

WPF/E를 활용한 효율적인 콘텐츠 제작 방법 연구

Study on Effective contents Implementation Method that Make use of WPF/E

오경숙, 류남훈, 한재정, 김응곤
순천대학교 컴퓨터학과

KyeongSug Oh, NamHoon Ryu, JaeJeong Han,
EungKon Kim
Dept. of Computer Science,
Sunchon National University

요약

기존에는 다양한 경로를 통해 얻은 데이터의 분석이 숫자, 문자의 표현이나 2차원 형태로도 충분했었다. 그러나 오늘날의 사용자들은 컴퓨터 그래픽스 기술의 발달로 인해 다양한 형태의 사용자 인터페이스를 접하게 되었고, 이로 인해 요구사항은 갈수록 증가하고 있는 추세이다. 초기의 프로그램은 데이터 분석을 위한 기능적인 측면에 중점을 두었으나 오늘날의 사용자들은 시각적, 감성적 느낌을 충족시킬 수 있으며, 즉각적인 분석이 가능한 화면을 요구하고 있다. 본 논문에서는 2차원, 3차원 형태의 그래프, 소리, 애니메이션 등 다양한 형태의 멀티미디어 데이터 구현이 가능한 WPF/E에 대해 알아보고, 이를 활용한 효율적인 콘텐츠 제작 방법을 제시하고자 한다.

Abstract

In the past days, analysis of data that derived from various routes were enough as presentation of numbers and letters or 2D forms. But because of the development of computer graphics techniques, users of nowadays can come in contact with user interface of diverse forms. Because they are able to touch such things readily, their requirements are increased as time goes on. The programs in earlier days laid stress on skillful side for data analysis, but users of nowadays demand screens that satisfy their visual and emotional feelings and can be analyzed immediately. This paper will look over WPF/E that are able to implement various multimedia data such as graph of 2D and 3D forms, sounds, animations and so on, and present effective contents making method with this.

I. 서론

최근의 인터넷 브라우저는 웹이 만들어 놓은 네트워크라는 세상 안에서 그 기능을 수행해야만 한다. 이러한 조건을 충족시키기 위해서는 기존 브라우저 위에서 플래시처럼 한 번의 플러그인을 설치하면 모든 응용 프로그램을 구동시킬 수 있는 기술이 필요했던 것이다. 그래서 탄생한 것이 바로 실버라이트(Silverlight)이다. 실버라이트는 웹 2.0 시대를 맞아 마이크로소프트에서 야심차게 개발한 차세대 RIA(Rich Internet Application) 기술이다. 가볍지만 매우 강력한 웹 브라우저 플러그인으로 RIA를 만들기 위한 벡터, 애니메이션, 고품질 비디오 등 다양한 기능이 포함되어 있다. 또한 운영체제 독립적으로 동작하기 때문에 실버라이트로 만든 애플리케이션은 윈도우, 맥OS, 리눅스 등에서 모두 실행된다. 실버라이트는 WPF의 기능들을 제공하고 성능이 뛰어나며 HTML이나 Ajax 등 웹 개발 기술과 쉽게 연동할 수 있다. 실버라이트는 사용자 경험(UX : User eXperience)을 기반으로 한 강력한

미디어 기능의 웹 어플리케이션 개발과 익스프레션 블렌드 툴을 이용한 애니메이션을 제작할 수 있다. 플래시와 비슷한 개념이고 기능도 비슷하지만 내부를 들여다보면 그 개발 환경은 사뭇 다르다. 플래시는 태생이 애니메이션을 위한 도구로 출발한 것인 반면 실버라이트는 디자이너와 개발자를 위한 새로운 플랫폼으로 출발한 기술이다.

웹 2.0세대에서는 화려한 어플리케이션 개발 환경에서 애니메이션을 표현하기에 플래시를 많이 썼으나, 유지 보수가 힘들고, 액션 스크립트라는 언어적 한계와 오픈 소스 플랫폼이 아니라는 단점을 가지고 있다. 그러나 실버라이트는 방화벽 환경에서도 편리하게 사용할 수 있으며, 막강한 미디어 기능을 지원하며, 닷넷 개발자가 접하기 쉽다. 그리고 유지보수도 쉽고 오픈소스를 지향하기 때문에 RIA 환경에서 실버라이트로 콘텐츠 제작을 하고자 하는 프로그래머들이 늘어나고 있다.

본 논문에서는 II장에서 본 연구와 관련된 내용에 대해서 알아보고, III장에서는 실버라이트의 적용 사례를 알아보고, IV장에서는 실버라이트의 장점을 알아보고, V장에서는 RIA 제품

을 비교하며, 마지막 VI장에서는 결론을 맺는다.

II. 관련 연구

1. 닷넷(.NET)

마이크로소프트가 설계·구현한 대표적인 컴포넌트 아키텍처로 COM(Component Object Model)이 있다[1]. COM은 내부 구현을 모르더라도 운영 환경에 관계없이 언어 중립적 방법으로 사용될 수 있는 컴포넌트의 실현이 가능하였으나 다른 컴포넌트 기술들과 마찬가지로 중요한 문제점이 뒤따른다. COM 컴포넌트를 개발하기 위해 언어 중립적 인터페이스인 IDL (Interface Definition Language)로 인터페이스를 정의한 후 실제 구현에 있어서는 C++와 같은 개발 언어를 사용해야만 한다. 결국 개발자는 작성하고자 하는 프로그램의 성격에 따라 또 다른 언어를 학습해야만 하는 경우가 종종 발생한다. 또한 컴포넌트의 사용자는 객체 모델의 표현을 위해 사용되는 형식 라이브러리에 의존하게 되는데, 만일 형식 라이브러리의 내용이 실제 구현과는 다르게 기술된 경우, 혹은 버전 관리의 문제로 인해 설치된 컴포넌트가 삭제될 경우 애플리케이션이 동작하지 않는 DLL Hell에 빠지게 된다[2].

닷넷은 COM의 이러한 문제점들을 해결하고, 유사한 목적을 지원하면서도 서로 다른 형식으로 제공되던 내부적 기술들을 통합하고 단일화하며, 마이크로소프트와는 다른 기술을 선택한 외부 시스템이나 컴포넌트와도 효과적으로 상호 운용할 수 있는 플랫폼으로 개발되었다. 닷넷은 근래의 보편적 애플리케이션 도메인을 지원하는 표준 라이브러리 등으로 구성된 "닷넷 프레임워크(.NET Framework)"를 그 핵심으로 하고 있다[3].

2. RIA

현재 가장 주목을 받는 소프트웨어 기술은 단연 RIA이다. Jesse James가 최초 소개한 Ajax(Asynchronous JavaScript and XML), MS의 Silverlight, Sun의 JavaFX, 최근에는 Adobe의 Air[4]등이 나오면서 RIA로 통칭되고 있으며, W3C에서도 Rich Web client로 워킹그룹[5]을 명명하고 2005년 말부터 활동이 시작되었다[6]. RIA는 정적이고 불편한 웹브라우저 환경을 데스크톱 환경처럼 응답 속도가 빠르고 사용하기 편한 유려한 어플리케이션 개발 환경으로 전환하고자 하는 일련의 노력을 말한다. 현재 RIA라는 용어는 자바 애플릿(Java Applet), Ajax, 플래시(Flash), 실버라이트 등 기존 웹의 한계에 도전하고자 나온 여러 기술을 통칭하고 있다[7].

3. 플래시

플래시 프로그램은 1996년 퓨처웨이브소프트웨어(Future Wave Software)사에서 '퓨처스플래시애니메이터'라는 프로그램을 발표하였고, 이 프로그램의 가능성을 인정한 매크로미디어(Macromedia)사는 퓨처웨이브 소프트웨어사를 인수하고, 1997년 1월에 플래시를 발표하면서 플래시 애니메이션의 역사가 시작된 것이다.

웹을 통한 화려한 이미지와 역동적인 표현으로 새로운 이미지 창출 도구가 아닌 개인의 정서를 표현하는 방법으로 사용되었다. 그래서 플래시는 단순한 테크놀로지를 넘어 사회의 현상을 설명하는 코드로 이해되었다. 특히 플래시를 이용한 웹사이트가 급증하고 플래시 플레이어도 많이 보급되어 세계적으로 플래시에 대한 열풍과 관심이 고조되었다. 이와 같이 인터넷에서 더욱 더 널리 플래시가 사용되는 이유는 Animate Gif 방식이나 프로그래밍 지식을 요구하는 JAVA 애플릿 등과의 차별성 때문이다[8].

플래시는 웹 2.0의 화려하고 동적인 사용자 환경을 구현가능하게 만들었기 때문에 애니메이션을 표현하기 적합하였으나 유지 보수가 힘들었다. 그래서 플렉스로 이를 보완해 주며 개발하기 쉽게 했다. 그러나 액션 스크립트라는 언어적인 한계와 무료로 제공하는 Ajax의 라이브러리나 프레임워크에 비해 오픈소스 플랫폼이 아니라는 것과 상업용 도구라는 것이 단점으로 존재한다.

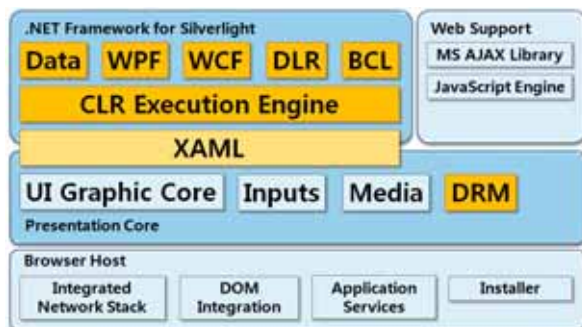
4. 실버라이트

이전에는 문자 기반 인터페이스만으로도 데이터의 분석이 충분했지만 오늘날의 사용자들은 그래픽 인터페이스에 익숙해져 있으며, 사용자 인터페이스에 대한 요구사항은 계속해서 증가하고 있다. 웹의 발전은 소프트웨어와의 편리한 상호작용을 기대하는 사용자 계층을 형성하게 되었고, 사용자들이 응용 프로그램을 사용하는 시간이 늘어날수록 해당 응용 프로그램에 대한 인터페이스 요구사항은 점점 높아만 가고 있는 추세이다. 이렇듯 점점 높아가는 인터페이스 요구사항에 부응하기 위해서는 사용자 인터페이스를 만드는 기술도 함께 발전해야 한다[9]. 더 나은 인터페이스를 통해 향상된 사용자 환경을 제공하게 되면 생산성을 높이고 우수 고객을 더 많이 확보할 수 있으며, 웹 사이트에서의 매출을 늘리는 등 다양한 효과를 얻을 수 있다[10].

마이크로소프트사는 WPF/E(Windows Presentation Foundation/Everywhere)이란 코드네임으로 실버라이트라는 웹 프레젠테이션 기술을 개발하였는데, 이 기술은 다양한 플랫폼에서 풍부한 기능과 화려한 비주얼, 인터랙티브한 실행이 가능하도록 개발되었다. 이 기술의 핵심은 XAML(eXtensible

Application Markup Language)를 토대로 한 프레젠테이션 기능이다. XAML은 텍스트 기반인 XML이기 때문에 방화벽 환경에서도 편리하게 사용할 수 있고 JavaScript를 통하여 손쉽게 이벤트 핸들러를 작성하거나 콘텐츠의 상호작용을 가능하게 할 수 있다.

미디어의 막강한 기능을 지원하고 닷넷 개발자가 접하기 쉬우며, 유지 보수 역시 쉽다. 또한 개발자와 디자이너의 손쉬운 협업을 지원한다[11]. 그림 1은 실버라이트 아키텍처 구성도이다.



▶▶ 그림 1. 실버라이트 아키텍처 구성도

그래픽에서 추출된 선, 도형, 공간, 컬러 등의 모든 개체들은 XAML로 추출해 낼 수가 있으며, 이를 그대로 화면으로 사용 가능하다. 그럼으로써 디자이너는 프로그래밍 언어나 기술을 알 필요 없이 디자인에 집중할 수 있게 된다.

현재 Mobiform Aurora, Electric Rain ZAM 3D, Microsoft Expression과 같은 도구들이 XAML을 지원하고 있으며, Adobe Illustrator에 대해서는 별도의 Plug-in이 준비되고 있다[12]. 심지어 XAML로 구성된 화면 데이터를 서버 상의 데이터베이스에 저장하고 동적으로 이를 가져와 화면을 구성할 수도 있다. XAML로 작성된 사용자 인터페이스 데이터는 텍스트 기반의 XML 데이터이기 때문에 쉽게 데이터베이스에 저장 가능하고, XAML은 동적으로 런타임 시 렌더링 될 수 있기에, 사용자 인터페이스 데이터를 데이터베이스 서버로부터 읽어와 동적으로 화면을 출력하는 것도 가능하다. 이는 마치 스마트 클라이언트가 사용자 인터페이스 데이터를 런타임 시에 서버로부터 다운로드하는 것처럼, 사용자 인터페이스 화면 자체를 웹 서비스를 통해서 가져와 동적으로 구성하는 것이 얼마든지 가능하다.

이제 디자이너는 XAML을 생성해 주는 새로운 도구를 이용해서 디자인을 해야만 한다. 그래서 마이크로소프트사에서는 익스프레션 제품군을 새로 출시하였다.

III. 실버라이트의 적용 사례

실버라이트가 발표된 이후, 해외에서는 꾸준히 그 기술을 적용해 미디어 서비스를 하는 곳이 늘어나고 있으며 MLB.com은 이미 야구 중계를 실버라이트로 서비스하고 있다. 또한 2008 베이징올림픽의 미국 내 방송 독점권을 보유한 미국의 NBC 방송사가 온라인 동영상 플레이어에 마이크로소프트의 웹 애플리케이션 '실버라이트'를 채택했다[13]. 그리고 국내에서도 최근 Mnet은 동영상 사이트를 실버라이트로 구현해 서비스하고 있으며, MBC, SBS 등의 사이트가 뉴스 동영상을 실버라이트로 서비스(NView)하는 것을 시작으로 그 관심이 꾸준히 높아지고 있다. 실버라이트의 딥줌 포토의 돋보기 기능을 이용한 (<http://deepzoom.sportschosun.com>)에 들어가면 사진 속 연예스타 얼굴을 줌인(zoom in)해서 얼굴 표정은 물론 피부 솜털까지 확인할 수 있고, 경기에 임하는 선수의 거친 호흡 모습까지 느낄 수 있어 마치 경기장에서 선수를 보는 착각이 들게 한다. 표 1은 분야별 실버라이트 적용 예이다.

[표 2] 분야별 실버라이트 적용 사례

Slide.Show[14]	MEMORABILIA[15]
Collections[16]	FOX MOVIES[17]
Tafit[18]	PCASO[19]
NEXON[20]	Mnet[21]

IV. 실버라이트의 장점

1. 크로스 브라우저, 크로스 플랫폼

실버라이트는 크로스 플랫폼, 크로스 브라우저를 지원하는 브라우저 플러그인이다. 현재 Windows와 Mac을 공식적으로 지원하고 있고 추가적으로 Linux에 대한 지원도 제공되며 브라우저는 IE는 물론 Firefox, Safari, Opera의 4대 브라우저를 모두 지원한다. 운영체제는 윈도우와 애플의 Mac OSX를 지원한다.

2. 경량화

실버라이트의 브라우저 플러그인은 경량화를 지향한다. 현재 실버라이트 1.0 베타 버전의 런타임은 1.38MB이다. 참고로 플래시 9.0의 런타임은 2.2MB이다.

3. 멀티미디어 지원

비디오는 윈도우 미디어 콘텐츠인 WMV와 WMA 포맷을 지원하며, HDTV 수준의 비디오 해상도도 지원한다. 오디오로는 MP3 포맷을 지원한다.

4. WPF, XAML 지원

WPF는 윈도우 비스타의 닷넷 프레임워크 3.0에 포함된 기술인데, 디자인을 표현하는데 있어 XAML을 이용한다. 플래시와 마찬가지로 벡터 기반이기 때문에 확대하거나 축소를 하더라도 이미지의 손실 없이 부드럽게 표현되는 것도 빼놓을 수 없는 특징이다.

5. 닷넷 프로그래밍 모델 지원

실버라이트 1.1버전부터는 닷넷 프로그래밍 모델도 지원한다. CLR(Common Language Runtime) 전부를 지원한다. C#과 비주얼 베이직은 기존에 있던 언어이지만 새롭게 동적 언어인 IronPython, Ruby, Managed JScript 지원을 추가했다.

6. 기존 응용 프로그램 지원

실버라이트는 자바스크립트와 서로 연동할 수 있다. 자바 스크립트 뿐만 아니라 ASP.NET, Ajax와도 쉽게 연동할 수 있다.

7. 압축 지원

XAML 파일은 텍스트 기반이기 때문에 그 용량이 상당히 크다. 이를 그대로 웹에 올리려면 용량이 문제가 된다. 플래시의 경우도 개발 시에는 FLA 파일 포맷을 사용하지만 실제 배

포할 때에는 압축된 SWF 포맷을 사용한다. 실버라이트도 그림2의 압축 기술을 지원한다. 실버라이트에서 지원하는 압축 기술인 ".xap"파일은 클라이언트 다운로드 사이즈를 줄이기 위해서 표준 zip 압축 알고리즘을 사용했다.



▶▶ 그림 2. 실버라이트의 압축 패키징

단일 XAP파일로 패키징 하여 배포하며 단순한 ZIP형태의 표준 압축으로 투명한 구조를 지니고 있으며 Assembly(.dll)와 Content(**)를 포함 한다. 그림 3은 실버라이트의 파일 압축률을 보여준다.

폴더	이름	압축 크기	크기	압축률
MySilverlight.xap	AppManifest.xaml	343	563	38%
	MySilverlight.dll	3,120	6,144	49%
	System.Windows....	73,502	269,344	72%
	System.Windows....	57,811	195,632	70%

▶▶ 그림 3. 실버라이트 파일 압축률

V. RIA 제품 비교

실버라이트의 그래픽 렌더링 성능에 대해 벤치마킹하여 보았다. 여러 개의 공을 표시하고 렌더링 하면서 초당 몇 프레임이 나오는지 계산해 보았다. 표 2는 브라우저별로 DHTML, Flex, Silverlight 1.1과 Silverlight 2.0을 성능 테스트한 결과이다. 결과를 보면 실버라이트의 렌더링 성능이 플렉스에 비해 월등히 높음을 확인할 수 있다.

[표 2] 성능테스트 화면 및 결과

브라우저	DHTML	Flex	Silverlight1.1	Silverlight2.0
익스플로러 7.0	64	64	64	814
파이어폭스 3.0	91	191	99	840

VI. 결론

실버라이트로 할 수 있는 것들은 무궁무진 하다. 강력한 미디어 기능을 지원하기 때문에 미디어 서비스와 관련된 웹 애플리케이션의 UI를 구성하는데 가장 많이 사용된다. 또 UX를 고려한 인터랙티브한 UI를 구성하는데 많이 이용될 것이다. 크로스 브라우저, 크로스 플랫폼이 완벽하게 지원이 된다면 우리가

가지고 다니는 휴대폰, PMP, UMPC 등의 다양한 종류의 디바이스에도 실버라이트를 이용해서 보다 훌륭한 UI를 구성하는데 활용 될 것으로 보인다. 기존 브라우저의 한계를 뛰어 넘는 자유로운 사용자 인터페이스는 새로운 웹 환경을 만들어 나가는데 큰 몫을 할 것이다.

감사의 글

본 연구는 정통부 및 정보통신연구진흥원의 지원을 받아 수행된 연구결과임(08-기반-13, 정보통신연구기반조성사업)

■ 참고 문헌 ■

- [1] Don Box, Essential Com, Addison-Wesley International, 1997
- [2] <http://msdn2.microsoft.com/en-us/library/ms811694.aspx>
- [3] 김명호, "마이크로소프트 컴포넌트 기술의 발전과 동향", 한국정보과학회지, 제24권, 제11호, pp.5-6, 2006.
- [4] labs.adobe.com/wiki/index.php/Apollo
- [5] www.w3.org/2006/rwc
- [6] 박차라, "RIA 기반 개인화 검색을 위한 Widget 응용의 구현", 한국정보과학회, 정보과학회논문지, 제 13권, 제 6호, pp. 402-406, 2007.11.
- [7] <http://blog.naver.com/minajung25/120045550908>
- [8] 황연주, 조영주, "초등미술과 수업에서 플래시 애니메이션의 지도 방안 연구", 한국교육대학교 미술교육학회, 미술교육연구논총, 제19집, pp. 165-189, 2006. 8
- [9] L. Moroney, msdn2.microsoft.com/en-us/library/bb190632.aspx
- [10] <http://msdn2.microsoft.com/en-us/library/Aa663364.aspx>
- [11] <http://www.hoons.kr/>
- [12] <http://blog.daum.net/jjogume/853067>
- [13] <http://www.zdnet.co.kr/news/internet/etc/0,39031281,39168361,00.htm>
- [14] <http://www.vertigo.com/SlideShow.as>
- [15] <http://memorabilia.hardrock.com>
- [16] <http://www.loc.gov/exhibits/earlyamericas/readingartifacts>
- [17] <http://silverlight.net/fox>
- [18] <http://www.tafiti.com>
- [19] <http://www.pcaso.kr>
- [20] <http://channel.nexon.com/>
- [21] <http://tvdeep.mnet.com/>