

인터넷 문서의 새로운 패러다임

XML의 기본 철학은 문서의 내용(Contents), 구조(Structure), 프리젠테이션(Presentation)의 분리이다. 즉, 웹 문서 및 일반적인 모든 문서는 문서 구조, 내용, 출판 포맷으로 이루어 지는데, 문서 구조 및 내용을 텍스트로 처리하고 태그를 사용하여 구조와 내용을 구분한다. 이러한 문서구조는 W3C에서 제안한 XML 규격 1.0 문법 형식을 따라야 한다.

■ 최한석/ (주)한국지식웨어 R&D연구소 소장.

XML 탄생

21세기 새로운 천년의 시작은 단순히 한 해가 바뀌는 것이 아니라, 지식사회(Knowledge Society)라는 새로운 사회구조 패러다임 구축을 요구하고 있다. 1989년 Tim Berners Lee 가 스위스 CERN 고차원 에너지 물리학 연구소에서 전세계의 흩어져 있는 연구원들의 연구결과 및 아이디어를 공유하기 위하여 시작한 인터넷 기반의 WWW 프로젝트를 소개한 이후로, 현재 전세계의 모든 비즈니스 및 일상 업무를 인터넷을 배제하고는 생활할 수 없을 정도로 개인 및 조직 가까이에서 인터넷이 활용되고 있다. 즉, 새로운 천년의 사회 인프라 구조로서 인터넷 환경의 지식공유 사회가 이미 형성되었음을 알 수 있다.

인터넷에서 웹문서를 생성하는 기본 형식은 HTML(Hyper Text Markup Language)이다. HTML 문서는 단일 문서유형 구조(DTD)에 하이퍼링크 기능을 추가한 하이퍼 텍스트 문서로서, 프로그램 하기가 쉽고 언어 자체를 쉽게 배울 수 있을 뿐만 아니라, 여러가지 웹 저작 도구들이 있어 사용자들이 쉽게 웹 문서를 생성할 수 있다.

그러나 HTML은 매우 제한적인 포맷언어로서 다양한 모든 문서의 유형 구조를 표현할 수 있는 SGML(Standard Generalized Markup Language)에서 파생된 단순 마크업 언어이다. 실제로 웹상에서 생성되고 유통되는 다양한 문서

들은 HTML 한가지 형식으로 표현하기는 많은 제약이 존재한다. 그렇다고 복잡한 문법 구조 및 스타일 형식을 제공하는 SGML을 웹 문서의 표준으로 채택할 수는 없었다.

따라서 1996년 W3C(World Wide Web 컨소시엄)에서는 사용자가 정의할 수 있는 구조화 문서 형식을 지원할 수 있고 웹 상에서 문서 유통 및 브라우징이 가능한 새로운 XML(eXtensible Markup Language) 마크업 언어를 제안하였다.

W3C XML 워킹그룹 Part 1에서는 XML 언어 규격 (<http://www.w3.org/TR/1998/REC-xml-19980210.xml>), Part 2에서는 XML 링크 언어(XLL)규격 (<http://www.w3.org/TR/1998/WD-xlink-19980303>), (<http://www.w3.org/TR/1998/WD-xptr-19980303>), Part 3에서는 XML 스타일시트 언어(XSL)규격 (<http://www.w3.org/TR/NOTE-XSL>)을 정의하였다. XML의 탄생은 기존의 HTML 제한사항을 극복하고 SGML의 복잡성을 최소화 하면서 다양한 하이퍼링크 및 스타일시트를 제공하여 웹문서의 새로운 패러다임을 창조하고 있다.

특히, XML이 인터넷에서 전자상거래, 디지털 도서관, 멀티미디어 등의 다양한 인터넷 응용과 접목되면서 새로운 천년의 웹 문서 표준으로 자리 매김을 하고 있다.

XML 이해

1) XML 문서

XML의 기본 철학은 문서의 내용(Contents), 구조(Structure), 프리젠테이션(Presentation)의 분리이다. 즉, 웹 문서 및 일반적인 모든 문서는 문서 구조, 내용, 출판 포맷으로 이루어 지는데, 문서 구조 및 내용을 텍스트로 처리하고 태그를 사용하여 구조와 내용을 구분한다. 이러한 문서 구조는 W3C에서 제안한 XML 규격 1.0 문법 형식을 따라야 한다.

XML 문서는 DTD가 선언된 Valid 문서와 DTD가 선언되어 있지 않지만 XML 규격에 따르는 Well-Formed 문서 두 가지로 구분된다. 일반적으로 XML문서를 만드는 방법은 DTD와 XML 인스턴스를 각각의 파일로 두는 방법(그림 1), DTD와 XML 인스턴스를 하나의 파일로 두는 방법(그림 2), DTD가 없는 Well-Formed XML 문서(그림 3)로 정의하는 세 가지 방법이 있다.

```
<?XML version="1.0" encoding="UTF-8" RMD='INTERNAL' ?>
<!DOCTYPE greeting SYSTEM "hello.dtd">
<greeting>Hello, world!</greeting>
```

〈그림 1〉 DTD와 인스턴스를 각각의 파일로 두는 방법

```
<?XML version="1.0" encoding="UTF-8" RMD='INTERNAL' ?>
<!DOCTYPE greeting [
<ELEMENT greeting(#PCDATA)>
]>
<greeting>Hello, world!</greeting>
```

〈그림 2〉 DTD와 인스턴스를 하나의 파일로 두는 방법

```
<?XML version="1.0" encoding="UTF-8" RMD='ONE' ?>
<greeting>Hello, world!</greeting>
```

〈그림 3〉 Well-Formed XML문서를 정의하는 방법

2) XML 문법

XML 문서는 W3C에서 제안한 XML Specification 1.0 규격에 따라 작성되어야 한다. XML 규격 1.0은 DTD 및 XML 인스턴스 생성 규칙을 BCNF 형식으로 제안하고 있다. XML DTD는 엘리먼트(Element), 애트리뷰트(Attribute), 엔티티

(Entity), 표기(Notation), 주석(Comments), 처리 명령어(Processing Instruction), CDATA 섹션 등으로 구성된다.

XML DTD에서 엘리먼트 선언부는 엘리먼트형과 엘리먼트의 내용 또는 서브 엘리먼트를 정의하는 것으로서 DTD의 기본 형식이다. 애트리뷰트 선언은 XML문서에서 나타나는 엘리먼트의 속성을 정의하는 것으로서 관계된 엘리먼트 형, 엘리먼트 이름, 데이터 형, 디폴트 값을 기술한다. 〈그림 4〉는 XML DTD의 엘리먼트 및 관계된 엘리먼트의 속성 선언 예이다.

〈!-XML 문서의 DTD->

```
<!ELEMENT Company (Intro+, IntroProduct?)>
<!ELEMENT Intro (Name, Address, (Phone|Fax|Homepage)*)>
<!ATTLIST Intro
ComID ID #IMPLIED
<!ELEMENT Name (#PCDATA)>
<!ELEMENT Phone (#PCDATA)>
<!ELEMENT Fax (#PCDATA)>
<!ELEMENT Homepage (#PCDATA)>
<!ELEMENT Address (Post, Addr)>
<!ELEMENT Post (#PCDATA)>
<!ELEMENT Addr (#PCDATA)>
<!ELEMENT IntroProduct (Product)*>
<!ELEMENT Product (ProductName+)>
<!ATTLIST Product
Number CDATA #IMPLIED
<!ELEMENT ProductName (#PCDATA)>
<!ATTLIST ProductName
ToolID ID #IMPLIED
Ver CDATA #IMPLIED
```

〈그림 4〉 DTD에서 엘리먼트 및 속성 선언 예

엘리먼트를 선언할 때 사용하는 발생지시자(Occurrence Indicator)는 +(한번이상 발생할 수 있음),?(한번 발생할 수도 있고 안 할수도 있음),*(0번이상 발생할 수 있음)로 구성된다. 이들에 대한 사용에는 〈그림 4〉의 엘리먼트 선언부에서 보여준다.

XML DTD에서 엔티티 선언은 문서에서 여러번 반복하여 입력할 필요가 있을 때 문서에서 참조할 수 있는 엔티티를 정의하여 사용할 수 있다. XML 인스턴스 자체에서 사용하는 텍스트 엔티티와 DTD에서만 사용되는 매개변수 엔티티가

존재한다. <그림 5>는 매개변수 엔티티 예이다.

```
<!ENTITY Hschoi "Choi, Han-Suk."
<!ELEMENT INFO (SUBJECT, AUTHOR, EMAIL?, VERSION?,
DATE?)
<!ENTITY % SA "SUBJECT, AUTHOR"
<!ELEMENT INFO (%SA;, EMAIL?, VERSION?, DATE?)
```

<그림 5> 매개변수 엔티티 예

XML문법에서 사용되는 표시(Notation)선언은 외부 이진 데이터의 구체적인 형을 표시함으로써, 명시한 정보가 해당 XML 인스턴스를 처리하는 응용에 직접 전달된다. 표시선언 사용 예는 <그림 6>과 같다.

```
<!NOTATION GIF87A SYSTEM "gwswin/gws.exe"
<!ENTITY boilerplate SYSTEM "/standard/logo.gif" NDATA GIF87A
```

<그림 6> 표시 언어 예

XML 문법에서 주석문은 아무런 의미도 갖지 않는 구문으로서 <!--> 과 --> 을 사용하여 정의한다. 주석문은 엘리먼트 선언 형식의 내용부분과 CDATA섹션 부분을 제외하고는 XML DTD 또는 문서 어디든지 사용 가능하다. XML에서 처리명령어는 <?> 와 처리명령 이름으로 시작되고 <?> 에 의해서 종료된다. CDATA 및 PCDATA섹션은 XML 파서가 인지하는 문자 집합을 일반문자와 같이 사용할 때 사용된다. XML규격 1.0에 의하여 검증된 XML 문서 예는 <그림 7>과 같다. <그림 7>은 <그림 4>에서 예시한 XML DTD를 기반으로 생성된 XML 인스턴스의 한 예이다.

```
<?xml version="1.0" encoding="EUC-KR" ?>
<?xml:stylesheet type="text/xsl" href="Kware.xsl" ?>
<!DOCTYPE Company SYSTEM "Kware.dtd" ()
<Company>
<Intro ComID="KWare">
<Name>Korea KnowledgeWare INC.</Name>
<Address>
<Post>137-130</Post>
<Addr>서울시 서초구 양재동 66 성경빌딩 4층 402호</Addr>
</Address>
<Phone>02-575-8434</Phone>
```

```
<Fax>02-575-8434</Fax>
<Homepage>http://www.kware.co.kr</Homepage>
</Intro>
<IntroProduct>
<Product Number="1">
<ProductName ToolID="XAT" Ver="1.0">
XML 저작도구
</ProductName>
</Product>
</IntroProduct>
</Company>
```

<그림 7> XML 인스턴스 예

XLL이해

XML 문법 규격 1.0에 이어서 1997년 7월에 발표된 XML 링킹언어 XLL은 XML문서내의 각 엘리먼트 내용에 하이퍼 링크 기능을 제공하기 위해 제안되었다. XLL은 두 자원 또는 두 자원 이상의 하이퍼 링크 기능을 제공하는 Xlink와 자원의 특정 위치를 가리키는 Xpointer로 구성된다.

XLL에서 Xlink는 두 자원에 대한 단순(Simple) 링크와 두 자원 이상에 대한 확장(Extended) 링크로 구분되는데, 링크에 참조될 수 있는 자원은 URL 주소 부여가 가능한 모든 객체로서 문서 자체, 문서 일부, 이미지, 프로그램 등 다양하다. Xlink는 자원 내부에서 링크되는 in-line 링크와 자원 외부 링크 될 수 있는 out-line 링크로 구성된다. 다음 <그림 8>은 단순 링크와 확장 링크 예이다.

```
<mylink XML-LINK="SIMPLE" TITLE="Citation"
HREF="http://WWW.yahoo.co.kr/XML/foo.XML"
SHOW="New"
ACTUATE="USER">It is simple Link</mylink>
<ELEMENT mylink (#PCDATA)>
<!ATTLIST mylink
XML-LINK CDATA #FIXED "SIMPLE"
HREF CDATA #REQUIRED
TITLE CDATA #IMPLIED
SHOW (EMBED|REPLACE|NEW) "REPLACE"
ACTUATE (AUTO|USER) "USER"
```

<그림 8> Xlink 사용 예

XLL에서 Xpointer 사용은 사용자에게 자원을 포함하는 문서의 엘리먼트 트리를 찾아서 자원의 위치를 알려주는 문법을

제공한다. Xpointer의 기본 구조는 관계명(인스턴스, 노드형)으로 구성된다. 여기서 관계명은 절대 위치를 나타내는 ROOT(), ORIGIN(), ID() HTML() 용어와 상대위치를 나타내는 CHILD(), DESCENDENT(), PRECEDING(), PSIBLING(), FOLLOWING(), FSIBLING() 및 스페닝 위치 용어인 ID(), SPAN(CHILD(), CHILD())으로 구성된다.

XSL 이해

XSL 규격은 XML 문서의 외양 및 프리젠테이션 모양을 제어하는 언어 규격으로서 브라우저상에서 XML 문서표현 방법을 제공한다. XSL 스타일시트 언어는 XML문서안의 각 엘리먼트들의 내용 자체나 색, 폰트, 크기, 글자모양, 페이지상의 위치를 지정하여 준다. XML문서에서 XSL를 적용하는 방법은 XML해당 문서 내부에 스타일시트 규격을 포함하거나 따로 작성된 스타일시트 파일을 불러와서 활용된다.

XSL은 SGML의 DSSSL 및 HTML CSS의 중간정도 복잡성을 가지고 있으며, 인터넷 IE5.0에서 CSS 및 XSL 일부를 지원하고 있다. XML문서 및 XSL 문서가 따로 작성된 경우 사용자는 원하는 모양대로 마음껏 표현 형식을 달리 할 수 있다. XSL의 기본 구조는 Construction Rule로서 패턴과 액션 두 부분으로 구성된다. 패턴은 XML 인스턴스에서 스타일 표현을 적용할 엘리먼트를 지정하는 부분이고, 액션 부분은 패턴에서 지정된 엘리먼트를 어떻게 표현할 것인지를 명시하는 부분이다. <그림 9>는 XML 문서에 XSL을 생성하는 예를 보여준다.

```

<!--XML 문서의 StyleSheet-->

<?xml version="1.0" encoding="EUC-KR"?>
<xsl:stylesheet xmlns:xsl="http://www.w3.org/TR/WD-xsl"
xmlns:HTML="http://www.w3.org/Profiles/XHTML-TRansitional"
<xsl:template> <xsl:apply-templates/> </xsl:template>
<xsl:template match="text()" <xsl:value-of/> </xsl:template>
<xsl:template match="/">
<HTML> <HEAD> <TITLE> </TITLE>
<STYLE>
HR color: #888833; margin-top:20px; margin-bottom:20px;
border-style:none; HR.trans .defaultfont font : bold : font-family:

```

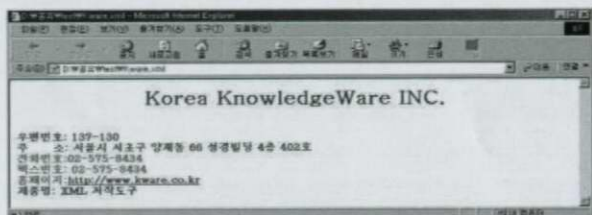
```

바탕:
</STYLE>
<SCRIPT language="JavaScript"><xsl:comment> <![CDATA( )]>
</xsl:comment></SCRIPT> </HEAD>
<BODY bgcolor="#FFFFDD" text="black" link="green" vlink="purple"
alink="red"
marginwidth="0" marginheight="0"> </BODY>
<xsl:apply-templates/> </HTML>
</xsl:template> <xsl:template match="Company"> <xsl:apply-
templates/>
</xsl:template> <xsl:template match="Company/Intro"> <xsl:apply-
templates>
<xsl:template match="Fax|Phone|Homepage">
<font class="defaultfont" style="font-size:12pt; color:red;">
<xsl:if match="Homepage"> <br/> 홈페이지:
<a href="http://www.kware.co.kr"> <xsl:apply-templates/> </a>
</xsl:if>
<xsl:if match="Phone"> <br/>전화번호:<xsl:apply-templates/> </xsl:if>
<xsl:if match="Fax"> <br/>팩스번호: <xsl:apply-templates/> </xsl:if>
</font> </xsl:template> <xsl:template match="Name"> <p align="
center">
<font class="defaultfont" style="font-size:23pt; color:blue;">
<xsl:apply-templates/>
<p/></p> </font> </p> </xsl:template> </xsl:apply-templates>
</xsl:template>
<xsl:template match="/Company/Intro/Address"> <xsl:apply-
templates>
<xsl:template match="Post,Addr"> <font class="defaultfont" style="
font-size:12pt;
color:blue;"> <xsl:if match="Post"> <br/>우편번호: </xsl:if>
<xsl:if match="Addr"> <br/>주 &#160;&#160;&#160;&#160;&#160;&#160;:
</xsl:if> <xsl:apply-templates/> </font> </xsl:template> </xsl:apply-
templates>
</xsl:template> <xsl:template match="
/Company/Intro/Product"> <xsl:apply-templates>
<xsl:template match="ProductName"> <font class="defaultfont"
style="font-size:12pt;
color:blue;"> <br/>제품명: <xsl:apply-templates/> </font>
</xsl:template> </xsl:apply-templates> </xsl:template>
</xsl:stylesheet>

```

<그림 9> XML 및 XSL문서 예

<그림 9>의 XML 인스턴스에 XSL을 적용하여 웹 브라우저에서 디스플레이 하면 <그림 10>과 같다.



(그림 10) intro.xml 문서의 XSL적용 예

실제로 XML문서를 생성하고, 웹 브라우저 상에서 XML문서를 아름답게 디스플레이 하기 위해서는 XSL구문과 스크립트 언어를 접목하여 매우 유용하게 반복적으로 문서의 외양을 표현할 수 있다.

XML의 인터넷 응용

새로운 천년의 웹 문서 표준으로 자리잡고 있는 XML 문서는 인터넷의 급속한 성장으로 다양한 응용에 접목되고 있다. 특히, 인터넷에서 전자상거래가 보편화 됨으로써 기존의 전자문서 교환(EDI)방식이 XML-EDI로 변환되고 있다. XML-EDI는 기존 EDI가 쉽게 해결할 수 없는 새로운 형태의 전자 문서 교환 및 다양한 사용자 요구를 수용한다.

XML-EDI는 다양한 EDI 문서 구조 표현이 가능하고, 웹 환경에서 일괄처리, 상호대화, 실시간 문서교환이 가능하다. 따라서 XML-EDI 서비스를 활성화 하기 위해서 국내외에서 많은 활동 및 파일럿 프로젝트가 진행중이다.

외국에서는 커머스넷 (<http://www.commerce.net/pojects/X12-XML>), W3전자상거래(<http://www.w3.org/Ecommerce/ovrview-xmldefact>), 국제 XML/EDI 그룹(<http://www.xmledi.com>), CEN/ISSS/ XML/EDI 파일럿 프로젝트(<http://www.ccnorm.be/iss/work-sop/ec/xmledi/issxml.html>) 등에서 활발하게 XML/EDI 표준 및 연구 활동을 진행하고 있고, 실제 상거래에 적용한다.

국내에서는 한국무역정보통신(<http://www.ktnet.co.kr>), 데이콤 EC 사업부(<http://ecommerce.dacom.co.kr>) 등에서 XML/EDI 파일럿 프로젝트를 구상하고 있는 중이다. 전자상거래와 관련된 또다른 XML 응용분야인 OFX(Open Financial Exchange)는 금융기관에서 소비자 बैं킹, 소규모 펌뱅크, 주식 및 채권 등의 금융정보를 전자적으로 교환하는데 사용되는 표준 형식이다. XML이 웹 브라우저를 통해 지

원됨으로써 SGML을 사용했던 OFX표준은 금융기관의 온라인 서비스를 가속화 하고, 웹상에서의 금융거래를 활성화 하기 위해 XML 사용으로 전환되고 있다.

또한 인터넷으로 연결된 기업들의 본사와 계열사의 전자 문서 교환 표준인 ICE(Internet Content and Exchange) 프로토콜, 웹상에서의 기업-소비자간 전자 상거래를 지원하는 OTP(Open Trading Protocol), 제품 카탈로그 및 상품의 메타데이터 기술언어인 CBL(Common Business Language), 인터넷을 통한 국제적인 상품구매 표준인 OBI(Open Buying on the Internet) 등에서 XML을 적용하고 있다.

XML의 또다른 중요한 응용분야는 사무실에서 다양한 문서를 생성하고 유통, 저장, 관리하는 문서관리 시스템(EDMS)이다. 사무실에서는 특정 문서 양식 기안, 메모작성, 보고서 작성, 문서 전달, E-mail 등 다양한 문서를 XML표준 문서로 작성 한 후, DB에 저장 관리하고, 웹을 통하여 검색하는 것이 보편화 되고 있다. 즉, 기존의 EDMS는 다양한 문서 형식에 따라 문서 전달에 따른 어려움 및 수많은 문서 뷰어가 필요하였으나 XML 문서관리는 하나의 웹 브라우저를 통해 모든 문서를 브라우징할 수 있는 장점을 제공하기 때문에 많은 사무실, 공공기관, 정부기관에서 XML문서 사용이 보편화 될 것이다. 이미 행정자치부에서도 6가지 공공문서 표준 XML-DTD를 제안하여 배포하였다.

XML은 대학, 연구소, 정부 산하기관, 정보센터에서 구축중인 디지털 도서관에 쉽게 응용되고 있다. 각 기관이 보유하고 있는 학술지 논문, 석·박사 학위논문, 연구보고서 등을 텍스트 기반의 XML표준 문서로 변환하여 DB에 저장·관리하고, 타 기관의 상호대차 및 원문 교환에 활용되고 있다.

현재 W3C에서는 XML응용을 다양하게 확장하고 있음을 보여준다. XML과 관련된 응용은 DOM(문서 객체모델), WIDL(웹자동화 인터페이스), CDF(채널정의프레임워크), MATHML(수학마크업언어), CML(화학마크업언어), SML(동기식 멀티미디어 인터페이스 언어), RDF(자원기술 프레임워크), XML 서명기술, XML 보안기술, 메타데이터, XML과 모빌 액세스 등이다. 앞에서 언급한 바와 같이 XML의 다양한 응용분야에 적용되고 활용됨으로써 새로운 천년의 웹 표준문서로 확실히 자리매김할 것이 예상된다. 