	업체의 국적	○		○
	소재지	○		○
	성명	○		
	전화번호	○		○
	팩스	○		○
판 매 정 보	E-mail	○		○
	거래제한			
	교환조건			○
	환불/반품조건			○
	품질보증마크			
	주문방법	○		
	영업일시	○	○	○
	판매지역	○		
	판매수량의 제한		○	
	사은품 유무			○

[표 20] cXML · xCBL · XBC 게시 권고안 표

제안된 XBC 모델에 컴퓨터와 도서 상품을 기술하는데 어려움이 없었으며, 각 상품의 속성 모두 효과적으로 기술되었다. XBC 모델이 표준으로 지정된 다른 전자카탈로그보다 효과적으로 기술된다는 것을 보여주기 위해서 4장에서 살펴본 표준으로 지정된 전자카탈로그에 컴퓨터와 도서 상품을 적용하여 한국전산원에서 제시한 “전자카탈로그의 게시항목 권고안”을 얼마만큼 만족하는지를 살펴보기로 한다. 위의 [표 20]에서는 cXML · xCBL · XBC가 제시 권고안을 만족하는지를 표시하였다. Rosetta Net은 전자부품에 한정되어 상세히 기술되기 때문에 비교 대상에서는 제외한다.

[표 21]은 cXML · xCBL · XBC 게시 권고안 비교표이다. 총 37개의 항목 중 cXML은 18개, xCBL은 13개, XBC는 25개 항목을 만족하여 각각 48%, 35%, 68%로 XBC가 가장 높게 나타나고 있고 있다. 자세히 살펴보면 13개의 제품정보, 7개의 가격정보, 8개의 업체정보, 9개의 판매정보로 나누어져 있다. XBC는 제품정보에서 판매정보까지 골고루 항목을 표시하고 있어 다른 전자카탈로그보다 효과적으로 기술할 수 있다.

	cXML	xCBL	XBC
제품정보(13)	5	9	12
가격정보(7)	3	3	3
업체정보(8)	7	0	6
판매정보(9)	3	1	3
총 (37)	18	13	25
	48%	35%	68%

[표 21] cXML · xCBL · XBC 게시 권고안 비교표

제 VII 장 결 론

본 논문에서 제시한 전자카탈로그는 전자카탈로그 원형에 기반하여 cXML, xCBL과 RosettaNet의 장점을 도입하고 각각의 단점을 보완하여 제안되었기 때문에 cXML, xCBL과 RosettaNet의 기본적인 기능을 모두 수용하고 있다. cXML, xCBL과 RosettaNet의 상품모델과 비교하여 다음과 같은 4가지의 독창적인 특징을 제공한다.

1. 상품정보의 분류 표현.
2. 분류된 상품정보의 계층적 표현.
3. 다양한 상품의 세부적인 표현.
4. 상품정보의 중복적 표현 회피.

첫째, 상품정보의 분류 표현은 [그림 18]의 XBC 모델 원형에서 보는 것과 같이 분류정보, 식별정보, 유형정보, 기본정보, 판매정보, 부가정보로 분류하여 표현한다. 특히, 기본정보에서 내용정보, 성능정보, 형태정보로 나누어 분류하는 것은 다른 표준에서 찾아볼 수 없는 특징이다. cXML은 크게 Supplier(공급자), Index(상품), Contract(주문계약)으로 분류를 한다. 상품정보를 나타내는 IndexItemAdd 엘리먼트인 [그림 2]를 보면 상품ID를 가지는 ItemID, 상품의 상세 정보를 가지는 ItemDetail, 추가정보를 가지는 IndexItemDetail로 분류하고 있다. xCBL은 CatalogHeader(카탈로그 관리 정보), CatalogSchema(분류체계), CatalogData(상품정보)로 분류하고 있다.

둘째, 분류된 상품정보의 계층적 표현은 RosettaNet의 특징중 하나인 요소의 재귀적 사용의 도입으로 상품의 정보를 계층화하여 나타낼 수 있다. Component 엘리먼트가 이 특징을 나타내고 있다. 예를 들어 등록하는 상품이 컴퓨터일때 컴퓨터는 마더보드라는 부품을 포함하며, 이 마더보드는 또 다른 메모리 슬롯의 부품을 갖는다. 이와 같은 방식은 상품특성을

단계적으로 세분화시키면서, 각 부품의 특성과 부품간의 연관관계를 상세히 기술할 수 있도록 해준다.

셋째, 분류정보와 기본정보의 내용정보, 성능정보, 형태정보로 나타냄으로써 다양한 상품의 세부적인 표현은 해결할 수 있다. 분류정보는 [그림 20]에서 보는 것과 같이 자신의 분류 이름과 별칭으로 부르는 분류이름을 포함하고 있다. 하위 분류체계가 있으면 하위 분류체계에 상위 분류체계의 ID를 가입함으로써 상/하 관계를 알 수 있기 때문에 다양한 상품의 등록이 가능하고 검색에서도 편리하다. [그림 22]의 Characteristic 엘리먼트는 기본정보의 내용정보, 성능정보, 형태정보를 기술하는 부분으로 서술식의 설명이나 여러 부품이나 기능들을 나열하여 나타낼 때 사용할 수 있다.

넷째, 상품목록의 중복적 표현을 회피하기 위해 [표 22]와 같이 전자카탈로그를 CatalogHead와 CatalogBody 부분으로 나누어 공통게시정보를 CatalogHead에 기술하도록 하고 고유게시정보를 CatalogBody 부분에 기술하도록 했다. 이는 각 상품의 정보 기술시 상품간 공통정보의 중복 표현을 회피할 수 있도록 해준다. 다음의 [표 22]에서 보는 것과 같이 중복적인 부분은 제조사부분을 따로 XML 파일로 만들어 인크루드할 수 있도록 하였다.

<pre> <Manufacturer> <Name>인디북</Name> <Address>주소 생략</Address> <E-mail>이메일 생략</E-mail> <Phone>전화번호 생략</Phone> <Fax>팩스 생략</Fax> </Manufacturer> </pre>
<pre> <Manufacturer xmlns:xi= "http://www.w3.org/2001/XInclude" > <xi:include href= "header.xml" /> </Manufacturer> </pre>

[표 22] 중복적 표현 제거 CatalogHead

이렇게 제시된 XBC 모델과 기존 연구들이 한국전산원에서 제시한 게시항목을 만족하는지를 하는지를 비교하여 XBC 모델이 다른 모델에 비해 더 많은 항목을 만족하고 있으며, 그 비중도 제품정보, 가격정보, 업체정

보, 판매정보 어느 한곳에 치중하지 않는다.

본 논문에서는 전자상거래에서 통용 가능한 대부분의 모든 상품정보를 통합하여 효과적으로 기술할 수 있는 상품 표현 모델, 전자카탈로그를 제안하고, 이 모델을 2개의 상품에 대해서 적용하였다. 적용결과는 단일 모델로 각 상품정보를 기술하는데 있어서 어려움이 없었다. 그러나 더 많은 상품에 적용해야 하며 제안된 모델이 효용성을 갖기 위해서는 상품의 고유 속성 명칭을 나타내는 ParticularName을 상품 혹은 상품 군별로 코드화, 즉 표준화해야 한다. 이러한 작업은 산업체에서 제품군 별로 컨소시엄을 구성하여 이루어져야 할 것이다. 수직-마켓(Vertical-market)에 기반을 둔 프레임워크를 개발하고 있는 RosettaNet은 전자/반도체 상품에 대해서 3,500개의 고유 속성 코드를 제공하고 있다. 향후 연구 과제로 본 논문에서 선택한 상품 외에 더 많은 상품에 적응시켜 실험할 필요가 있을 것이다. 위에서 설명한 상품의 코드화 역시 차후 연구되어야 할 과제이다.

[참 고 문 헌]

< 국 내 문 헌 >

- 강재구, “BtoB 전자상거래 시스템을 위한 XML/EDI 구축 방안에 대한 연구”, 세종대학교 정보통신대학원, 1999.
- 김선호외 5인, “전자카탈로그 표준화 동향”, 한국전자거래(CALS EC)학회, 한국전자거래(CALS/EC)학회지, Vol.6, No.1, 2001.
- 문희철, “전자상거래 표준 관련동향과 발전방향”, 한국인터넷전자상거래학회 2001년 추계학술 및 정책 세미나 논문집, 한국인터넷전자상거래학회, 10, 2001.
- 서의호, 이태웅, “전자상거래 표준화 기술로서의 XML”, 한국경영정보학회, 1999 추계국제학술대회 논문집, 1999.
- 엄선홍, “전자상거래에서 XML 기반 Catalog 어플리케이션 개발”, 고려대학교 대학원, 2001.
- 이상구, “전자카탈로그의 기술 표준화 방향”, 한국전자거래(CALS EC)학회, e - BIZ World Conference 2001(CALS Korea 2001), 2001.
- 이승호, “XML 기반의 전자카탈로그 구현 사례”, 정보처리학회지, 한국정보처리학회, 2001.
- 전자상거래 표준화 통합 포럼, “전자카탈로그 표준현황 및 적용지침”, 제1권, 04, 2001.
- 전자상거래 표준화 통합 포럼, “상품속성정보 게시 및 표준현황 및 적용지침”, 제1권, 04, 2001.
- 전자상거래 표준화 통합 포럼, “XML기반의 전자카탈로그 리포지토리 구축 기술”, 2001.
- 조일제외 5인, “XML/EDI 시스템의 설계 및 구현”, 한국 정보처리학회 춘계 학술 발표 논문집, 제7권, 제1호, 2000.

홍언주, “전자카탈로그 관련 기술 및 사업의 현황분석과 개선방안”,
한국전산원, 1999.

한국전산원, “CALS/EC 표준화 로드맵 연구”, 한국전산원, 6, 1998.

< 국 외 문 헌 >

- Corey Haines, “Creating summary Report in XML and XSLT” ,
<http://asptoday.com>, 09, 2000.
- Crarg Murphy, “Creating Application Using SOAP and XML HTTP” ,
<http://asptoday.com>, 06, 2000.
- Didier Martin, *PROFESSION XML*, Wrox Press Ltd.2000.
- Don Box “A Young Person’s Guide to The Simple Object Access
Protools: SOAP Increase Interoperability Across Platforms and
Languages” , <http://msdn.microsoft.com>, 03, 2000.
- Frank Bounpher with coworkers, *PROFESSIONAL XML
APPLICATION* Wrox Press Lid, 1999.
- Hans-W.Gelleraen and Martin Gaedke, “OBJECT ORIENTED WEB
APPLICATON DEVELOPMENT” IEEE INTERNET
COMPUTING, 01.02, 1999.
- Hayes, L. B., “ Pace is in the Business of Style” , *Catalog Age*,
Vol.10, No.9, 1993, pp.134-136.
- Hoffman, D. L., T. P. Novak, and P. Chatterjee, “Commercial
Scenarios for the web: Opportunities and Challenges” , *Journaml
of Computer-Madiated Communication*, Vol.1, No.3, 1996,
<http://www.ascusc.org/jcmc/voll/issue3/huffman.html>.
- Huizingh, E. K. R. E., “The Content and design of Web Sites : An
Empirical Study” , *Information & Management*, Vol.37, 2000,
pp.123-134.
- James Q. stansfield, “Using XMLHTTP To Gather And Display News
Headlines From The Web” , <http://www.asptoday.com>,10,2000.
- Keller, A. M., “Smart Cataloged and Virtual Catalogs” , in Readings
in Electronic Commerce, edited by Kalakota, R., and A. B.
Whinston, 1997, pp2.59-271.

- Koenemann, J., and M. Stolze, "CHI 99 SIG : User Interfaces for Electronic Product Catalogs" , *SIGCHI Bulletin*, Vol.31, NO.4, 1999, pp.47-48.
- Lewis, H. G., *Direct Marketing Strategies and Tactics*, Dartnell Corporation, 1992.
- Liu, Chang, Kirk P. Arnett, Louis M. Capella, and Robert C. Beatty, "Web Sites of the Fortune 500 Companies : Facing Customers through Home Pages" , *Information & Management*, Vol.31, 1997, pp.335-345.
- Mark Baartse with coworkers, *Professional ASP XML*, Wrox Press Lid, 2000.
- Mark Wilson and Tracey Wilson, "XML Program with VB and ASP" , manning, 2000.
- Phal Sandler, "Site News Application Using ASP and Server Side XML" , <http://www.asptoday.com>, 08, 2000.
- Poon, S. and P. M. C. Swatman, "A Combined-Method Study of Small Business Internet Commerce" , *International Journal of Electronic Commerce*, Vol.2, No.3, 1998, pp.31-46.
- Robert Thibadean, ph.D.Lange with coworkers, "E-Commerce Catalog Construction" , *D-Lib Magazine*, Vol5, ISSUE2, 02, 1999.
- Shasho-Jones, G., "Cover Story : A Step-by-Step Guide to Designing Effective Catalog Covers," *Catalog Age*, Vol.8, No.11, 1991, pp.107-108.
- Spiller, Peter, and Gerald L. Lohse, "A Classification of Internet Retail Stores" , *International Journal of Electronic Commerce*, Vol.2, No.2, 1997-8, pp29-56.
- Vadapalli, Anand and K. Ramamurthy, "Business Use of the Internet : An Analytical Framework and Exploratory Case Study" ,

International Journal of Electronic Commerce, Vol.2, No.2,
1997-8, pp.71-94.

Wan, H. A., “Opportunities to enhance a commercial website” ,
Information & Management, Vol.38, 2000, pp.15-21.

RosettaNet, <http://www.rosettanet.org>

Extensible Markup Language(XML), <http://www.W3.org/xml>

xCBL, <http://www.xcbl.org/xcbl35/xcbl35.htm>

감사의 글

돌아보면 지난 2년 간의 석사과정의 시간이 너무 짧게만 느껴지고 무사히 마칠 수 있었던 것은 많은 분들의 사랑과 관심이 있었기 때문입니다.

먼저 지도교수님이신 김하균 교수님께 감사하다는 말을 전해드리고 싶습니다. 지도학생으로 받아 주셨고 부족한 저의 논문을 완성할 수 있게 지도와 조언을 해주신 교수님께 다시 한번 더 진심으로 감사 드립니다.

본 논문이 완성되기까지 많은 조언으로 심사해 주신 이현규 교수님, 김영봉 교수님 그리고 학업의 전진에 도움을 주신 이재정 교수님, 항상 논문에 관심을 많이 가져 주신 배상욱 교수님 감사 드립니다. 그리고 격려해 주신 모든 정보시스템협동과정 교수님들께도 감사드립니다.

“중간고사 기말고사는 잘 쳤나?” 라고 물어보는 영식선배, “포장을 잘 해라지” 라고 늘 말하는 한진선배, 선배님들의 격려에 고마운 마음을 전합니다. 메신저로 “교수님 계시냐?” 라고 물어보는 윤정선배, 밥 먹을 때면 전화하는 혜선선배, 가끔 엉뚱한 모습을 보여주는 동엽선배, “원샷노브레이크” 를 외치는 정인형, 일본에서 열심히 공부하고 있는 주형이, 조교업무와 논문 쓴다고 바쁜 봉원씨, 지원씨, 앞으로 밝은 모습을 볼 수 있을 성태씨, 이외 후배들에게 모두 고마움을 전합니다.

나의 영원한 친구 용식이 워드 친다고 고생했다. 사업한다고 바쁜 현주형, 베필멤버인 체영이형, 세진형, 몸과 마음을 바쳐 충성을 다 한 선영이에게도 고마움을 전합니다.

무엇보다 어려움 속에서도 학업을 마칠 수 있도록 사랑과 이해로 돌보아주시고 자식을 믿어주신 부모님께 더 더욱 감사 드립니다. 부모님의 헌신적인 노력으로 인해 학업을 마무리 지을 수 있었습니다. 그리고 서울에서 격려한 누나와 매형에게도 감사 드립니다.

많은 분들의 도움으로 작은 결실을 맺게 되었음을 항상 기억하고 앞으로 더 나은 미래를 위해 노력하여 사회에 도움이 되는 사람이 되겠습니다.

2003년 8월

김 성 재