

Ajax 를 이용한 웹기반 개발도구의 구조*

류성태, 이은석
성균관대학교 전자전기컴퓨터공학과
e-mail : xeno0718@gmail.com, eslee@ece.skku.ac.kr

Architecture of Web-based Development Tool Using Ajax

Sung-tae Ryu, Eunseok Lee
Dept. of Electrical and Computer Engineering, Sungkyunkwan University

요 약

웹개발 기술의 빠른 발전에 따라 컴퓨팅 패러다임이 변화하고 있고, 이에 따라 클라우드 컴퓨팅이라는 새로운 시대가 열리고 있다. 또한 웹으로 공유된 다양한 정보와 서비스들은 사용자들에 의해 조합되어 매쉬업이라는 웹어플리케이션 형태로 등장하고 있다. 본 논문에서는 웹에 기반하여 접근성을 높이고 오픈 API 의 사용이 용이하며 다양한 플랫폼으로 코드를 배포할 수 있는 개발도구의 구조를 제안한다. 또한 이 구조를 이용하여 Uizard 라는 개발도구를 구현하고 이것을 이용한 지도 매쉬업의 실제 개발 사례를 평가하여 제안하는 웹기반 개발도구 구조의 높은 효율성을 확인하였다.

1. 서론

오늘날 웹개발 기술의 빠른 발전은 컴퓨팅 패러다임에 새로운 변화를 가져오고 있다. 이러한 변화에 따라 사용자는 웹과 더 강력한 상호작용을 할 수 있게 되었다. 특히 Ajax(Asynchronous JavaScript and XML) 기술은 웹이 뛰어난 사용자 경험을 제공하게 해준다. 이러한 변화는 클라우드 컴퓨팅이라는 새로운 시대로 이어지고 있으며, 특히 구글의 Chrome OS 와 같은 웹 OS 계획, HTML5 출시와 JavaScript 의 괄목할만한 성능 향상이 이를 뒷받침해준다.

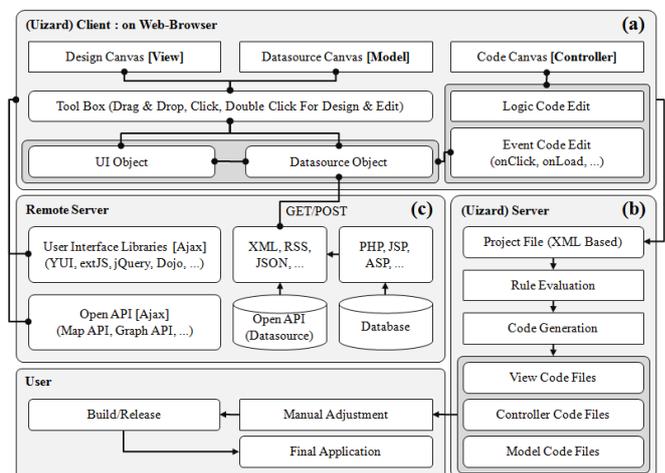
또한 오픈 소스로 시작된 공유의 흐름은 현재 오픈 API 로 활성화되었고 향후에는 오픈 서비스라는 개념이 등장하리라 예상된다. 이러한 흐름은 사용자들의 필요에 맞게 조합된 매쉬업이라는 웹어플리케이션을 등장시켰다. 하지만 매쉬업을 개발하기 위해서는 전문적인 지식과 개발 환경을 필요로 하여 접근성이 떨어지는 것이 사실이다.

본 논문에서는 웹환경에 기반한 개발도구의 구조를 제안한다. 이 구조는 다양한 라이브러리와 오픈 API 를 지원하고 다양한 플랫폼으로 배포 가능하게 설계되었다. 그리고 이러한 구조로 개발된 웹기반 개발도구인 Uizard¹도 소개한다. Uizard 는 Ajax 기술로 만들어진 SaaS(Software as a Service)로서의 개발도구로서 뛰어난 접근성을 가지며, 별도의 설치 없이 플랫폼의 제한을 받지 않는 일관된 개발환경을 제공한다. 마지막으로 제안 구조의 효율성을 입증하기 위해 지도 매쉬업을 실제 Uizard 로 개발하여 평가하였다.

2. 웹기반 개발도구의 구조

제안하는 웹기반 개발도구의 구조는 크게 3 개의 레이어로 이루어진다. (그림 1)의 (a)는 클라이언트 레이어로서 사용자의 웹 브라우저를 통해 개발 환경을 제공한다. (그림 1)의 (b)는 서버 레이어로서 프로젝트 파일을 규칙에 따라 결과를 생성한다. (그림 1)의 (c)는 원격 서버 레이어로서 웹환경에 존재하는 다양한 유저 인터페이스 라이브러리 및 데이터소스를 클라이언트 레이어에 제공한다.

클라이언트는 총 3 개의 캔버스로 이루어진다. 디자인 캔버스는 유저 인터페이스를 디자인할 수 있는 캔버스이다. 툴박스에서 드래그앤드롭을 이용하여 쉽게 유저 인터페이스 객체를 배치할 수 있다. 데이터소스 캔버스는 데이터소스 객체를 관리하는 캔버스이다. 원격지 서버에서 제공하는 XML, RSS, JSON 데이터를 객체와 GET/POST 와 같은 Http Request 방식으로 연결



(그림 1) 웹기반 개발도구의 구조

* 이 논문은 2009년도 정부(교육과학기술부)의 재원으로 한국과학재단의 지원을 받아 수행된 연구임(No. 2009-0077453)

¹ User Interface Wizard. 현재 <http://www.uizard.org/>에서 Experimental Version 인 Uizard v0.9 의 실행이 가능하다. Mozilla FireFox 와 Google Chrome 을 지원한다.

한다. 마지막으로 코드 캔버스는 코드 편집 환경을 제공한다.

서버는 결과 코드의 생성 및 배포를 담당한다. 클라이언트는 UI 변경 및 코드 변경 이벤트 발생 시에 실시간으로 서버와 통신하여 XML 기반의 프로젝트 파일을 갱신해두는데, 서버는 이 파일과 미리 설정된 배포 규칙을 이용하여 결과물을 생성한다. 프로젝트는 배포 규칙에 따라 다양한 플랫폼으로 배포될 수 있다.

원격 서버는 웹에 존재하는 다양한 유저 인터페이스 라이브러리와 오픈 API, 다양한 데이터소스들을 갖고 있는 다양한 원격지 서버들을 하나로 묶은 개념의 레이어이다. 유저 인터페이스 라이브러리와 오픈 API의 경우 클라이언트에 연결된다. 데이터소스들은 XML, RSS, JSON의 형태로 데이터소스 객체에 직접 연결된다.

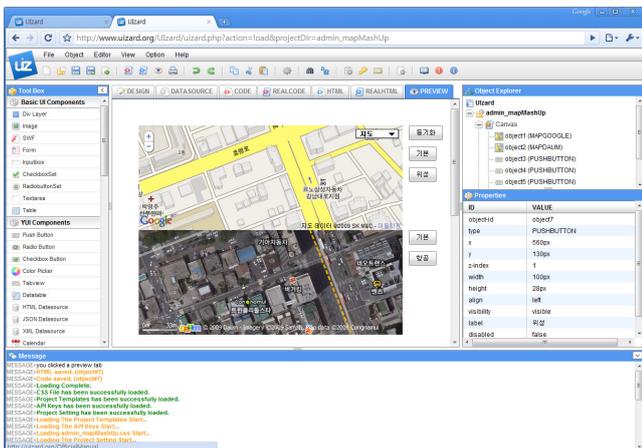
3. MVC 패턴의 적용

UIzard의 개발 환경에는 MVC 패턴이 적용된다. 디자인 캔버스는 View 계층을 개발하는 환경을 제공하며, 코드 캔버스는 Controller 계층을 개발하는 환경을, 데이터소스 캔버스는 Model 계층을 개발하는 환경을 제공한다. 그리고 이렇게 개발된 결과물은 XML 파일로 기록되어 배포 확장성을 유지한다.

배포 규칙에 따라 다양한 플랫폼에 알맞은 코드로 배포가 가능하며, 현재 구현된 UIzard는 일반 웹 플랫폼과 소셜 API 기반의 위젯 플랫폼에 대한 배포 규칙을 지원한다. 웹 플랫폼 배포 규칙을 사용할 경우 최종 결과물에서는 View 계층은 HTML과 CSS 파일로 생성되어 나타나며, Model과 Controller는 JavaScript 파일로 생성되어 나타난다.

4. 구현 및 평가

제안하는 구조를 적용하여 웹기반 개발도구인 UIzard를 구현하였다. 개발도구의 유저 인터페이스 및 사용할 수 있는 유저 인터페이스 객체는 YUI 라이브러리[3]를 사용하였다. 제안 구조의 효율성을 검증하기 위하여 지도 매쉬업 개발을 평가한다. (그림 2)는 UIzard를 이용하여 실제 개발한 지도 매쉬업이



(그림 2) 개발된 지도 매쉬업의 정상작동 확인

<표 1> 지도 매쉬업 코드의 크기 [Bytes]

	실제작성	자동생성	최적화
View (HTML/CSS)	0	3,460	3,460
Controller (JavaScript)	381	2,841	2,123
Model (JavaScript)	154	154	154
전체	535	6,455	5,737

다. 이것은 서로 다른 지도 API를 이용하여 출력하며, 드래그, 줌 이벤트 시에 좌표를 동기화한다. 그리고 지도의 타입을 바꿀 수 있는 버튼을 주변에 배치한다.

해당 매쉬업이 얼마나 효율적으로 작성되었는지 평가하기 위해 작성한 코드와 생성된 코드, 수동으로 최적화한 코드의 크기를 비교한다. 정확하고 공정한 비교를 위해 코드의 내용은 유지하면서 공백 및 줄바꿈을 제거하는 방법 등을 통해 HTML/CSS/JavaScript 파일의 용량을 최대한 줄여주는 YUI Compressor[4]를 이용하여 파일을 압축하여 그 크기를 평가하였다.

<표 1>과 같이 기능을 구현하는 데 직접 입력한 코드의 크기는 535 바이트이다. 실제 사용을 위해 UIzard가 자동으로 생성한 코드는 총 6,455 바이트이다. 자동으로 생성된 코드를 수동으로 최적화할 경우, 크기를 총 5,727 바이트로 줄일 수 있었지만, 생성된 코드와 큰 차이를 보이지는 않았다. 자동으로 생성된 코드와 수동으로 최적화한 코드는 모두 웹브라우저에서 정상 작동함을 확인하였다.

5. 결론 및 향후 연구

웹기반 개발도구는 향후 클라우드 컴퓨팅 환경에서 강력한 개발도구가 될 가능성이 높다. 본 논문에서는 웹기반 개발도구의 구조를 제안하고 이러한 구조로 실제 개발된 UIzard라는 개발도구를 소개하였다. 또한 MVC 패턴이 적용된 구조를 통해 Controller, Model 부분의 코드만을 작성함으로써 효율적이고 빠른 개발이 가능하다는 것을 지도 매쉬업 개발 사례를 통해 확인하였다.

UIzard의 궁극적인 목표는 Eclipse와 같이 다양한 플랫폼과 프로그래밍 언어를 지원하여 웹을 통해 통합개발환경(IDE)를 제공하는 것이다. 이를 위해 향후에는 UIzard v2.0의 개발을 통해 현재 웹기반 개발도구의 구조에 UML과 MDA(Model-Driven Architecture)를 적용하여 복수의 플랫폼에 대해 효율적으로 코드를 배포하는 방법에 대해 연구할 계획이다.

참고문헌

- [1] Tao Jiang, Jing Ying, Minghui Wu, Canghong Jin, "A Method for Model-driven Development of Adaptive Web Applications", IEEE, 2008
- [2] F. Piero and P. Paolo, "Model-driven development of Web applications: the AutoWeb system", ACM Transactions on Information Systems, 2000, 18(4), 323-382.
- [3] "Yahoo User Interface Library", <http://developer.yahoo.com/yui/>, 2009
- [4] "YUI Compressor", <http://developer.yahoo.com/yui/compressor/>, 2009