

VRML환경에서 Java3D기반 이벤트 처리 설계 및 구현

김성예*, 김해동, 구분기, 최병태, 오원근
한국전자통신연구원

Java3D-based simple event processing for VRML

S.Y. Kim, H.D. Kim, B.K. Koo, B.T. Choi, W.G. Oh

Electronics and Telecommunications Research Institute

E-mail : inside@etri.re.kr, hdkim@etri.re.kr, bkkoo@etri.re.kr, btchoi@etri.re.kr, owg@etri.re.kr

요 약

오늘날 월드 와이드 웹(www)의 성장과 함께 인터넷 상에서 3차원 세계를 표현해야 할 필요성이 급격히 증가하고 있다. 이러한 요구에 부응하여 VRML97이 개발되었고, 이는 인터넷에서 3차원 그래픽스 데이터의 표준으로 중요한 역할을 해왔다. 그리고 현재 VRML97로 기술된 데이터를 위한 브라우저가 많이 개발되어 왔다. 현재까지는 대부분 OpenGL이나 Direct3D와 같은 API를 주로 사용해왔으나 SUN의 Java3D가 활성화 되면서 Java3D 기반의 VRML97 브라우저도 많이 개발되고 있다. 또한 SUN에서는 이미 3D 브라우저를 위한 프로토타입을 개발하여 제공하고 있다. 본 논문에서는 본 연구팀에서 수행하고 있는 서버-클라이언트 모델의 가상현실 시스템의 클라이언트로서의 X3D 브라우저 개발을 위한 기반 연구로서 수행된 Java3D 기반의 VRML97 브라우저의 이벤트 모델에 대하여 기술한다.

1. 서론

오늘날 월드 와이드 웹의 성장과 발전은 인터넷 상에서 실감있고 다양한 형태의 멀티미디어 데이터를 요구하게 되었다. 그러한 데이터들 중의 하나가 3차원 가상공간을 표현하기 위한 VRML이다. VRML은 인터넷상에서 HTML의 성공에 자극받아 가상현실 데이터도 인터넷 표준화가 필요함을 실감하고 SGI사의 주도로 1995년에 버전 1.0이 발표되었다. 그러나 이는 주로 3차원 객체를 표현하는데 그쳐 다양한 애니메이션 효과가 불가능 하였다. 이를 개선한 버전 2.0이 1997년에 발표되었는데 이를 VRML97이라 부른다. VRML97은 가상세계를 구성하는데 필요한 3차원

물체, 색상, 텍스처 그리고 사운드 등을 제공하며 또한, Walkthrough나 Flythrough와 같은 다양한 탐색방법을 통하여 사용자가 가상 세계를 체험할 수 있게 하고 있다. VRML97은 텍스트 형태로 기술된 인터프리터 언어로서 브라우저에 의해서 해석되어 사용자의 디스플레이 장치에 보여지게 된다. VRML97의 주요 특징으로는 자바스크립트(javascript), 음향(sound), 애니메이션(animation), 인라인(inline)을 통한 URL 지정, 그리고 외부 응용 프로그램과의 연동을 위한 EAI (External Application Interface)등을 지원하고 있다는 것이다. 하지만 인터넷에 올리기에 너무 무거워 속도가 느리다는 단점이 있고, 또한 다양한 인터랙션이 불가능하여 많은 응용 분야에서 사용하는 데는 성공

을 거두지 못했다. 따라서 이를 개선해야 한다는 요구가 끊임없이 제기되었고, 이에 부응하여 현재 W3C 산하 Web3D 컨소시엄 주관으로 X3D라는 이름의 버전 3.0이 표준화 진행 중에 있고 2002년에 완료될 예정이다. 한편 MPEG4 표준화쪽에서는 장면을 구성하기 위하여 BIFS(Binary Format for Scene)을 채택하고 있는데 이는 VRML 문법을 가지고 다양한 멀티미디어 데이터를 합성할 수 있도록 확장한 것이다.

X3D 데이터 기술은 XML을 채택하고 있고 MPEG 그룹과도 연동하여 스트리밍 문제나 데이터 압축 문제를 해결하려고 노력하고 있다. 현재까지 확정된 X3D의 주요 기능을 살펴 보면, 지형 데이터 표현, H-anim, 멀티미디어 콘텐츠 기술 등을 들 수 있고, 가볍고 다양한 응용분야에서 사용될 수 있도록 코어(core), 메인(main), 확장(extension) 등으로 모듈화하였고, 현재 확장(extension)을 위해 20여개의 워킹그룹(Working Group)이 활동 중에 있다.

본 논문에서는 X3D 표준화에 발 맞추어 그 브라우저 개발이 요구됨에 따라 X3D 브라우저의 개발을 목표로 하였다. 하지만 대부분의 X3D 명세가 VRML97과 동일하기 때문에 VRML97 브라우저를 먼저 개발하고 여기에 X3D에서 기술된 기능을 확장하여 X3D 브라우저를 개발하고자 하였다. 그 중에서도 VRML97 브라우저에서 사용되는 시스템 구조와 이벤트 처리에 대해서 설명한다.

2. 관련연구

2.1 VRML 브라우저

VRML은 3차원 가상공간을 기술하기 위해서 필요한 객체(노드)들의 자료구조와 기능들을 기술하고 있다. 그리고 VRML 브라우저라 불리기 위해 최소한으로 지녀야 할 것들도 규정하고 있다. 그렇지만 브라우저 구현에 사용되는 언어나 3차원 그래픽스 렌더링을 위해서 사용되는 라이브러리에 대해서는 규정하고 있지 않다. 따라서 사용되는 언어나 3차원 그래픽스 라이브러리에 따라서 많은 브라우저가 개발되어 왔다. 지금까지 개발된 VRML 브라우저들을 살

펴보면 그림 1에서 보는 바와 같이 주로 OpenGL이나 Direct3D 같은 3D 그래픽스 라이브러리를 사용하였고 SUN은 Java와 Java3D기반의 브라우저를 개발하였다[2]. 이들 브라우저의 장단점은 주로 프로토타입, 스크립트 그리고 EAI의 지원 정도가 결정한다. 그림 2는 VRML 브라우저의 일반적인 시스템 구조를 보여준다[3].

Browser	Platforms	Rendering Libraries
CosmoPlayer	Mac, SGI, Win32	OpenGL
blaxxun Contact	Win32	OpenGL
blaxxun CC3D	Win32	Direct3D
OpenWorlds	SGI, Win32, Solaris	Performer, Optimizer, OpenGL, IrisGL
WorldView	Win32, Mac	Direct3D, QuickDraw3D
DpIV	Win32	Direct3D
Cortona	Win32	OpenGL, Direct3D
CASUS Presenter	SGI, Solaris, Win32	Inventor (on OpenGL)
FreeWrl	Linux, Solaris, Digital	OpenGL

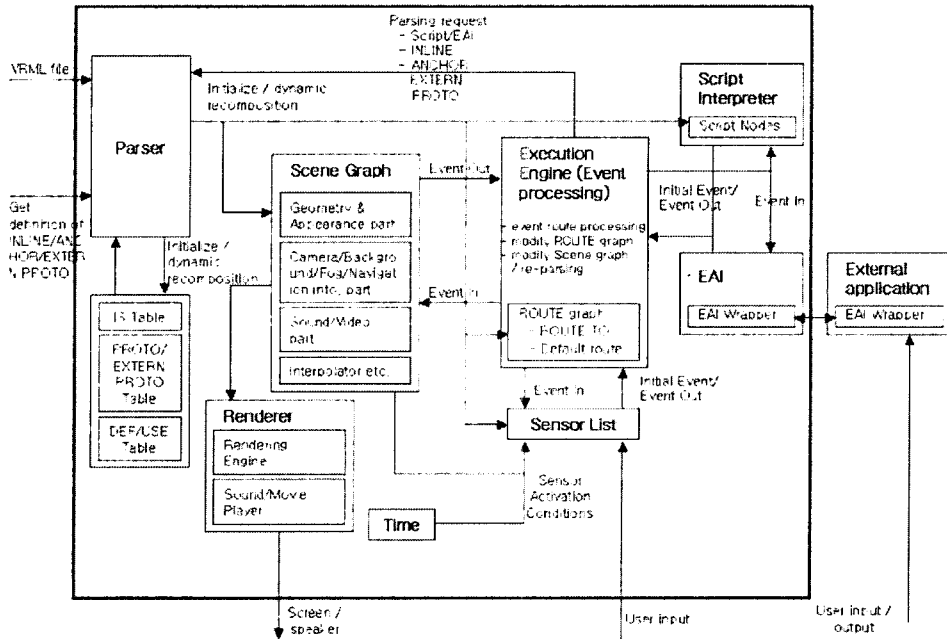
[그림 1] VRML 브라우저의 렌더링 라이브러리

2.2 Java3D

Java3D는 3차원 그래픽스 응용 프로그램을 개발하기 위해 SUN에 의해서 개발된 3차원 그래픽스 렌더링 API 이다. Java3D의 하부는 OpenGL과 Direct3D로 되어 있기 때문에 그래픽스 하드웨어에서 지원하는 것에 따라서 OpenGL로 돌릴 수도 있고 아니면 Direct3D로 돌릴 수도 있는 고수준(high level) API이다 [4]. Java3D의 장면 그래프는 VRML에서 규정하는 장면 그래프와 유사하여 Java3D를 이용하여 VRML 브

라우저를 구현하는데 용이하다. 그리고 장면을 렌더링하고 표현하는데 간단, 용이하고 또한 행위 (behavior) 노드를 제공하고 있어 애니메이션을 위한

인터플레이션의 구현이라든가 다양한 이벤트를 처리하고 전달하는데 편리한 메커니즘을 제공하고 있다.



[그리 2] VRML 브라우저의 구조

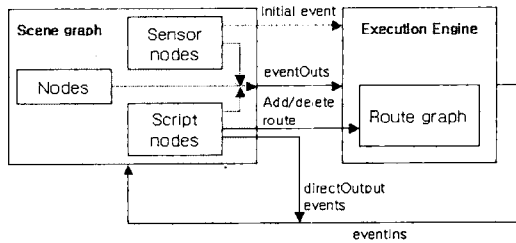
3. 이벤트 프로세싱

그래픽 윈도우를 제공하는 대부분의 응용 프로그램들은 이벤트를 기반으로 동작한다. 즉, 프로그램은 이벤트 루프를 가지고 있고 프로그램에서 쓰이는 이벤트를 그 이벤트 루프에 등록하고 실제로 이벤트가 발생하면 콜백(callback) 루틴을 수행하여 원하는 동작을 한 다음 다시 이벤트 루프로 복귀하는 형태로 수행된다. 마찬가지로 VRML 브라우저도 이벤트 루프를 돌게 되어 있는데 장면 그래프의 구성과 이의 업데이트시 재 렌더링이 이벤트 루프에 의해서 이루어지게 된다. Java3D에서는 이러한 VRML의 이벤트 처리를 쉽게 구현할 수 있도록 알파(alpha), 행위 (behavior), 픽틀(picking utility), 장치 입력 이벤트 (device input event)등과 같은 기능을 제공하고 있다.

VRML 환경에서 이벤트는 장면 그래프내의 한 노드로부터 다른 노드로 전달되는 정보를 의미한다.

예를 들면, 장면 그래프의 외부에서 발생한 사용자의 입력은 어떤 노드의 애니메이션을 시작하도록 하기 위한 소스가 될 것이다. 일단 이벤트가 발생하면, 순차적으로 다음 노드에게 그 이벤트 정보가 전달되어 처리된다. 이러한 일련의 과정을 이벤트 라우팅(event routing)이라고 하며 노드의 상태를 변화시키거나 또 다른 이벤트를 발생시키거나 장면 그래프의 구조를 변화시킬 수 있다. 이와 같은 가상공간의 변화는 애니메이션과 인터랙션에 의해서 발생한다. VRML에서는 이를 위하여 다양한 인터polator(Interpolator) 노드와 센서(Sensor) 노드를 제공한다. 뿐만 아니라 VRML의 스크립트 노드는 개발자가 임의의 이벤트 프로세싱을 정의할 수 있도록 한다. 이상적인 이벤트 모델은 모든 이벤트를 발생된 순서에 따라 순차적으로 거의 동시적으로 처리되는 것이다. 실제로, VRML97 명세는 이벤트 라우팅을 위한 Frozen Time 을 기술하고 있다[5]. 그림 3은 VRML 브라우저의 일

반적인 실행모델(execution model)을 나타낸다[6]. 그림 3과 같이 장면 그래프는 가상공간을 표현하기 위한 기하노드와 인터플레이터 노드, 센서 노드 그리고 스크립트 노드로 구성된다. 이 노드들 중에 단지 센서 노드와 스크립트 노드만이 초기 이벤트를 발생시킬 수 있다. 이 초기 이벤트는 실행 엔진내의 라우트 그래프를 참조하여 각 이벤트의 목적지 노드로 전달된다. 개발자의 입장에서는 이러한 실행 모델을 구성할 때 쉽게 Java3D API가 제공하는 메소드들을 이용할 수 있다.



[그림 3] VRML 브라우저의 개념적 실행모델

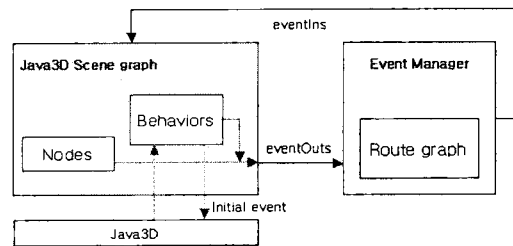
그림 4는 VRML97과 Java3D API에서 유사한 기능을 하는 애니메이션과 인터랙션 노드들을 나타낸 것이다. 이 경우에 Java3D 메소드의 많은 부분이 VRML과 일대일 대응이 가능하다. 실제로 VRML의 모든 인터플레이터 노드들은 Java3D가 제공하는 인터플레이터, 패스인터플레이터(PathInterpolator) 객체로써 표현될 수 있다. 그림 4에서 Java3D의 센서 객체는 VRML의 센서 노드와는 달리 조이스틱이나 HMD와 같은 하드웨어 입력 장치의 추상화 개념을 가진다.

VRML97	Java3D API
CylinderSensor	Alpha
PlaneSensor	Behavior
ProximitySensor	Pick Utility
Sensor	SphereSensor
	TimeSensor
	TouchSensor
	VisibilitySensor

	ColorInterpolator	ColorInterpolator
	OrientationInterpolator	RotationInterpolator
	PositionInterpolator	PositionInterpolator
Interpolator	ScalarInterpolator	ScaleInterpolator
	CoordinateInterpolator	TransparencyInterpolator
	NormalInterpolator	SwitchValueInterpolator
		PathInterpolator
Etc.	Script	KeyNavigatorBehavior
		Sensor

[그림 4] 애니메이션 & 인터랙션 노드들

본 논문에서 구현된 Java3D 기반의 VRML 브라우저에서는 이벤트를 2가지 타입으로 분류하여 처리하였다. 첫째, 사용자에 의해서 외부로부터 발생하는 초기 이벤트이다. 이 초기 이벤트는 Java3D 레벨에서 처리 되도록 할 수 있으므로 별도의 전달 및 관리를 고려하지 않아도 된다. 둘째, 그 밖의 이벤트들에 대해서는 라우트 그래프를 참조하여 이벤트 매니저에 의해서 처리되도록 한다. 즉, 이벤트 매니저는 장면 그래프와 라우트 그래프를 참조하여 현재 처리되는 이벤트를 다른 어떤 노드에 전달할지를 결정한다. 그림 5는 이러한 구조를 가지는 이벤트 모델의 개념을 나타낸다.

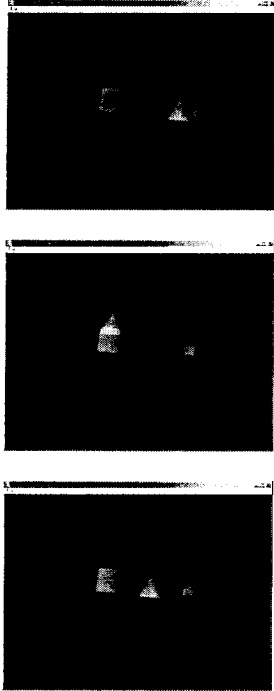


[그림 5] 구현된 VRML 브라우저의 개념적 실행모델

4. 구현 및 결론

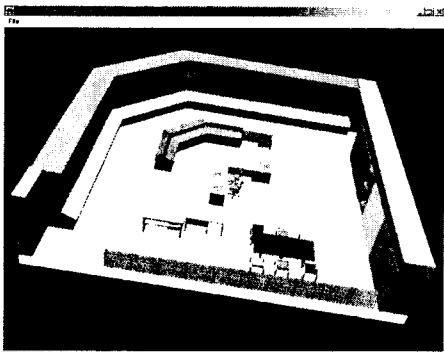
본 결과는 Windows2000 환경에서 Java1.3, Java3D API 1.2.1를 이용하여 구현되었다. 그림 6은 간단한 모델에 대한 애니메이션과 인터랙션을 보여준다. 이 예제는 외부 사용자에 의해서 발생된 초기 이벤트를 가지고 각 물체의 위치(Position), 색상(Color), 투명도

(Scalar), 크기(Scalar), 방향(Orientation)을 변화시키기 위한 인터플레이터가 사용되었다.



[그림 6] 간단한 애니메이션과 인터렉션의 예

그림 7은 조금 복잡한 모델에 대한 실행화면을 보여준다. 이 모델도 색상과 투명도를 변화시키기 위한 인터플레이터를 가지고 있다.



[그림 7] 복잡한 모델의 예

현재 본 결과는 본 연구팀이 수행하고 있는 네트워크 가상현실 및 시스템 개발 과제의 클라이언트 부분의 X3D 브라우저를 개발하기 위한 기반 기술로

서 사용되었다. 또한 이와 같은 VRML 및 X3D 브라우저 기술은 다양한 분야에서 응용될 수 있다.

[참고문헌]

- [1] <http://www.chbergarage.org/top/index.html>
- [2] Greg Seidman's VRML97 Browser Datasheet, <http://web3d.about.com/compute/3dgraphics/library/weekly/aa070698.html>
- [3] T. J. Park, Block Diagram of X3D browser, ETRI, TM200000815, May, 2000.
- [4] Sun Microsystems, The java3D API Specification, Version1.2, April, 2000.
- [5] Rikk Carey, Gavin Bell, The Annotated VRML97 Reference Manual
- [6] Bernie roehl, Justin Couch, Cindy Reed-ballreich, Tim rohaly, and Geoff brown, Late Night VRML2.0 with Java, ZD Press, 1997.