

차세대 웹에서의 멀티미디어 동기화 기술 (Synchronized Multimedia Technology on Next Generation Web)

신 봉 희* 김 성 종**
(Bong-Hi Shin) (Seong-Jong Kim)

요 약 웹 서비스를 기반으로 한 인터넷은 비약적으로 발전하고 있으며, 차세대 웹 표준화를 위한 노력이 전세계적으로 진행 중에 있다. 처음 개발된 당시 웹은 텍스트 기반의 구조에 맞추어 HTTP, HTML, URL이 제안되어 이를 통해 비동기적인 형태의 검색과 단순하고 단일한 방식의 표현 방식을 사용해 왔다. 그러나 최근 인터넷 상의 상당수의 데이터들은 보다 복잡해지고 구조화되어 가고 있으며, 동기적인 멀티미디어 정보를 포함하는 등 새로운 구조 및 표현 방식을 요구하게 되었다. W3C의 사용자 인터페이스 도메인 중 멀티미디어 동기화 그룹에서 현재 표준화 작업중인 언어는 SMIL로 SMIL은 웹 상에서 멀티미디어 요소들이 잘 통합되어 어느 위치에서 얼마동안 표현되는지를 기술하는 XML-기반 언어이다. 본 논문에서는 SMIL 관련 표준화 동향 및 주요 이슈들을 연구 분석하고 기술 개발 내용에 대하여 논의한다.

Abstract Internet based on web service is growing rapidly and the effort to standardize the next generation web is world-widely being made. When the web was developed for the first time, HTTP, HTML, and URL were designed based on the structure of text background. Through those, asynchronous search and simple, unified expression patterns have been used. But recently many data on internet are becoming complicated. Consequently new structure and expression patterns including synchronous multimedia information are requested. The currently used standard language among user interface domain of W3C is SMIL which is XML-based one. SMIL describes where and how long the multimedia factors are integrated on the web. In this paper the standardization trend and important issues related to SMIL are reviewed and analysed. Also the development of technology is discussed.

1. 서 론

과학기술의 발달로 정보의 디지털화는 정보와 지식으로 대표되는 지적 재산의 대량생산을 가능하게 하였으며, 정보통신의 광역화와 초고속화는 정보의 생산자와 소비자로 하여금 자유롭게 부가가치를 창출할 수 있는 가상공간의 새로운 세계로 인터넷을 확장해 가고 있다. 전세계 인터넷 인구는 2002년에 2억명, 2005년에는 3억 5천만명에 이를 것으로 추정되고 있으며, 비약적인 인터넷 성장의 뒤에는 웹 서비스가 있다. 웹 서비스로 인해 인터넷은 학술적인 목적의 네트워크에서 모든 사람의 생활 환경으로 자

라 잡은 것이다. 이러한 현상의 배경에는 정보를 소유하려는 인간의 원초적 욕구와 웹 서비스가 서로 합치되는 것과 무관하지는 않지만 기술적으로 볼 때 웹의 하부구조가 갖는 장점인 단순성과 개방성을 들 수 있다. 인터넷 어느 곳의 정보든 접근할 수 있는 표준화된 URL(Uniform Resource Locator)과 사용자가 쉽게 배워서 이용할 수 있는 HTML(HyperText Markup Language) 표현 그리고 이들 내용을 전송할 수 있는 HTTP(HyperText Transfer Protocol) 표준 프로토콜의 3가지 요소의 개방형 특성이 사용자 및 개발자의 확대를 유도하였다.

1989년 당시 CERN(Center for European Particle Physics)의 Tim Berners-Lee에 의해 처음 개발되었던 WWW(World Wide Web)의 구조 및 사용자 인터페이스는 매우 간단하였으며, 텍스트 기반을 통한 비동기적인 형태의 서비스 유형으로 문서 검색과 단순한 표현 방식을 사용하였다. 그러나 최근 인터넷 상의 다양한 멀티미디어

*인천대학교 전자계산학과
**극동대학교 전산정보학부

데이터의 등장과 폭발적인 웹 사용자의 증가는 새로운 구조 및 다양한 인터페이스 제공 뿐 아니라 멀티미디어 정보들의 동기화 및 웹이 갖는 사회적인 이슈 등도 고려되는 새로운 웹 기술의 개발을 요구하게 되었다.[1-2] 초창기의 단순 문자 기반 분산 하이퍼텍스트 시스템에서 현재는 이미지, 애니메이션, 동영상 및 음성 등의 연속 미디어를 포함하는 분산 하이퍼미디어 시스템으로 발전하였으며, 소수의 전문가 층을 위한 정보 서비스 환경에서 불특정 다수를 대상으로 하는 환경으로 발전하였다. 또한, HTML 문서를 기반으로 하는 정보의 수동적 응용 환경에서 전자상거래 등을 위한 능동적 응용 환경으로 발전하고 있으며, 최근에 등장한 XML(eXtensible Markup Language)은 이러한 발전 방향에 큰 영향을 미치고 있다.

오늘날 많은 일반 사용자나 응용 서비스 개발자 및 콘텐츠 제공자들은 텍스트뿐만 아니라 오디오 및 동영상 데이터를 전송하거나 받을 수 있는 서비스를 요구한다. 이러한 요구는 인터넷에서 멀티미디어를 전송하는데 필요한 여러 통신 프로토콜의 개발을 촉진하였고, 대표적인 표준으로 IETF(Internet Engineering Task Force)에서 제공한 RTP(Realtime Transport Protocol), RTCP(Realtime Transport Control Protocol)[3-4], 및 RTSP[5](Real Time Streaming Protocol)와 W3C(World Wide Web Consortium)의 SMIL(Synchronized Multimedia Integration Language)[6]을 들 수 있다.

본 논문에서는 차세대 웹 표준화를 담당하고 있는 W3C에 대하여 간략히 소개하고, W3C에서 현재 추진중인 차세대 웹 기술들 중에서 사용자 인터페이스 도메인에 속해있는 워킹 그룹의 하나인 멀티미디어 동기화의 핵심 기술인 SMIL 관련 표준화 동향 및 주요 이슈들에 대해 연구 분석하고 기술 개발 내용에 대해서 기술한다.

2. W3C

2.1 활동 목적 및 배경

웹은 인터넷 상에서 광범위하게 분산된 다양한 형식의 데이터를 손쉽게 검색할 수 있으며, 모든 사용자 환경을 손쉽게 포함시킴으로써 새로운 데이터의 통합을 위한 가장 성공적인 기술로 평가받고 있다. 처음 개발된 당시 웹은 텍스트 기반의 구조에 맞추어 HTTP, HTML, URL이 제안되어 이를 통해 비동기적인 형태의 검색과 단순하고 단일한 방식의 표현 방식을 사용해 왔다.[7] 그러나 최근 인터넷 상의 상당수의 데이터들은 보다 복잡해지고 구조화되어 가고 있으며, 동기적인 멀티미디어 정보를 포함

하는 등 새로운 구조 및 표현 방식을 요구하게 되었다. 예를 들어 실시간 멀티미디어 스트림들을 웹 상에서 CD-ROM과 같은 동기화된 형식으로 재생하거나, 웹 문서의 색인 검색, 데이터베이스 저장, 플랫폼 독립적인 문서 표현 인터페이스의 사용 등을 들 수 있다. 차세대 웹은 기존의 HTTP, HTML, URL 뿐만 아니라 HTTP-NG(HTTP Next Generation), XML, SMIL, RDF(Resource Description Framework), DOM(Document Object Model) 등의 새로운 표준 기술을 이용하여 확장될 것이며, 이를 통하여 웹이란 인터페이스는 모든 인터넷 상의 사용자 응용을 대변하는 수단으로 구축될 것이다.

웹과 관련된 기술 개발은 W3C를 중심으로 표준화가 이루어지고 있으며, W3C는 이를 3개의 도메인(domain), 즉 구조(Architecture), 사용자 인터페이스(User Interface), 기술 및 사회(Technology and Society) 도메인과 1개의 프로젝트 WAI(Web Accessibility Initiative)로 나뉘어 체계적으로 추진 중에 있다.[8] 웹 서비스에 대한 기술의 개발은 웹 인터페이스 환경을 통하여 이루어지며, 웹의 급속한 발전으로 CERN이 전체 망과 관련 표준기술을 독자적으로 개발하는 것은 불가능하게 되어 DARPA(Defence Advanced Research Projects Agency)와 EC(European Commission)로부터 지원을 받아 1994년 10월 미국 MIT(Massachusetts Institute of Technology)에 W3C를 처음 설립하게 되었다. W3C의 연구 목적은 W3C의 모토인 "Leading the Web to its Full Potential"에서 볼 수 있듯이, 가능한 모든 웹의 가능성을 개발하여 웹의 미래와 앞으로의 진화를 위한 기술적인 방안 및 가이드를 제시하는 것으로 웹과 관련한 응용 및 서비스뿐만 아니라 사회적 인 영향까지 포함하고 있다.

2.2 도메인 별 워킹 그룹

<표 1>은 현재 W3C 도메인 별 워킹 그룹을 나타낸 것이다. 각 워킹 그룹들은 연구 결과로 표준 문서를 만들고 이를 실제로 시험하기 위하여 Jigsaw 웹서버, libwww 라이브러리, Amaya 등에 적용하여 구현하는 과정을 거친다. W3C에 의해 개발된 표준 문서들은 회원들에 의해 승인된 이후에 검토과정을 거쳐 워킹 드래프트(Working Draft), 제안 권고안(Proposed Recommendation), 권고안(Recommendation) 과정을 거쳐 표준 문서를 개발하고 있다.

<표 1> W3C 도메인 별 워킹 그룹 분류

도메인	워킹 그룹	공개 코드
구조 도메인	<ul style="list-style-type: none"> ☞ HTTP ☞ HTTP-NG ☞ 웹 특성화 ☞ XML ☞ TV와 웹 	Jigsaw 웹 서버 Libwww 프로토콜 라이브러리
사용자 인터페이스 도메인	<ul style="list-style-type: none"> ☞ HTML ☞ 스타일 시트 : CSS, XSL ☞ 멀티미디어 동기화 ☞ DOM ☞ 수식 : MathML ☞ 그래픽스 : SVG, WebCGM ☞ 국제화 ☞ 이동 접근 ☞ 음성 브라우저 	Amaya 브라우저 / 편집기
기술 및 사회 도메인	<ul style="list-style-type: none"> ☞ XML 서명 ☞ 메타데이터 : RDF, PICS ☞ 사생활 : P3P ☞ 전자상거래 	
WAI	<ul style="list-style-type: none"> ☞ Accessibility Guidelines ☞ Technical Activity International Program Office 	

2.2.1 구조 도메인

구조 도메인에서는 웹 상에 전세계적으로 분산된 다양한 데이터를 이음매 없이 통합하고 검색할 수 있도록 하는 웹 구조 전반에 관한 기술들을 표준화하고 있다. 이 도메인의 주요 관심사는 컴퓨터간 통신 관점에서 발생하는 이슈들을 포함한다.

2.2.2 사용자 인터페이스 도메인

사용자 인터페이스 도메인에서는 사용자와 컴퓨터간 통신 관점에서 발생하는 다양한 기술들을 표준화하는데 주력하고 있으며, 주로 사용자 언어 및 표현 방식에 관한 표준들을 제정하는 일을 하고 있다.

본 논문과 관련된 멀티미디어 동기화 워킹 그룹은 1996년 "Real-time Multimedia Web" 워크샵을 계기로 웹 상에서 실시간 오디오, 비디오와 같은 다양한 멀티미디어 정보들을 CD-ROM과 같이 동기화된 형태로 표현하기 위하여 만들어 졌으며, 처음에는 멀티미디어 HTTP, URL 등 프로토콜을 포함한 다양한 이슈들을 포함하여 구조 도메인의 활동으로 존재하다가 최근 멀티미디어 동기화 언어에 초점을 맞추어 사용자 인터페이스 도메인으로 옮겨 관련 언어의 표준화 연구에 주력하고 있다.

멀티미디어 동기화 그룹에서 현재 표준화 작업중인 언어는 SMIL로 SMIL은 웹 상에서 멀티미디어 요소(이미지, 오디오, 비디오 클립, 애니메이션, 텍스트)들이 잘 통합되어 어느 위치에서 얼마동안 표현되는지를 기술하는 XML-기반 언어로써 SMIL 1.0이 권고안으로 되어 있다. 최초의 상업용 SMIL 재생기는 Real Network사의 G2로써 WAV, AVI, JPEG, MPEG 등의 데이터들을 SMIL에 정의된 문법에 따라 재생시켜 준다.

<표 2> 멀티미디어 동기화 워킹 그룹의 작업 문서

제 목	상 태	일 자
Synchronized Multimedia Integration Language(SMIL) 1.0 Specification	권고안	1998. 6

2.2.3 기술 및 사회 도메인

현재 웹 기술의 발전은 단순히 컴퓨터 관련 분야 뿐 아니라 사회적인 문제까지 영향을 미치게 되었으며, 이에 따라 W3C 기술 및 사회 도메인은 사용자 간 통신 관점에서 발생하는 다양한 이슈들에 대한 표준들을 제정하는 일을 하고 있다.

2.2.4 WAI

W3C의 또 다른 관심 중의 하나는 단순히 기술 개발을 통한 표준화에만 있는 것이 아니라 웹을 사용하는 모든 사람들, 특히 전세계 약 7억 5천만이 넘는 장애인과 같은 별도의 관심이 요구되는 그룹을 고려한 교육 및 개발 지침서를 만드는데 있다. WAI의 기술 활동으로는 크게 기존 W3C에서 개발한 HTML 4.0, CSS 2와 같은 표준들을 장애인들을 위해 수정, 확장하는 작업을 하는 기술개발 부분과 이를 위한 지침서 작성 부분, 그리고 이를 실제 도구로 구현하여 배포하는 일을 담당하는 도구 부분으로 나눌 수 있다.

3. 멀티미디어 동기화 기술 언어

3.1 SMIL의 등장

SMIL은 텍스트 편집기 등으로 간단한 멀티미디어 표현물을 쉽게 만들 수 있도록 하기 위하여 개발되었으며 동기화된 하이퍼미디어를 쉽게 텍스트 편집기로 만들 수 있다. 현재의 기술로 멀티미디어를 웹에서 보여 줄 경우 한번 사용한 미디어를 다른 응용물에서 다시 사용하기가 곤란하며 또한 각각의 미디어에 대해 서로 다른 플레이어를 가지고 있어야 한다는 문제점이 있다. 이러한 한계를 극복하고자 1996년 10월에 W3C에서 "Real Time Multimedia and the Web"이란 주제로 워크숍[9]이 있었고 1997년 3월에는 W3C에 SYMM(SYNchronized MultiMedia) 워킹 그룹이 결성되었다. 그 결과로 1997년 11월 6일 첫 SMIL Working Draft[10]가 발표되었다. SMIL 표준화 그룹인 SYMM은 CD-ROM, Interactive Television, 웹 그리고 오디오 비디오 스트리밍에 관련된 전문가들로 구성되어 있다.

SMIL은 인터넷에서 TV와 같은 품질의 콘텐츠를 전송할 수 있도록 하기 위한 목적으로 승인한 일종의 마크업 언어로서 지난 1998년 6월에 버전 1.0이 제정되었다. SMIL의 특징은 기존의 HTML 등 하이퍼텍스트를 위한 마크업 언어에서 도외시 되었던 멀티미디어 데이터 간의 시간적인 동기 관계를 명시할 수 있도록 함과 아울러 시간 축에서 하이퍼링크도 지정할 수 있도록 한다는 점이다. SMIL은 문자, 그림, 오디오, 비디오와 같은 미디어들의 URL을 개체의 지시자로 사용하고, 이들의 표현을 동시 혹은 순차로 할 것인지에 대한 계획을 기술하는 언어이다. SMIL 문서의 문법은 SMIL DTD에 의해 정의되며 DTD 표기를 사용하여 정의할 수 없는 속성값의 문법은 속성값을 가질 수 있는 속성을 사용하여 첫 번째 엘리먼트와 함께 정의된다. 이와 같은 속성의 문법은 XML 1.0 규격에 정의된 EBNF(Extended Backus-Naur Form)을 사용하여 정의한다.

SMIL에서는 시간적인 동기 관계의 명시함에 있어서 두가지 주요 태그인 <seq>와 <par>을 이용하여 모든 동기 관계를 명시하도록 함으로서 배우기에 쉽고 사용도 편리하도록 하고 있다. 또한 CWI, RealNetwork, NIST 등에서 SMIL 편집기와 플레이어 등을 참조용으로 또는 베타버전으로 제공하고 있다. 본 장에서는 SMIL 규격서(W3C Working Draft 09-November-97)에 나와있는 웹의 성장에 대한 뉴스방송을 표현한 예제를 인용하여 SMIL의 동기관계 명시 기능을 연구 분석하여 소개하도록 한다.

대화형 뉴스방송의 시나리오는 다음과 같다. 먼저 화

면의 왼쪽에 웹의 성장 곡선을 나타내는 그래프를 왼쪽에 보이고 오른쪽에서는 아나운서가 나와서 그래프에 대해 설명하는 비디오를 보인다. 물론 배경화면이 뒤에 차지하고 있다. 다음으로 웹 성장곡선 그래프는 인터뷰를 위하여 인터뷰에 초대된 사람의 화면으로 대체된다. 인터뷰하는 동안 사용자는 인터뷰에 초대된 사람의 화면을 클릭함으로써 그 사람의 홈페이지를 방문할 수 있다.

```
<smil>
  <head>
    <layout type="text/smil-basic">
      <tuner id="left-video" left="20" top="50"
z="2" />
      <tuner id="left-text" left="20" top="120"
z="2" />
      <tuner id="right-video" left="150"
top="50" z="2" />
      <tuner id="right-text" left="150" top="120"
z="2" />
    </layout>
  </head>
  <body>
    <par>
      <img href="bg" />
      <seq>
        <par>
          <img href="graph" loc="left-video"
dur="60s" />
          <text href="graph-text"
loc="left-text" />
        </par>
        <par>
          <a
href="http://www.w3.org/People/Berners-Lee">
            <video href="tim-video"
loc="left-video" />
          </a>
          <text href="tim-text" loc="left-text" />
        </par>
      </seq>
    </par>
  </body>
</smil>
```

```

</seq>
<seq>
  <audio href="joe-audio" />
  <audio href="tim-audio" />
</seq>
<video id="jv"
href="rtsp://www.w3.org/video/joe-video.mpg"
loc="right-video" />
  <text
href="http://www.w3.org/video/joe-text.txt"
loc="right-text" />
</par>
</body>
</smil>

```

<코드 1> SMIL 문서 예

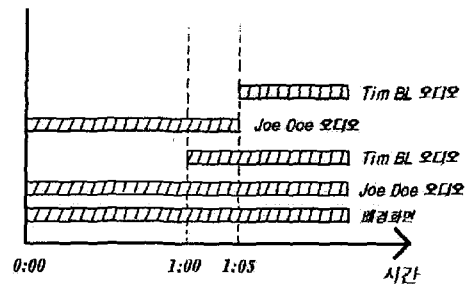
<코드 1>에서 보는 바와 같이 SMIL 문서의 기본 골격은 HTML과 매우 유사하다. 즉 모든 SMIL 문서는 <smil> 태그로 시작하여 </smil>로 끝나고 그 내부는 <head> 부분과 <body> 부분으로 구성된다. 우선 헤드 부분에서는 SMIL 문서의 메타 정보를 제공하는데 여기에는 저자이름, 제목, 저작권 등을 명시할 수 있을 뿐 아니라 이후 바디에서 레이아웃으로 사용될 채널을 정의할 수 있다. <코드 1>은 웹의 성장에 관한 뉴스를 SMIL로 마크업한 것으로 네 개의 채널이 정의되어 있다.

바디 부분은 각종의 미디어 리소스와 이들의 동기관계를 명시한 것으로 <그림 1>과 같이 도식화 할 수 있다. 제일 먼저 병렬 태그인 <par>가 있고 </par>와의 사이에 5개의 요소, 즉 , <seq>, <seq>, <video>, <text>가 있다. 이들 5개 요소는 <par>의 속성에 의하여 동시에 시작된다. 즉, 플레이가 시작되면 백그라운드 이미지가 출력되면서(href="bg"), 화면의 왼쪽에 웹의 성장에 관한 이미지(href="graph")가 이미 헤드에서 정의된 left-video 채널에 60초간 출력됨과 아울러 그바로 밑에 graph-text에 저장된 텍스트가 left-text 채널에 출력된다. 한편 이와 동시에 //www.w3.org/video/joe-video.mpg에 있는 Joe의 비디오가 RTSP 프로토콜을 통하여 스트리밍 되면서 right-video 채널에 출력되면서 바로 밑에 //www.w3.org/video/joe-text.txt에 있는 텍스트가 http 프로토콜을 통하여 전달되어 right-text 채널에 출력된다. 또한 동시에 Joe의 음성이 시작된다(href="joe-audio"). 여

기서 joe-audio의 러닝타임이 65초라 가정한다면 65초 후엔 순차 태그인 <seq>문의 속성에 의하여 tim-audio가 시작된다(href="tim-audio").

<그림 1>은 오디오와 비디오를 중심으로 나타낸 것이고 그림에 포함되지 않은 것은 다음 5 가지이다.

- ① 웹 성장 그래프를 시간 0:00에서 1:00까지 보여준다.
- ② 'Web Growth'라는 문자열을 보여준다. 이 문자열은 후에 'Tim B.-Lee'라는 문자열이 나와 겹쳐져서 없어지게 된다.
- ③ 'Joe Doe'라는 문자열을 시간 0:00에서 Joe Doe 비디오가 끝날 때까지 보여준다.
- ④ 'Tim B.-Lee'라는 문자열을 Tim 비디오가 시작되는 동안 보여준다.
- ⑤ 사용자는 Tim의 비디오가 보이는 동안 비디오와 텍스트를 마우스로 선택함으로써 그의 홈페이지로 이동할 수 있다.



<그림 1> 코드 1의 동기 관계 도식

또한 두 번째의 <seq> </seq> 쌍 내에는 다시 두 개의 <par> </par> 쌍이 있는데 이들은 순차 태그인 <seq>의 속성에 따라 순차적으로 수행된다. 즉, 첫 번째 <par> </par> 쌍 내의 의 출력 기간이 dur="60s"로 60초로 지정되었으므로 두 번째 <par> </par> 쌍은 시작 후 60초가 지난 후에 작동하게 되어 left-video 채널에는 tim-video가 그리고 left-text 채널에는 tim-text가 출력되기 시작한다. 한편 두 번째 <par> </par> 쌍 내의 <a>와 태그는 HTML의 하이퍼링크와 유사한 개념으로 60초 후에 tim-video와 tim-text가 출력되기 시작한 후에는 사용자가 언제든지 left-video 채널이나 left-text 채널에 해당하는 영역을 클릭하면 HTML 브라우저가 팝

업되면서 Tim의 홈페이지인 //www.w3.org/People/Berners-Lee가 나타나게 된다.

SMIL에는 이상에서 설명한 <par>, <seq>, <a> 태그와 아울러 통신환경이나 사용자 컴퓨터의 QoS에 따라 선택이 가능한 <switch> 태그와 시간축 상에서 비디오 상으로부터 하이퍼링크가 가능한 <anchor> 태그가 제공된다. 특히 <a> 태그가 해당 미디어 리소스 전체에 대하여 시작시점부터 끝시점까지 링크가 가능하도록 고정된 것에 반하여 <anchor> 태그는 정의된 기간동안 화면 어느 곳, 특히 플레이되고 있는 비디오 위에도 하이퍼링크 영역을 지정할 수 있다.

3.2 SMIL의 특징

SMIL의 특징적 요소로는 다음과 같은 점들이 있다.

- ① 배우기 쉬운 동기화 : SMIL에 대한 개념은 두 개의 동기화 요소(순차, 병렬)를 이해함으로써 90% 이상이 설명된다.
- ② 시간 축에서의 하이퍼링크 : HTML의 하이퍼링크를 모두 허용하며 동영상과 같이 변화하는 화면에서 각 시점별로 하이퍼링크를 설정할 수 있다.
- ③ 미디어 객체에 대한 재사용 : 모든 멀티미디어 요소들은 URL을 통하여 사용될 수 있다. 즉, 이 요소들은 SMIL 파일에 내포된 것이 아니어서 다른 웹 페이지에서도 URL을 통하여 사용할 수 있다.
- ④ 부하 균형 : 한 화면에 보이는 각각의 미디어 객체는 서로 다른 서버에 저장될 수 있다. 이것은 각 미디어 객체가 URL을 통해 사용할 때 생기는 또다른 이점으로 미디어에 대한 부하를 분산시킬 수 있다.
- ⑤ 언어 선택 : 저작자는 사용 가능한 언어를 미리 선언하여 다중언어를 고려한 웹 페이지를 만들 수 있어 사용자에게 편리한 면을 제공한다.
- ⑥ 대역폭 선택 : 저작자는 오디오와 같은 미디어들을 서로 다른 버전으로 선언한 다음 사용자와의 네트워크 통신 속도에 따라 적절한 버전으로 선택할 수 있다. 따라서 통신 속도가 느린 곳에서도 사용자가 지루함 없이 멀티미디어를 즐길 수 있다. !

3.3 SMIL의 문서 구조

3.3.1 smil(<smil>....</smil>)

SMIL 문서는 <코드 2>의 구문 구조에 따라 속성 선언부(smil-attribute)와 헤드 그리고 바디 부분을 갖는다. 본 논문의 <코드 1>에서는 속성 선언부가 선언되지 않아 SMIL 문서의 기본값을 쓰게된다. 기본값은 현재 브라우저의 구현 속성에 따른다. 여기서 가질 수 있는 속성 선언부는 id와 lipsync가 있는데 id는 SMIL 문서에서 고유변수를 지정하기 위한 것이고 lipsync는 SMIL 문서의 기본 Lipsync 형식을 쓸 것인지 말 것인지를 결정하기 위한 것이다.

```
smil-doc = "<smil" *smil-attribute (">" [head] [body] "</smil">" "/>")
```

```
smil-attribute = id lipsync
```

```
id = "id" "=" (<">id-value<">" ""id-value""")
```

```
id-value = (any legal XML name symbol)
```

```
lipsync = "lipsync" "=" (<">lipsync-value<">" ""lipsync-value""")
```

```
lipsync-value = "true" "false"
```

```
head = "<head" *head-attribute (">" *head-element "</head">" "/>")
```

```
head-attribute = id
```

```
head-element = comment layout-section; more to be defined later
```

```
body = "<body" *body-attribute (">" [body-content] "</body" ">" "/>")
```

```
body-attribute = id
```

```
body-content = *comment-or-ool-link schedule-or-switch *comment-or-ool-link
```

```
comment-or-ool-link = comment out-of-line
```

```
schedule-or-switch = schedule switch
```

```
comment = "<!--" *text "-->" ; the string "--" must not occur within comments
```

<코드 2> SMIL 문서 구문 구조

```
<switch>
  <audio href="joe-audio-better-quality"
```

```

bitrate="16000" />
<audio href="joe-audio" bitrate="8000" />
</switch>

```

<코드 3> switch 요소의 사용 예

```

layout-section          = layout "<switch"
*switch-attribute (">" *layout "</switch" ">"
                    "/>")
layout                  = "<layout" *layout-attribute
(">" layout-element "</layout" ">"
                    "/>")
layout-attribute       = id layout-type
layout-type            = "type" "="
(">layout-language<>"
<'>layout-language<'>)
layout-language       = "text/smil-basic"
external-language
layout-element        = layout-element-basic
external-layout-element
layout-element-basic = *tuner
external-language    = *TEXT

```

<코드 4> Layout-section의 구문 구조

3.3.2 head(<head>....</head>)

Head의 구문 구조는 <코드 2>에서 정의하고 있으며 헤드 태그와 어트리뷰트 그리고 헤드 요소들로 구성된다. 헤드 요소로는 주석과 layout-section이 있다. 또한 layout-section에서는 레이아웃을 정의하기 위한 레이아웃 요소와 사용자의 환경을 고려하기 위한 switch 요소가 있다. 이 switch 요소는 전송속도 또는 사용언어 등을 다변화하기 위한 요소이다. <코드 3>은 switch 요소의 사용 예이다.

레이아웃-섹션의 구문 구조는 <코드 4>에 보였다. <코드 4>에서는 헤드 어트리뷰트가 정의되어 있지 않으며 헤드 요소로 레이아웃을 쓰고 있다. 레이아웃은 어트리뷰트로 id와 layout-type을 갖는다. 레이아웃-타입으로는 'text/smil-basic'이라는 SMIL에서 제공하는 기본

layout-language를 선택할 수 있다. 물론 다른 레이아웃-언어를 선택할 수도 있다. 'text/smil-basic' 레이아웃의 기본 특징은 다음과 같다.

- ① 위치 지정은 좌측 상단부터 시작한다.
- ② 사용되는 변수는 left, top, z, width, height가 있다. z는 화면에 뿌려질 물체의 깊이를 나타낸다. 기본 값으로는 1이며 이것은 바탕화면 깊이에 해당한다. 값이 클수록 화면의 앞쪽에 표시된다.

다음으로 레이아웃 요소가 있는 <코드 5>에서는 tuner를 사용하였으며 튜너의 구문 구조를 보였다. 튜너는 시각 매체(문자열, 이미지, 비디오 등)의 크기와 위치를 정한다. 본 논문의 코드에서 튜너는 4개의 화면영역에 대한 위치와 크기를 정하였다. 이것은 나중에 바디 부분에서 튜너의 아디를 이용하여 그 화면 위치를 선택할 수 있도록 한다. <코드 1>에서 이미지의 위치를 loc="left-video"로 하여 튜너 정의를 선택하였다. 이것은 매체의 위치를 튜너를 통하여 결정한 것이다.

```

tuner                  = "<tuner" *tuner-attribute ("/>"
"></tuner" ">")
tuner-attribute       = id left top z width height
left                  = "left" "=" (<">*DIGIT[%]<">
<'>*DIGIT[%]<'>)
top                   = "top" "=" (<">*DIGIT[%]<">
<'>*DIGIT[%]<'>)
z                     = "z" "=" (<">*DIGIT<">
<'>*DIGIT<'>)
height                = "height" "="
(">*DIGIT[%]<"> <'>*DIGIT[%]<'>)
width                 = "width" "="
(">*DIGIT[%]<"> <'>*DIGIT[%]<'>)

```

<코드 5> Tuner 구문 구조

3.3.3 body(<body>....</body>)

SMIL의 바디 부분은 <코드 2>에 정의하고 있다. 바디는 크게 2가지 요소로 나누어진다. 하나는 병렬 요소이고 다른 하나는 순차 요소이다. 이 두 요소는 바디의 스케줄 요소에 해당되며 SMIL의 핵심으로 볼 수 있다. 즉,

각 미디어를 순차적으로 나타낼 것인지 병렬로 나타낼 것 인지를 결정하여 하나의 시간 축에 멀티미디어를 표현하는 것이다. |

3.3.3.1 병렬 요소

병렬 요소는 매체 제어를 병렬로 처리하기 위한 것이다. 이것은 주로 비디오 매체와 오디오 매체를 동시에 플레이하기 위하여 주로 사용된다. 병렬 요소의 구문 구조는 <코드 6>에 보였다.

```
parallel          = "<par" *par-attribute (">"
*par-content "</par" ">" "/>")
par-content      = comment schedule switch link
par-attribute    = id endsync lipsync dur repeat
*sync-attribute *new-attribute
endsync          = "endsync" "="
(">endsync-value<>" "<'>endsync-value<')
endsync-value    = "first" "last" id-ref
id-ref           = "id(" id-value ")"
dur              = "dur" "=" (">clock-value<>"
"<'>clock-value<')
repeat           = "repeat" "=" (">*DIGIT<>"
"<'>*DIGIT<')
new-attribute    = attribute-name "="
(">attribute-value<>" "<'>attribute-value<')
                  ; attribute-name is an XML
name
                  ; attribute-values must
conform to XML syntax
```

<코드 6> 병렬 요소의 구문 구조

병렬 요소의 끝 시간은 병렬 요소의 모든 요소 중 끝 시간이 최대인 것과 같다. 만약 모든 요소들의 끝 시간이 없을 경우에는 이 병렬 요소들의 끝 시간 또한 알 수 없는 것이다. 이런 경우 외부의 이벤트로써 끝을 내든가 사용자가 stop을 지시하여야 한다.

병렬 요소의 속성은 <코드 6>에서 보듯이 동기화를 위한 것과 반복 그리고 플레이 시간을 위한 것들이 있다.

본 논문에서는 속성을 사용하지 않았으나 간단히 살펴보면 다음과 같다.

① Lipsync : lipsync는 병렬 요소에 있는 매체들과의 동기화를 어떻게 하는가를 정한다. 먼저 이 값이 'TRUE'이면 모든 매체들이 같은 시계를 사용하여 동기화 된다. 반면 이 값이 'FALSE'이면 각 매체가 독립된 시계를 사용하여 플레이 된다.

② Endsyntax : endsync는 병렬 요소에 있는 매체 사이의 끝 시간을 결정하기 위한 것이다. 먼저 이 값이 'LAST'(기본 값)이면 모든 병렬 요소에 있는 매체는 그 중 가장 긴 시간을 갖는 매체의 끝 시간을 전체 요소의 끝 시간으로 한다. 반면 이 값이 'FIRST'이면 병렬 요소에 있는 매체 중 그 시간이 가장 짧은 시간을 전체 요소의 끝 시간으로 한다. 이 외에 각 매체를 자신의 고유 시간에 끝낼 수도 있다.

③ dur : 이것은 병렬 요소에서 매체의 시작 시간과 끝 시간의 차이를 기술하는 것으로 플레이 기간을 말한다.

④ Repeat : 이것은 매체의 플레이 회수를 의미하며 기본 값은 1이며 그 값이 0인 경우 무한대로 반복한다.

⑤ New-attribute : bitrate, language 등을 기술한다.

3.3.3.2 순차 요소

순차 요소는 미디어의 제어를 순차적으로 하기 위한 것이다. 순차 요소의 구문 구조는 <코드 7>에 보였다. 순차 요소의 속성은 병렬 요소에서 이미 설명한 id, dur, repeat, sync-attribute, new-attribute가 있다. 순차 요소에 있는 매체는 기본적으로 다음과 같은 속성을 갖는다.

① 순차 요소에 있는 첫 번째 매체의 시작 시간은 순차 요소의 시작 시간이 된다.

② 순차 요소에 있는 모든 매체는 앞에 있는 매체의 끝 시간이 자신의 시작 시간이 된다.

③ 순차 요소의 끝 시간은 마지막 매체의 끝 시간이다.

```
sequential       = "<seq" *seq-attribute (">"
*seq-content "</seq" ">" "/>")
seq-content      = comment schedule switch link
seq-attribute    = id dur repeat *sync-attribute
*new-attribute
```

<코드 7> 순차 요소의 구문구조

3.3.3.3 동기화-속성

동기화 속성은 앞에서 설명한 스케줄의 병렬 요소와 순차 요소에 들어있는 속성으로 각 매체의 동기화에 대한 속성을 지정하기 위한 것이다. 동기화 속성의 구문 구조는 <코드 8>에 보였다.

```

sync-attribute = begin end
begin          = "begin" "=" event-val
end           = "end" "=" event-val
event-val     = <">event-spec<">
<'>event-spec<'>
event-spec    = qualified-event offset
qualified-event = id "(" id-value ")" "(" event
                ")" ["+" delay]
event         = clock-val "begin" "end"
                "ready"
delay         = clock-val
clock-val     = full-clock-val
partial-clock-val timecount-val
full-clock-val = hours ":" minutes ":"
seconds ["." units]
partial-clock-val = minutes ":" seconds ["."
units]
timecount-val = timecount ["." fraction] ["h"
"min" "s" "ms"]; default is "s"
hours         = 2DIGIT ; range from 00 to
23
minutes       = 2DIGIT ; range from 00 to
59
seconds       = 2DIGIT ; range from 00 tp
59
units         = 1*DIGIT ; range from 0 to
ups-1
timecount     = 1*DIGIT
fraction      = 1*DIGIT
offset        = clock-val
    
```

<코드 8> 동기화 속성의 구문 구조

동기화 속성에는 'begin'과 'end'가 있으며 이것은 스케줄 요소 어떤 것에서도 사용할 수 있다. 이 속성은 각 매체의 시작과 끝을 변경할 수 있게 해준다. 동기화 속성은 사용에 따라 두 가지로 분류된다. 하나는 시간 값(offset value)에 의한 것이고 다른 하나는 지정된 사건(qualified event)에 의한 것이다.

먼저 시간 값에 의한 동기화 속성은 두 가지로 분류되는데 하나는 병렬 요소에서 사용되는 경우이고 다른 하나는 순차 요소에서 사용되는 경우이다. 병렬 요소에서 사용된 경우는 begin에 지정된 값이 병렬 요소의 시작 시간에 대한 time-offset 값이 되며, 순차 요소에서는 앞선 매체의 끝 시간에 대한 time-offset 값이 된다.

다음으로 지정된 사건에 의한 동기화 속성은 어떤 특정 사건이 발생되면 행동을 취하기 위함이다. 사건의 종류는 다음 4가지가 있다.

- ① Begin : 매체가 활성화되는 시점을 나타낸다.
사용법 : begin="id(x)(begin)".
- ② End : 매체가 종료되는 시점을 나타낸다.
사용법 : end="id(x)(end)".
- ③ Clock-val : 매체가 특정 시간에 도달했음을 나타낸다.
사용법 : begin="id(x)(45s)".
- ④ Ready : 매체가 활성화될 준비가 된 시점을 나타낸다. 즉, 파일을 다 받은 다음 실행해야 된다면 파일을 다 받은 시점을 나타낸다. 이 경우 최소 준비 시간은 'dur'을 사용하여 지정할 수 있으며 이것을 interlude duration이라 한다.

3.4 신택스(Syntax) 표기 형식

SMIL 구문 표기 형식은 두 가지로 표기되고 있다. 하나는 HTTP 1.1에서처럼 BNF(Backus-Naur Form)를 기본적으로 사용하는 형식(Augmented BNF Notation)이고 다른 하나는 XML의 문서 표기 정의 DTD(Document Type Definition)를 사용하는 형식이다.

간단히 BNF 표기 형식을 살펴보면 다음과 같다. |

- ① 구문은 '='을 중심으로 LHS(Left Hand Side)와 RHS(Right Hand Side)로 나누어진다.
- ② RHS에서 문자열을 표시할 때에는 이중 따옴표(" , ")로 둘러싼다.
- ③ RHS에서 반복된 문자열 또는 다른 LHS를 위해서는 '*'를 문자열 앞에 붙인다. 만약 반복 회수를 제한하고

자 할 때는 $n*m$ 을 사용한다. 여기서 n 은 최소 반복 회수를 말하고 m 은 최대 반복 회수를 말한다. 반면 반복적인 리스트를 나타내고자 한다면 반복 요소 중간에 쉼표를 표시하여야 한다. 이를 나타내기 위해서 '#' 기호를 사용한다. 예를 들어 #element는 *와 마찬가지로 element가 0개 이상 나타난다는 것이다. 다른 점은 element 사이에 쉼표가 들어간다. 즉, element, element, element 등으로 표시된다. 만약 n 만을 사용하면 n 만큼의 반복 회수가 나타나야 한다. 다시 말해 2DIGIT는 두 자리 숫자를 말한다.

④ RHS에서 손톱괄호((,))는 구문 묶음을 위해 사용되는 기호이다. 즉, 여러 항목 중 하나를 선택하고자 할 때, 하나의 요소로 취급하고자 할 경우 이를 사용하여 한 묶음을 표시한다.

⑤ RHS에서 이중 따옴표에 해당하는 선택을 위한 기호로는 긴 막대(|) 표시를 사용한다. 예를 들어 A=(B|C)인 경우에 A는 B 또는 C임을 나타낸다.

⑥ RHS에서 부등호 괄호(< , >)는 문자열임을 나타낸다. 즉, 이중 따옴표 등의 기호 문자를 쓰거나 할 때 이 부등호 괄호를 사용한다.

⑦ RHS에서 꺾쇠 괄호([,])는 선택사항을 나타낸다.

⑧ 주석을 달기 위해서 세미콜론을 사용한다. 이것은 세미콜론 기호에서부터 그 라인이 끝날 때까지를 주석으로 처리한다.

⑨ 기타 OCTET, CHAR, UPALPHA, LOALPHA, ALPHA, DIGIT, CTL 등이 기본 기호 로 사용된다.

4. 결 론

웹은 지금도 수없이 많은 콘텐츠를 생성하며 자신의 영역을 넓혀가고 있다. 더욱 더 많은 새로운 서비스들이 생겨날 것이고 다양한 환경에서 보다 쉽고, 빠르게 접근할 수 있을 것이다. 단순히 기술적인 변화뿐만 아니라 수많은 삶의 방식을, 정치와 문화 사업 영역의 변화를 이끌고 새로운 제 3의 물결로 우리에게 다가오고 있다.

본 논문에서는 웹을 기반으로 멀티미디어 서비스를 제공하도록 함으로써 차세대 웹을 위한 핵심 기술로 손꼽히고 있는 W3C의 SMIL에 대해 살펴보았다. SMIL은 인터넷의 여러 미디어들을 각각의 객체로 취급할 수 있으며 이들을 합쳐 자신의 멀티미디어 표현을 만들 수 있다는 장점 외에도 다음 두 가지의 특징을 가지고 있어 그 파급효과가 대단할 것으로 예측된다.

첫째, 배우기 쉽다. HTML이 어떤 기초지식 없이도 배울 수 있는 언어이듯 SMIL 또한 간단한 태그를 이용하여 구성된 언어이므로 배우기가 매우 쉽다.

두 번째, 소스를 공유할 수 있다. HTML의 경우 다른 사람들이 만들어 놓은 페이지를 방문하여 그 페이지의 소스를 자신의 것으로 만들 수 있듯이 SMIL 또한 소스를 볼 수 있어 훤내내기가 쉽다.

SMIL은 표준이 이미 제정되었지만 아직도 많은 보강과 연구가 필요하다. 특히 SMIL은 QoS를 반영하는 문제와 SMIL 문서를 WYSIWYG 방식으로 편집할 수 있도록 하기 위하여 멀티미디어 데이터베이스나 멀티미디어 처리 기술과 연계된 많은 연구가 필요하다. 본 논문에서는 이와 관련하여 현재 W3C에서 추진중인 차세대 웹 관련 기술들 중 멀티미디어 동기화 기술을 중심으로 현재까지의 작업 결과 및 앞으로의 계획 등을 살펴보았다. 결론적으로 차세대 웹에서 멀티미디어 동기화에 사용될 SMIL을 위해 국내에서도 관련 기술들을 자체 개발하고, 이를 통해 기술 축적을 함으로써 새로운 서비스 발굴 및 W3C로의 상향식 표준화 작업을 추진해 나가야 할 것이다.

참 고 문 헌

- [1] 한국전자통신연구원, 정보통신 표준개발 연구: 인터넷 콘텐츠 표준(안) 및 W3C 기술 보고서, 1998.
- [2] W3C : World Wide Web Consortium, Leading the Web to its Full Potential, 1998.
- [3] H. Schulzrinne, S. Casner, R. Frederick, and V. Jacobson, "RTP : A transport protocol for realtime applications.", RFC-1889. Feb., 1996.
- [4] H. Schulzrinne, "RTP Profile for Audio and Video Conference with Minimal Control", RFC-1890. May, 1996.
- [5] H. Schulzrinne, A. Rao, R. Lanphier, "Real Time Streaming Protocol(RTSP)", RFC-2326, 1998.
- [6] W3C, "Synchronized Multimedia Integration Language(SMIL) 1.0 specification.", W3C 1998, <http://www.w3.org/TR/1998/REC-smil-19980615>.
- [7] Tim Berners-Lee, Robert Cailliau, Jean-Francois Groff, Benerd Pollerman, World-Wide Web : The Information Universe, 1992.

[8] W3C, URL: <http://www.w3.org>.

[9] <http://www.w3.org/pub/WWW/AudioVideo/RTM96.html>.

[10] <http://www.w3.org/TR/WD-smil>.



신 봉 희

1977년 2월 인하대학교 전자공학과
졸업(공학사)
1981년 2월 인하대학교 대학원 전자
공학과 졸업(공학석사)
1995년 2월 단국대학교 대학원 전자
공학과 졸업(공학박사)

1991년 ~ 1993년 미국 아이오와주립대학 교환교수
1978년 3월 ~ 현재 인천전문대학 전자계산과 교수.
관심분야 : 마이크로프로그래밍 언어 설계, 다중처리 컴퓨
터, 퍼지제어, 신경회로망 등



김 성 중

1987년 2월 전자공학 학사
1989년 2월 전자공학 석사(컴퓨터
전공)
1988년 8월 전자공학 박사(컴퓨터
전공)
1988년 3월 ~ 현재 극동대학교 전산정
보학부 전임강사

관심분야 : 영상처리, 전자 상거래, 병렬처리 등