

자바 TV를 활용한 데이터 방송 저작 시스템

(주)포디얼 정동훈

1. 서 론

디지털 방송은 디지털 혁명의 중심에서 산업전반에 큰 영향을 주고 있다. 우리나라에서는 작년에 디지털 방송에 대한 정부의 정책이 구체화되고 9월초부터 지상파 디지털 방송이 시험방송 형태로 시작되면서 본격적으로 디지털 방송 시대를 맞이하고 있다.

디지털 방송의 특징은 크게 세 가지로 요약해 볼 수 있다. 첫째는 이미 일반인들도 잘 인식하고 있듯이 고화질, 고음질 방송이 가능하다는 것이다. HD(High Definition)급 방송의 경우 기존 아날로그 방송에 비해 두 배 이상의 고해상도 화질과 CD 수준 이상의 음질을 제공할 수 있다. 둘째는 방송 채널 활용도를 높일 수 있다는 것이다. 즉, 하나의 물리적 채널(physical channel) 대역폭을 이용해 여러 개의 디지털 방송 동시에 송출하여 가상 채널(virtual channel)을 구성할 수 있는 것이다. SD(Standard Definition)급의 디지털 방송의 경우 2개 이상 6개까지도 동시에 송출하고 수신하는 것이 가능하다. 이는 방송의 디지털화 과정에 데이터 압축 효과가 있기 때문이다. 마지막으로 가장 잠재력이 크다고 여겨지고 있는 특징은 데이터 방송이 가능하다는 것이다. 데이터 방송이란 기존의 화면과 음향 정보 외에 디지털 데이터를 함께 전송하는 것으로서, 시청자에게 동영상 외에 데이터 서비스를 제공하는 것이다. 그리하여, 디지털 방송 시청자들은 일방적인 시청형태에서 벗어나 TV를 통해서 기호에 맞는 정보를 대화형으로 열람할 수 있는 것이다.

이제 디지털 방송은 단순히 아날로그 방송 방식의 디지털화뿐만 아니라 데이터 방송을 가능하게 함으로써 방송과 통신의 융합(Digital Convergence)이라는 새로운 패러다임을 만들어 내고 있다고 할 수 있다.

유럽의 경우에는 아날로그 방송에서부터 이미 OpenTV, Canal+ Technologies 등과 같은 업체에서 비표준 시스템(Proprietary System)을 기반으로 오래 전부터 데이터 방송을 서비스해 왔고 전 세계적으로 확장되고 있는 추세이다. 그러나 로열티, 시스템 개선, 가격문제, 호환성 등의 다양한 문제 야기의 개연성이 대두되어 국제적인 표준에 대한 필요성이 커지게 되었다. 현재 국제적인 데이터 방송 표준은 크게 북미 표준, 유럽 표준, 일본 표준으로 구분되고 북미, 유럽은 자바(Java)와 xHTML이 콘텐츠 표현 방식으로 결정이 되었고 일본은 XML(bML)로 결정이 되었다. 데이터 방송 콘텐츠 저작 시스템(authoring system)은 데이터 방송에 필요한 콘텐츠를 쉽고 빠르게 저작, 생성하는데 필요한 시스템이다. 일본의 bML 경우 일본 국제 방송전시회인 InterBEE99에 저작도구(authoring tool) 전문업체, 도시바와 같은 대형 장비업체 등에 의해 개발된 저작도구가 전시되었다. xHTML의 경우 현재 표준 스펙은 다음과 같다.

* DASE-DA(DASE Declarative Application 이전의 Presentation Engine)

1. XHTML 1.0
2. CSS1+CSS2 subset + ATSC properties
3. ATSC DTD provided
4. DOM1 and 2

- 5. ECMAScript Module(ECMA262)
- 6. Media types
text/html, text/plain, image/png, image/jpg, audio/basic, audio/atsc, video/atsc)

7. SMIL

* DVB-HTML

- 1. XHTML modules(XHTML 1.1 subset)
- 2. DOM Level 1 Core and DOM Level 2
- 3. MHP CSS
- 4. ECMAScript Module(ECMA262)
- 5. Media types(text/xml, application/xml, text/css, text/plain, text/dvb.utf8, text/dvb.subtitle, audio/mpeg, image/jpeg, image/png, image/gif, image/mpeg, image/dvb.subtitle, video/mpeg)

6. SMIL

* ATVEF

- 1. HTML4.0
- 2. CSS Level 1
- 3. DOM Level 0
- 4. Media type(text/html, text/css, image/png, image/jpg)

5. Trigger

데이터 방송 콘텐츠의 구현 기술로 자바가 채택되고 있는 이유는 스펙과 기능의 확장성이 좋다는 점과 하드웨어 플랫폼에 독립적이어서 이식성과 호환성이 좋기 때문이다. 참고로, ATVEF(Advanced TV Enhanced Forum)는 Microsoft를 중심으로 형성된 조직이다. ATVEF의 스펙을 따르는 저작도구는 지난 4월 미국 방송전시회(NAB)에 Intel사와 AVID사의 공동개발에 의해 전시되었다.

필자는 자바 저작도구를 중심으로 이야기를 전개하고자 한다. 자바 저작도구는 디지털 TV용 대화형 콘텐츠(interactive contents)를 제작하기 위한 도구이다. 대화형 콘텐츠는 그 자체가 일종의 컴퓨터 소프트웨어로서 시청자의 조작에 따라 그래픽, 동영상, 음향 등의 멀티미디어 정보를 TV화면과 함께 표시할 수 있으며, 더 나아가 TV 기능을 제어할 수 있어야 한다.

자바 저작도구는 주요 국제 표준에서 채용하고 있는 자바 프로그램으로 구성된 대화형 콘텐츠를 생성해 주는 도구로서, 저작에 의해 하나의 컨텐

츠(데이터 방송 프로그램)를 구성하게 되면, 저작 도구에서는 이 콘텐츠가 실제로 셋탑 박스에서 실행할 수 있도록 자바 source code를 생성해 낸다. 생성된 소스는 셋탑 박스에서 그대로 실행될 수 있으며, tool 에서 가상 셋탑 박스 환경도 제공하는 것이 바람직하다.

자바 저작도구는 프로그래밍 언어인 자바에 대한 지식 없이도 자바 프로그래밍을 통해서만 얻을 수 있었던 동적이고 다양한 표현과 확장성을 디자이너가 쉽게 얻을 수 있도록 제작되어야 한다.

2. 최근 기술동향

최근 막바지에 와 있는 디지털 TV 데이터 방송의 표준화 작업에 의해 현재 대화형 콘텐츠 시장은 거대한 단일 시장의 형성을 앞두고 있다. 이러한 표준화 작업은 인터넷에서 축적된 멀티미디어 콘텐츠 기술을 근간으로 이루어지고 있으며, 현재 북미와 유럽 모두 자바와 HTML을 표준의 근간으로 채택하고 있다. 아울러 콘텐츠에 있어 자바쪽 표준은 양대 표준이 모두 동일한 표준 규약을 포함하고 있으므로, 사실상 전세계를 통합하는 하나의 표준이 성립되게 되는 셈이다.

HTML은 이미 인터넷을 통해 잘 알려진 표준으로, 서로 연결되어 있고 멀티미디어가 포함된 문서를 표현하기 위한 수단으로 설계되었다. 기본적으로 TV에서의 HTML의 도입은 다음과 같은 목적에서 추진되고 있다. 첫째, HTML로 작성되어 축적된 인터넷 상의 정보를 PC와 공유한다는 점이 있다. 둘째, HTML을 통해 축적된 콘텐츠 제작 경험과 개발 도구, 인력 활용한다는 것이다. 하지만 PC와 TV는 엄연히 다른 상황에 있기 때문에 본래 의도와는 달리 HTML은 TV에 적용하는데 있어 많은 한계를 가지고 있다. 그 한계점을 정리하면 다음과 같다.

첫째, 구체적으로 보면 우선 HTML을 보완할 수단이 없다. 최근의 웹사이트에서 요구하는 동적인 멀티미디어를 HTML만으로는 수용할 수 없기 때문에 오늘날의 웹 사이트들은 Flash, Shockwave 등의 플러그인(plug-in)이나 자바 애플릿 등을 사용해서 이 한계를 극복하고 있다. PC와 달리 TV는 제조사에 따라 다양한 CPU와 운영체제를 사용하기 때문에 플러그인의 사용이

사실상 불가능하다.

둘째, TV를 위한 콘텐츠와 PC를 위한 콘텐츠는 본질적으로 틀리다. TV는 일반적으로 여러 명이 PC와 비교해서 먼 거리에서, 수동적으로 시청하며, 조작용을 위해 주로 마우스 대신 리모콘을 사용한다. TV를 위해 작성된 콘텐츠는 PC를 위해 작성된 콘텐츠와 글자의 크기, 화면 배치, 조작 방식 등 모든 면에서 달라야 한다. 실제 HTML로 작성된 TV용 콘텐츠는 일반적인 HTML의 사용과 매우 다른 방식으로 작성되는 것이 보통이다. 즉, 단순히 HTML로 작성된 것만으로는 PC와 TV가 동일한 콘텐츠를 공유하는 것은 사실상 불가능하다.

셋째, 콘텐츠에 대한 세밀한 제어가 불가능하다. PC에서도 사용자가 사용하는 웹 브라우저의 종류에 따라 동일한 웹페이지가 서로 다르게 보인다. TV의 경우도 제작사에 따라 동일한 HTML 문서의 모양이나 조작방법에 차이가 생기게 된다. 이는 HTML자체가 화면의 정확한 생김새를 기술하기 위해 만들어진 언어가 아니기 때문이다.

넷째 확장성에 제약이 있다. HTML은 모든 일을 가능하게 하기 위해 만들어진 언어가 아니다. 즉, 서로 연결된, 멀티미디어 정보를 포함하고 있는 문서를 쉽게 표현하기 위해 설계된 언어이다. 디지털 TV의 북미, 유럽 표준의 근간에 확장성을 고려한 수단으로 자바를 포함하고 있는 이유는 여기에 있다.

다음은 자바에 관한 특성과 문제점이다. 자바는 본래 TV와 같은 가전기기에 내장되기 위해 만들어진 기술로, 1995년 등장 이후 주로 웹과 연관되어 인터넷 언어로서의 입지를 확고히 했다. 엄밀히 이야기해서 자바는 프로그래밍 언어라기 보다는 서로 다른 하드웨어와 운영체제를 사용하는 다양한 형태의 컴퓨터를 소프트웨어 입장에서 동일하게 보이게 하는 일종의 소프트웨어 실행 환경이다. 즉, 자바로 작성된 프로그램은 일반적인 운영체제 상에서 실행되는 소프트웨어와 동일한 기능을 하지만, 자바 기술이 들어있는 어떠한 형태의 하드웨어에서도 동일하게 실행된다.

즉, 자바 자체는 본래 HTML과 같은 콘텐츠 형식이라기보다는 콘텐츠를 보여주기 위해 필요한 하부구조라고 할 수 있다. 따라서, HTML이

서로 연결된 멀티미디어 문서를 만들기 위해 만들어진 것과 같이, 애니메이션을 표현하기 위한 HTML 아닌 또 다른 콘텐츠 형식을 만드는 것 역시 가능하다. 이는 자바의 확장성을 단적으로 보여준다.

대화형 콘텐츠 표준에 있어서 자바는 TV자체를 세밀하게 제어할 수 있는 여지를 만들기 위해 도입되었으며, 유럽쪽에서는 아예 HTML, MHEG과 같이 현재 독립적으로 사용되고 있는 다른 콘텐츠를 자바 위에 구축해서 보여주도록 하는 것을 표준의 방향으로 정한 바 있다. 자바를 사용할 경우 다음과 같은 일이 가능하다.

첫째, OSD(On Screen Display), TV채널, 화면, 음량 등 TV 하드웨어의 제어가 가능하다. 새로운 형태의 동영상 형식 등 새로운 미디어를 지원하거나, 다채널을 이용해 multi-angle 스포츠 중계 등의 특수한 서비스가 가능해진다.

둘째, 방송을 통해 전달되는 프로그램 정보, 데이터들의 획득이 가능하다. 방송국이 자신의 채널에 맞는 EPG(Electronic Program Guide)를 만들거나, 실시간으로 전달되는 정보에 반응하는 콘텐츠의 작성이 가능하다. 또한 이러한 정보는 방송의 특성으로 인해 일반적인 인터넷 연결에 비해 고속으로 다수의 대상에게 전달될 수 있다.

셋째, 백 채널(back channel)을 통한 다양한 형태의 네트워킹이 가능하다. HTML의 경우와 같이 HTTP 등 특정한 형태의 프로토콜에 국한되지 않고, 필요에 따라 POP, IMAP과 같은 메일 프로토콜이나 RTP와 같은 미디어 스트리밍(media streaming) 프로토콜은 물론 실시간 게임을 위한 프로토콜 등 어떠한 형태의 프로토콜이든 사용하는 것이 가능하다.

결국 자바의 특성은 확장성과 유연성으로 규정 지을 수 있다. 다만 이러한 특성으로 인해 대화형 콘텐츠를 제작하기 위한 수단으로서 다음과 같은 문제점을 가진다.

첫째, 콘텐츠를 만들기 위해서는 프로그래밍 기술이 필요하다. 앞서 언급한 바와 같이 자바는 멀티미디어 콘텐츠를 만들기 위한 전용 기술이 아니다. 따라서, 비교적 간단한 화면 구성을 위해서도 상당히 복잡한 프로그래밍이 필요하다. 또한, 일반적으로 프로그래머만으로는 콘텐츠 제작이 어려우므로 디자이너와의 협업이 필요한데 이

는 비용과 효율면에서 좋지 않다. 프로그래머와 디자이너가 독립적으로 일할 수 있고, 가급적 프로그래밍이 필요한 부분을 최소화할 필요가 있다.

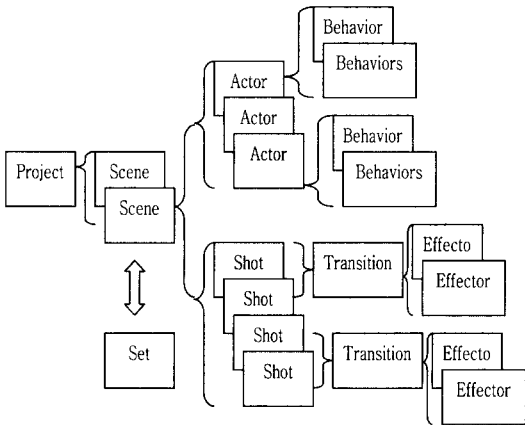
둘째, 개발 도구들이 주로 프로그래밍 지식을 염두에 두고 있다. 자바 프로 그래밍을 돕는 도구들이 이미 많이 나와있지만, 기본적으로 사용자가 프로그래머라는 전체를 두고 있다. 따라서, 디자이너가 쉽게 사용하기에는 문제가 있다.

3. 자바 저작도구

현재 개발되어진 국내 디지털 데이터 방송 시스템 개발 전문업체의 저작도구를 통해 저작 시스템에 대한 이해를 돕고자 한다.

3.1 Overall Structure

이 저작시스템의 저작모델은 전체적으로 영화의 구조를 모델로 하고 있으며, 대부분의 용어들도 영화 용어에서 따온 것이다.



위는 저작모델을 그림으로 표현한 것이다. Project는 일반적으로 영화에 해당하고, 영화에서의 Scene이 보통 한 공간에서 일어나는 장면을 나타내는 것과 같이 이 저작도구의 Scene은 서로간에 공유하는 것이 없는 한 장면을 나타낸다. 영화에서 Scene에 배우(Actor)가 나오는 것과 같이 이 저작도구의 Scene에는 그 Scene에 등장하는 Actor들이 명시된다. 또, 한 Scene은 보통 하나하나의 카메라의 움직임에 해당하는 Shot으로 구성되는데, 이 저작도구에서도 마찬가지로 Actor들의 위치나 상태를 기록한 Shot들이

Scene을 구성하게 된다.

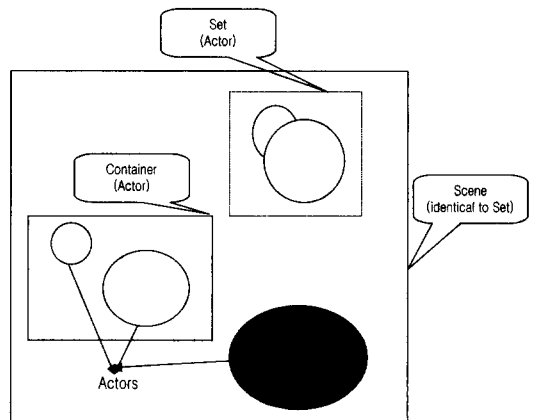
단, 정해진 극본(screen play)에 따라서 흘러가는 영화와는 달리 이 저작도구에서는 각 Shot이 지정된 상태로 설정되면 Scene내의 Actor들은 TV 시청자의 입력에 반응하여 동적으로 상태를 바꾼다는 차이점이 있다. 물론 Shot들간에 정해진 순서가 없다는 점도 영화와는 틀린 점이다.

다음으로 영화에서는 Shot들 간의 장면 전환시 효과를 넣을 수 있을 것이다. 이와 마찬가지로 Scene내에서는 임의의 두 Shot들 간의 전환을 넣을 수 있는데 이를 Transition이라고 한다. Transition 중의 효과는 Effector라는 객체가 담당하는데 Effector는 서로 다른 두 Shot에서 Actor들의 상태의 차이를 기반으로 어떻게 한 Actor가 한 Shot에서 다른 Shot으로 옮겨가는지를 담당하는 객체이다. 예를 들어 Effector는 두 Shot에서 한 Actor의 크기와 색깔이 다를 경우, 그 위치 사이를 직선으로 이동한다든지 색깔이 한 색깔에서 다른 색깔로 천천히 변한다든지 하는 효과를 얻을 수 있다.

3.2 Screen Layout

이 저작 툴에서는 화면에 나타나는 그래픽을 생성하거나 혹은 보이지 않더라도 소리를 낸다든지 네트워크를 통해 서버와 통신을 하는 등 어떤 역할을 하는 객체를 모두 통칭하여 Actor라고 부른다. 한 Scene의 화면은 이들 Actor 중 화면에 나타나는 것들로 이루어진다.

Actor 중에서는 다른 Actor를 포함할 수 있는



특수한 Actor들이 있으며 또한, 하나의 Scene이 별도의 무대장치(set)로 취급되어 Set라는 개념으로 화면에 포함될 수 있다. Set은 Scene과 완전히 동일하며 다른 Scene이나 Set내에서는 Actor와 동일하게 취급된다.

그림에서 보는 것과 같이 Container로 작용하는 Actor들이 있으며 Container내에 또 다른 Container가 포함되는 것 역시 가능하다.

3.3 Resources

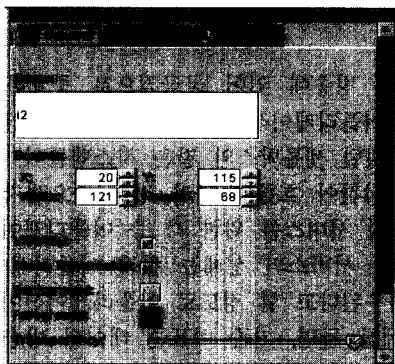
화면에 등장하는 Actor를 비롯해 그림, 오디오 파일 등 콘텐츠를 구성하는 재료들은 Resource로 취급된다. 이를 끌어다가 Scene에 추가하면 실제 Actor로 등록된다.

Resource를 두는 이유는 우선 Resource 자체를 수정했을 경우, 그 Resource에서 생성된 모든 Actor에 그 변화가 자동적으로 파급된다는 점을 들 수 있다. 콘텐츠 전체의 버튼의 모양을 바꾸고 싶다면 하는 경우에는 상당히 편리한 기능이 될 수 있다.

다음으로 하나의 Resource를 통해서 생성된 Actor들 모두가 Resource로 관리되는 데이터를 공유한다는 이점이 있다. 예를 들어 같은 이미지를 여러 개 끌어서 화면에 놓아도 실제 이미지 데이터 자체는 콘텐츠 전체를 통틀어 단 하나만이 존재한다. 즉, 같은 이미지를 반복하여 로드하지 않는다.

3.4 Editing Actors : Property Sheet

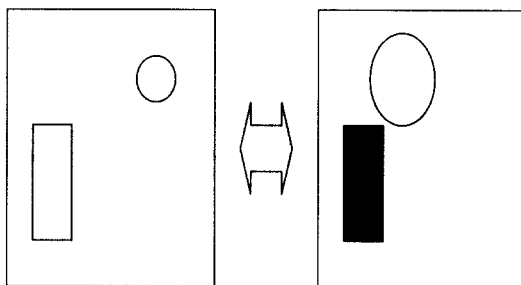
화면에 등장한 Actor들은 직관적인 UI를 통해 쉽게 편집할 수 있도록 하고 있다. 우선 각



Actor들의 상태를 편집하게 위해서 제공되는 것은 property sheet이다.

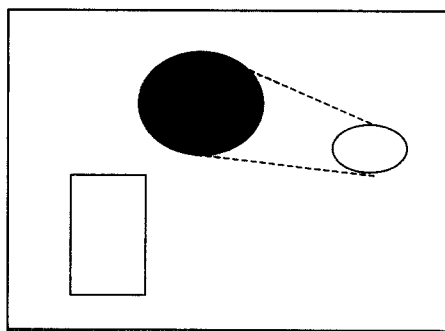
3.5 Shot

Shot은 Shot이 포함된 Scene(Set도 마찬가지)내의 Actor들이 어떤 상태에 있는지를 나타낸다. Scene이 어떤 Shot으로 이동했다는 것은 그 Shot에 기록된 상태로 바뀐다는 것을 의미한다. 아래 그림에서 두 Shot이 있는데, 사각형 형태의 Actor는 색깔의 변화가 있고, 타원 형태의 Actor는 위치와 크기에만 변경이 있다. 이와 같이 Shot은 각 Actor의 상태를 기록함으로써 화면상의 특정한 layout을 필요할 때 복원할 수 있다.



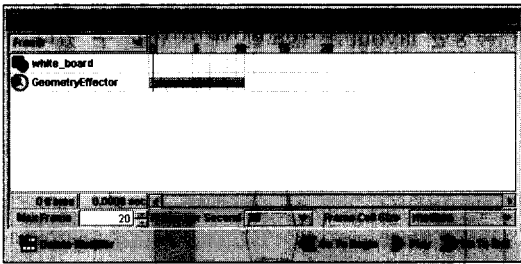
3.6 Transition

Transition은 하나의 Shot에서 다른 Shot으로 이동할 때 어떤 식으로 변화가 일어나는가를 기술한 것이다. Transition은 timeline상에서 Scene을 구성하는 다양한 Actor들의 상태들(위치, 크기, 색깔 등)이 어떠한 상대적 타이밍에 따라 어떻게 변화하는가를 기술한다. 아래 그림에서 타원 모양의 Actor의 크기 및 위치가 서서히 변화하도록 하는 등의 효과를 줄 수 있다.



Actor의 상태 중 어떤 것을 어떤 식으로 변화할 것인가는 전적으로 Effector에 의해 결정된다. 위의 그림은 두 Shot 사이에 타원 모양 Actor의 크기와 위치에만 차이가 있는 예이다. 여기서 사용된 Effector는 서로 다른 두 Shot들에서 타원 모양 Actor의 크기와 위치가 점차적으로 커지는 애니메이션 효과를 주고 있다.

여러 Actor들과 각 Actor들의 다양한 상태(위치, 크기, 색깔 등)들의 변화에 있어서 상대적인 타이밍은 앞서 말한 것과 같이 다음과 같은 timeline으로 지정할 수 있다.



3.7 Scripting

Actor들은 원하는 데이터가 도착했다든지 사용자의 키 입력이 왔을 경우 등에 Event를 발생시킨다. 이 저작 툴에서 콘텐츠가 사용자의 입력이나 외부의 데이터에 따라 어떻게 반응하는지를 결정하는 것은 Actor에 연결되는 Behavior이다. Behavior는 Script언어로 작성된 Script를 가지고 있는데, 이 저작도구에서는 ECMAScript (JavaScript)를 축소하고 변형시킨 형태의 Script언어를 사용한다.

Script에는 Event를 처리할 함수(function)들을 지정할 수 있는데, Actor에서 Event가 발생되면 시스템은 Actor에 연결된 Behavior 중에서 적절한 이름의 함수를 찾아서 실행시키게 된다.

4. 저작도구 개발시 고려해야할 점

첫째, 동적이고 화려한 화면을 간단히 구성할 수 있는 저작 방식이 제공되어야 한다. Flash와 같은 멀티미디어 저작도구와 같이 직관적인 GUI를 통해 쉽게 화면을 구성, 전환하며 특수한 효과를 넣을 수 있게 제작되어 있다. 디자이너는 자바 프로그래밍 지식이 없이도 쉽게 콘텐츠를

제작할 수 있고, 필요에 따라 쉬운 스크립트 언어를 사용하여 콘텐츠를 작성할 수 있어야 한다.

둘째, 디자이너와 프로그래머의 작업을 효과적으로 분리될 수 있어야 한다. 저작도구를 사용하면, 디자이너는 프로그래머의 도움 없이 대화형 콘텐츠를 제작할 수 있어야 한다.

셋째, XML로 콘텐츠를 저장하여 기존 인터넷 기술과의 연동이 용이하도록 한다. XML로 콘텐츠를 저장하면 기존의 인터넷 기술을 이용해서 자동으로 콘텐츠를 변형하거나 데이터베이스와 연동해 필요한 데이터를 삽입하는 것이 가능해진다. XML은 다시 자바 소스코드로 변환되어 TV로 보내질 수 있다.

넷째, 자바 TV API, HAVi UI API 등 북미 및 유럽 표준에서 공통적으로 지원하고 있는 표준을 충실히 지원해야 한다. 표준을 충실히 지원하여 셋톱박스에서의 상호 운용성을 확보한다.

다섯째, A/V 화면위에 조화를 고려한 콘텐츠 편집 가능해야 한다. 아직 멀티미디어 콘텐츠를 위해 사용된 전력이 거의 없는 자바의 특성상 본 발명품과 동일한 확장성과 유연성을 가지는 도구는 아직 개발되지 않았으며, 단지, Flash와 같이 기존 멀티미디어 저작도구로 작성된 콘텐츠를 자바를 사용해서 보여주는 기술은 일부 나와 있다. 하지만, 이러한 기술들은 물론 TV를 겨냥하고 있지 않으며, HTML과 마찬가지로 확장성의 결여로 인한 한계를 가지고 있다.

현재 대화형 콘텐츠를 통한 새로운 사업 모델이 나타나는 시점에서 자바의 확장성을 이용한 새로운 콘텐츠 모델이 활발히 시도되고 있는 상황이며, 이를 위해서는 비프로그래머가 쉽게 사용할 수 있는 저작 도구는 필수적이다.

5. 결 론

데이터 방송에 있어 궁극적으로 중요한 것은 다양한 어플리케이션이다. 초기에는 헤드엔드 장비나 디지털 셋톱박스과 같은 시스템들이 중요하겠지만 시간이 흐름에 따라 점차 중요해지는 것은 양방향 서비스를 위한 각 종 어플리케이션 콘텐츠이다. 서비스의 성패는 킬러 어플리케이션에 의해 좌우된다고 할 정도로 어플리케이션이 중요하다. 예를 들면, 기존 비표준 데이터 방송 솔루션 업체는 200여 개의 어플리케이션 개발 파트너

들이 있어서, 날씨, 스포츠, 뉴스 등 다양한 서비스를 할 수 있는 어플리케이션들이 개발되어 제공될 수 있었다. 그런데, 그 바탕에는 다양한 어플리케이션 개발을 직접적으로 도울 수 있는 저작도구가 있었기 때문이라고 한다. 결국, 데이터 방송의 활성화를 위해서는 기능성이 우수한 저작도구가 필수적이다. 데이터 저작도구는 디지털 방송뿐 아니라 DVD와 같은 유사 디지털 미디어 용으로도 개발하여 DVD에 양방향 서비스를 부여하는 것이 가능하다. 이와 같이 데이터 방송용 저작도구는 다른 목적을 위한 툴로서도 개조가 가능하며 그런 면에서 가치가 높다.

디지털 방송은 우리나라가 세계적으로 도약할 수 있는 좋은 계기가 될 것이라는 것에 대해서는 의심의 여지가 없다. 디지털 방송에서 영상 콘텐츠 못지 않게 잠재력을 가지고 있고 산업적 시너

지를 발생시킬 수 있는 영역이 바로 데이터 방송 관련 시스템과 서비스이다. 데이터 방송 관련 사업은 정부차원에서 적극 장려하여야 하며 관련 업체들도 자부심을 가지고 세계시장을 개척해 나가야 할 것이다.

정 동 훈



1993 포항공대 전자계산학과 학사
1996 SKC&C 전임연구원
1999 (주)클릭스 대표이사
1999~현재 (주)포디얼 상무이사
E-mail: dharna@4dl.com
http://www.4dl.com

● 제27회 정기총회 및 추계학술발표회 ●

- 일 자 : 2000년 10월 27 ~ 28일
- 장 소 : 숙명여자대학교
- 문 의 처 : 한국정보과학회 사무국
Tel. 02-588-9246/7, 4001/2
http://www.kiss.or.kr