

차세대 정보포맷 표준 XML(eXtensible Markup Language)

XML은 웹이나 인트라넷 환경에서 정보를 생성하고 데이터 자체와 데이터의 포맷 형태를 동시에 공유할 수 있도록 해주는 차세대 정보포맷 표준으로서, 이러한 XML을 통해 다양한 정보를 생성하고 이질적인 시스템에 들어 있는 기업의 다양한 문서나 정보를 보다 쉽게 효과적으로 통합 함으로써 기업의 업무생산성 향상에도 도움을 줄 것으로 기대된다.

송 윤 철 / 뉴미디어사업부 멀티미디어팀장

XML 개요

XML(eXtensible Markup Language)은 1996년 국제 인터넷 기술표준 제정기관인 World Wide Web Consortium(W3C)의 후원으로 형성된 XML Working group에 의해 개발된 것으로(1998년 2월10일 W3C 권고에 준한 XML 1.0 사양이 발표) 지난 호에 소개된 바 있는 SGML(Standard Generalized Markup Language)이 간략화된 버전으로, SGML의 실용적인 기능만을 모은 부분집합(subset)이라 할 수 있다. 즉 XML은 SGML의 장점과 일반성을 최대한 수용하는 한편, SGML의 특정 부분을 발췌·요약하여 전문가와 일반인을 가리지 않고 쉽게 익혀 웹에서 구현할 수 있도록 한 언어이다. 그러므로 XML은 SGML의 부분집합의 관계가 되어 이들간에는 변환이 용이함으로써 XML 문서를 SGML Application에서 그대로 사용할 수 있는 장점이 있게 된다.

XML에 대해 관심이 집중되고 있는 것은 XML이 HTML의 한계를 극복할 수 있다는 점 때문으로 우선

XML을 살펴보기 전에 HTML(Hypertext Markup Language)과 SGML의 한계점을 검토해 보는 것이 XML에 대한 이해를 보다 쉽게 할 것이다.

먼저 HTML은 중첩 구조를 지원하지 못하며 문서의 구조가 단순하기 때문에 전문 검색이 용이하지 못함으로써 객체 지향적인 문서작업을 할 수가 없다. 또한 독특한 형식의 웹 문서나 기존의 스타일과 다른 포맷을 요구하는 클라이언트가 있을 경우 레이아웃 정보와 같은 물리적인 구조와 논리적인 구조를 분리 하지 않고 있는 HTML 마크업(Markup) 특성은 이러한 요구를 만족시킬 수가 없게 된다. 그리고 HTML에서 지원되는 단일 링크형은 실제로 페이지들 사이에서 발생하는 다양한 관계를 지원하지 못하며 정보를 재사용하기도 어렵다. 동일한 데이터를 웹에 발행할 때에도 카탈로그 형태로 인쇄를 하고 이를 데이터베이스 내부에 유지해야 할 뿐만 아니라 변환을 해주거나 경우에 따라서는 직접 재포맷을 해주어야 한다. 그리고 수학공식이나 특정 문자에 대한 지원이 부족하여 이러한 문자를 지원한다고 해

도 플랫폼이 변경되는 경우에 문제가 발생할 수 있다. 웹 페이지는 끊임없이 이동하고 있으며, 웹 이용자들이 변화하는 URL을 찾아 다녀야 하는 문제점 외에 HTML은 여러 버전을 갖고 있어 웹상의 호환성 문제와 사용자의 환경에 맞게 태그를 추가할 수 있는 확장성이 없는 점도 다양한 웹 문서 작성 요구를 충족시킬 수 없는 결점이 되고 있다.

따라서 이러한 HTML의 한계점으로 인해 이용자의 다양한 요구를 만족시킬 수가 없게 되면서, 이를 해결하기 위한 새로운 방안을 모색하게 된 것이다. 그러나 이러한 HTML의 한계점을 극복하고 SGML을 웹상에 적용하기 위한 많은 연구와 어플리케이션 개발이 있었음에도 SGML은 다음과 같은 한계점을 갖고 있다.

GML은 웹상에서 사용되고 있는 하이퍼링크(Hyper Link) 기능 및 웹상에서의 사용을 전혀 고려하지 않고 만들어졌다. 따라서 SGML이 단지 논리 구조에 대한 표준 마크업 언어이므로 포맷과 관련된 어떠한 정보도 지원하지 않는 문제점을 해결하기 위해 DSSSL과 같은 Stylesheet을 개발하여 사용하려고 시도하였지만, 현재까지 이를 지원하는 어플리케이션이 없다는 것이다.

웹상에서 SGML의 사용을 멀리하는 다른 이유는 SGML은 많은 선택적 특성(optional feature)과 복잡한 사용법으로 인해 어플리케이션 개발이 용이하지가 않다.

현재까지 SGML을 완벽히 지원하는 브라우저가 없는 상황이므로 SGML문서를 HTML문서로 변환시켜 HTML브라우저를 통해 웹상에 보여준다. 그러나 사실상 SGML을 HTML로 완벽하게 변환시켜 주는 프로그램은 없다.

따라서 현 시점에서 SGML을 그대로 웹상에 적

용하기는 힘들 것으로 판단 새로운 대안을 찾은 결과 1996년 썬마이크로시스템즈(Sun Microsystems)의 Jon Bosak을 의장으로 한 W3C를 발족하게 되었고, 웹언어로 일반화 한 표준언어 HTML과 SGML의 단점은 없애고, 사용자에 의한 확장성(Extensibility)을 지원함으로써 웹 문서 및 어플리케이션에서 쉽게 구현할 수 있는 장점만을 모아 놓은 새로운 표준인 XML을 제안하게 된 것이다.

W3C의 의장인 Jon Bosak은 이러한 XML에 대해 “향후 XML은 웹 기술상에 있어서 가장 핵심적인 진보를 가져올 것이며, 웹의 근본을 송두리째 바꿀 것이다. XML은 안전한 전자상거래 구축을 가능하게 하고, 새로운 분산 어플리케이션(Application) 시대를 이끌어 나갈 것이다. 또한 XML은 소프트웨어 개발자와 고객의 관계를 새롭게 변화시킬 것이다. 다시 말해서 XML은 어떤 플랫폼에서나 읽을 수 있는 포맷을 제공하기 때문에 특정회사의 제품과 관련된 특정환경에 얹매이지 않아도 된다.”고 설명하고 있다.

HTML과의 비교

XML은 국제 표준 마크업 언어인 SGML을 계승한 언어이다. 따라서 SGML이 가지고 있는 확장성(Extensibility), 복합구조(Complex Structure), 검증(Validation)의 특성을 그대로 가지면서 웹 문서 및 어플리케이션에 쉽게 구현된다는 점이 XML의 특징이자 장점인 반면 HTML과의 명확한 차이점이다.

먼저 HTML이 태그를 추가하거나 새롭게 정의 할 수 없는 반면, XML은 응용 프로그램에 따라 또는 사용자의 편의에 따라 자신의 데이터를 구분하기

위해 새로운 태그 세트와 속성을 정의할 수 있는 확장성을 제공하는데 이는 사내의 인트라넷·엑스트라넷과 같이 특화된 웹 기술 활용용 분야에 적용될 가능성이 크다. 그리고 수시로 새로운 기능을 추가해 업그레이드 할 필요가 없어 브라우저와의 호환성 문제가 해결되며 어떤 플랫폼에서나 읽을 수 있는 포맷을 제공하기 때문에 특정 환경에 얹매이지 않아 다이나믹 HTML이나 자바와 혼합프로그래밍이 가능하여 더욱 유연한 브라우저를 만들 수 있다.

둘째로 문서 구조상의 연속적인 중첩을 허용함으로써 필요시 HTML이 지원하지 않는 객체지향 구조 혹은 데이터베이스 스키마의 구성을 위한 여러 번의 중첩을 허용한다는 것이다. 그리고 문서 구조의 검증이 필요한 어플리케이션을 위해 문법적인 구별을 문서 안에서 제공할 수 있어 DTD와 instance를 파싱(parsing)하였을 때 문서의 오류를 쉽게 판단할 수가 있다는 점 외에도 어떠한 형태의 문서라도 다양한 유형의 문헌 모델링이 가능하며 웹사이트들의 간단한 시스템 관리를 제공하기도 한다.

XML로 만들어진 웹 문서는 동적목차(Dynamic Table of Contents)를 적용할 수 있어 정보를 영역별로 구분할 수 있으며 XML로 구축한 웹페이지는 인터넷 사용자들에게 정보를 정렬하거나 정보검색작업을 손쉽게 해주며, 차세대의 하이퍼텍스트 기능의 제공, 문서의 내용과 스타일 정보를 분리시킴으로써 문서의 재사용이 용이한 점도 빼 놓을 수 없다.

향후 개인화된 웹과 개선된 검색서비스, 채널 정의 포맷을 이용해 PC로 정보를 보내는 푸시기술을 이용한 정보제공 및 복잡한 데이터 구조의 정보들은 모두 XML로 작성될 것으로 보인다. 특히 사내의 인트라넷이나 엑스트라넷과 같이 특화된 웹 기술 활용용 용분야에 적용될 가능성이 크다.

SGML과의 비교

(DTD측면) XML 문서가 XML 규칙에 따라 잘 구성된 문서(Well-formed document)라면 DTD를 반드시 필요로 하지는 않는 반면, SGML은 어떠한 경우든 반드시 DTD를 필요로 한다. 그러나 잘 구성된 XML 문서일지라도 외부 파일을 사용하거나 XML 문서의 구조를 변경하는 등 전문적인 기술을 부가할 경우에는 DTD를 사용하는 것이 효과적이다.

(선언부측면) SGML에서는 markup minimization, link type features, 그 외의 특성을 지원하지만 XML에서는 이러한 특성을 지원하지는 않는다. 아래에 이를 열거해본다.

Markup minimization

o DATATAG

어떤 character를 data character와 markup character로 모두 사용할 수 있도록 허용하는 특성이다. XML에서는 이를 지원하지 않는다.

```
<!DOCTYPE foo [  
<!ELEMENT foo - O ( [ bar, "," ]) +  
>  
<!ELEMENT bar OO (#PCDATA)>  
>]  
<foo><bar>Bar1, Bar2, Bar3
```

〈그림 1〉 DATATAG의 예

o OMITTAG

태그의 생략여부를 표시하는 기능으로 (-, O)으로 표시한다. 그러나 XML에서는 시작 태그와 끝 태그를 모두 마크업 해 주어야 한다.

- o SHORTTAG

다음과 같은 사항은 SGML에서는 허용하지만 XML에서는 지원하지 않는다.

```
Unclosed start-tags : <BODY  
<P>Hello</p></BODY>  
Unclosed end-tags : <BODY>  
<P>Hello</p</BODY>  
Empty start-tags, : <MEMO><>Jane  
Empty end-tags : <BODY><P>In Animal  
...</>
```

〈그림 2〉 SHORTTAG의 예

- o RANK

엘리먼트들의 레벨을 표시하는 기능이다. 예를 들어, 엘리먼트가 중첩 되었을 경우 rank 번호를 자동으로 삽입하여 상·하위 수준을 표시하는 기능이다.

그 외 XML에서 사용되지 않는 특성

- o CONCUR

한 문헌에서 사용할 수 있는 DTD의 수를 결정하는 기능이지만 XML에서는 단일 DTD만을 지원한다.

- o SUBDOC

한 문서내에 서로 다른 구조를 갖는 여러 개의 문서를 지원하는 기능으로 SGML subdoc entity의

수를 표시하는 특성이지만 XML에서는 subdoc 기능이 없다.

- o FORMAL

public identifier를 formal public identifier로 사용할 것인지 여부를 표시하는 특성이다. 이 특성은 hypertext 시스템에서 외부객체와 링크를 시킬 때 사용하는데 XML에서는 이 기능을 지원하지 않는다.

Reference concrete syntax 측면

Reference concrete syntax에서 다음과 같은 사항이 다르다.

- o 단축참조 딜리미터로 선언하지 않는다. XML에서는 SHORTREF와 USEMAP 선언을 사용하지 않는다.

- o Quantities와 Capacities는 제한하지 않는다.

- o 이름은 대소문자를 구별한다.

- o Underscore와 colon은 이름에서만 사용한다.

- o 이름은 Unicode 문자로 사용할 수 있고, ASCII 문자로 제한하지 않는다.

ENTITY 선언 측면

XML에서는 다음과 같은 엔티티 선언을 허용하지 않는다.

- o PI 엔티티는 허용하지 않는다.

- o #DEFAULT 엔티티로 선언하지 않는다.

- o 외부객체는 반드시 시스템 식별기호와 공용 식별기호를 함께 기술해야 한다.

- o External CDATA, SDATA 그리고 internal CDATA, SDATA는 허용하지 않는다.
- o 각괄호 텍스트 엔티티는 허용하지 않는다.

```
<!ENTITY formula MS "RCDATA [ <a> =
<pc/|E|>]" >
```

〈그림 3〉 각괄호 텍스트 엔티티

ELEMENT 선언 측면

XML에서는 다음과 같은 엘리먼트 선언을 제약한다.

- o & 연결기호를 사용하지 않는다. (&는 선언된 엘리먼트는 반드시 발생 하지만 그 발생 순서는 상관없다.)
- o 여러 개의 엘리먼트를 그룹으로 선언할 수 없다.

```
<!ELEMENT (chatpter, paragraph, p) -- -
(#PCDATA | text | table) >
```

〈그림 4〉 그룹 엘리먼트 선언

- o 엘리먼트 선언에서 CDATA와 RCDATA는 사용할 수 없다.
- o 엘리먼트 선언에서 exclusion, inclusion은 사용할 수 없다.
- o Mixed content 선언에서는 항상 #PCDATA를 선언해 주어야 한다.

```
<!ELEMENT (chatpter, paragraph, p) -- -
(#PCDATA | text | table) >
```

〈그림 5〉 mixed content 선언

- o 엘리먼트 선언 값으로 #CURRENT, #CONREF를 지원하지 않는다.

ATTRIBUTE 선언 측면

- o 속성값은 항상 인용부호 (" ")를 사용해야 한다.
- o 속성값으로 외부 개체 참조는 허용하지 않는다.
- o #CURRENT 속성은 허용하지 않는다.
- o NAME과 NAMES, NUMBER와 NUMBERS, NUTOKEN과 NUTOKENS 속성값은 지원하지 않는다. 다만 NUTOKEN만을 지원한다.
- o Name token group은 반드시 'or' 연결자시기 호를 사용해야 한다.

기타

- 마크업 선언 안에서 command는 허용하지 않는다.

```
<!ELEMENT memo (from, to, body) -- 메모
DTD이다. --> -- 틀립
```

〈그림 6〉 마크업 선언내의 command

XML의 장점

이러한 XML의 장점으로는 무엇보다 구조검색 및 전문검색이 가능하며, XML 문서의 구조는 컴퓨터와 사람이 쉽고 명확하게 볼 수 있다는 것이다. 그리고 유효한 XML의 문서는 헤더 정보를 가지고 있어 자신의 문서구조 및 규칙 그리고 처리 명령을 포함하고 있다. 따라서 XML어플리케이션은 XML 문서를 보다 정확하게 처리할 수 있는 것이다.

또한 DTD를 이용하여 문서의 논리구조를 다양한 형식으로 표현할 수 있는 유연성이 있으며, 기존의 단순형 링크보다 양방향 링크라든가 다방향 링크 기능을 지원하고 있다.

마지막으로 XML은 문서의 논리구조와 스타일을 분리하므로 하나의 정보를 이용하여 다양한 목적에 맞도록 정보를 재 가공 할 수 있다. 즉 정보자원에 적절한 스타일 시트를 추가함으로 프린트형태, 웹상의 문서, 점자형태 등 이용자가 원하는 포맷으로 만들 수 있는 것이다.

XML의 구현환경

XML 문서를 만들기 위해서는 다음과 같은 환경과 소프트웨어가 필요하다.

첫째, XML 문서를 작성하기 위한 도구로 문서 편집, 마크업 및 데이터 수정, 그리고 문서 구성요소들을 편집하기 위한 수단을 제공하여 쉽게 문서를 작성할 수 있도록 지원하는 어플리케이션인 “XML 에디터”가 있다.

둘째, XML 문서가 잘 구성된 문서인가 또는 유효한 문서인가를 검사하는 소프트웨어인 파서(Parser)는 문서 파일을 일련의 논리적인 문서 요소별로 구분하고 이들 문서 요소들이 관련된 DTD와 합당한지를 검사하는 프로그램으로 문서가 유효한 것인지 아닌지를 검토하여 오류가 발생하였을 경우 에러를 검출하여 알려주는데, 주로 XML processor에 내장된 형태가 많다.

셋째, XML 문서의 특성을 그대로 반영하여 Web 상에 보여주기 위해서는 XML을 지원하는 브라우저(browser)가 필요한데 구조화된 XML 문서를 컴퓨터 스크린에 보여주는 기능의 소프트웨어이

다. XML browser는 다양한 검색기능을 갖추어 이용자들이 원하는 정보를 정확하고 신속하게 검색할 수 있도록 해야 한다.

넷째, “DTD Editor”는 DTD를 개발할 경우 단조로운 작업의 반복과 제어, 그리고 XML의 모든 규칙에 따라 적합한 DTD를 개발할 수 있도록 해주는 도구로서 매우 복잡하고 많은 시간이 소모되는 작업이다.

다섯째, “XSL Software”로 XML 문서는 문서의 내용과 스타일 정보를 분리하고 있다. W3C는 XML 문서의 스타일 정보를 기술하기 위한 표준으로 XSL(eXtensible Style Language)을 제안하고 있다. 문서의 스타일 정보를 이용하여 Stylesheet을 설계하는 작업은 DTD를 개발하는 작업 못지 않게 복잡하고 어려운 작업이다. 이에 대한 보다 능률적이고 효과적인 지원을 위해 Style Sheet Editor가 필요한데 이는 XML의 기본적인 구성

요소에 대한 정보를 포함하고 있어야 하며 스타일 정보와 문서내의 요소를 매칭시켜 해당 요소를 호출하고 디스플레이하는 기능을 갖추어야 한다.

여섯째, XSL의 기능 가운데 transformation 기능을 전문적으로 지원하는 소프트웨어인 Converter는 비 XML 문서를 XML 문서로 또는 XML 문서를 새로운 포맷으로 자동 생성하기 위한 도구이다.

결론

지금까지 XML의 전반적인 것에 대하여 간략히 살펴보았는데, 전술한 바와 같이 XML은 HTML과 SGML의 장단점을 중간자적인 입장에서 수용한 새로운 언어기술 방식이다. 그런데 HTML은 HTML 나름대로의 독자적 영역을 확보하고 발전해 나갈 것이며, XML은 또다른 영역에서 사용될 것이다. 그

것은 XML에서 HTML을 포함하고 있을지라도 사용자들은 간단한 홈페이지를 사용하기 위해서는 HTML을 사용하는 것이 나을 수 있으며, 간단한 홈페이지를 위해서 HTML 보다는 훨씬 복잡한 XML을 이용하여 자신만의 태그를 설정하고 그 태그를 이용하여 홈페이지를 만들지는 않을 것이기 때문이다.

그러므로 XML, HTML, SGML은 향후 각각의 영역별로 발전해 나갈 것이다. 인터넷상에서 홈페이지 기술은 HTML이 지속적으로 발전·변화해 갈 것이며, 보다 세밀하고 복잡한 응용은 SGML을 이용한 독자적인 전용 브라우저를 통하여 될 것이다.

현재 미국 마이크로소프트와 넷스케이프(Netscape)를 포함한 인터넷 관련업체는 인터넷 브라우저 차기 버전이 XML을 지원할 것이라고 밝힌 바 있으며, 마이크로소프트는 XML을 이용하는 CDF(Channel Definition Format)를 제안하였고, Explore 4.0 버

전은 XML을 지원하도록 개발되어 있다. 그리고 지난 4월 캐나다의 음니마크사가 첫 상용제품을 발표했을 정도로 세계적으로 시장이 초기단계에 있는 상황이다. 국내에서도 XML의 저작도구 개발이 활기를 띠고 있는 가운데 자체기술로 개발한 XML 저작 도구가 조만간 선보일 것이며 XML은 올 한해 진입기를 거쳐 내년부터 빠르게 확산될 전망이다.

결국 XML은 웹이나 인트라넷 환경에서 정보를 생성하고 데이터 자체와 데이터의 포맷형태를 동시에 공유할 수 있도록 해주는 차세대 정보포맷 표준으로서, 이러한 XML을 통해 다양한 정보를 생성하고 이질적인 시스템에 들어 있는 기업의 다양한 문서나 정보를 보다 쉽게 효과적으로 통합함으로써 기업의 업무생산성 향상에도 도움을 줄 것으로 기대되며, 전자출판·디지털도서관·전자상거래·경영·법률·의학·판매자동화 등에서 매우 광범위하게 이용될 것이다. ☀

Y2K문제대응

『Y2K 추진사례』 홍보용비디오 대여안내

우리 협회에서는 국내 Y2K문제 해결을 위한 최고경영층의 인식을 확산시켜 나가기 위하여 외국의 Y2K문제 대응사례를 모은 홍보용 비디오를 한국 유니시스(주)와 공동제작하여 '98년 6월 20일부터 대여하고 있습니다.

○비디오 대여문의 : 한국정보통신진흥협회 홍보자료팀

전화 : 02)580-0583, 0584

○주요내용

- 이 비디오는 미국, 오스트리아등의 운송, 금융, 소매업의 최고경영자와 전산관련 책임자가 말하는 자사 Y2K 문제 대응 현황과 추진사례에 대한 내용을 담고 있습니다.