

RIA기술을 적용한 웹 지도 서비스의 특징 연구

The Characteristic of Web Map Service Using RIA Technologies

김 문 기* 고 준 환**
Moongie Kim June Hwan Koh

요 약 최근 국내외에서는 민간기업 뿐만 아니라 관공서에서도 웹 지도 서비스가 활발하게 이루어지고 있는 상황이다. 플랫폼으로써 데스크 탑에서부터 스마트 폰까지 다양하며, 이에 사용되는 기술도 계속 발전하고 있는 추세이다. 또한 사용자들에게 현재 서비스 중인 공간정보를 사용하여서 매쉬업 할 수 있도록 다양한 API를 제공하고 있다. 본 논문에서는 웹 지도 서비스에서 최근 유행하고 있는 RIA기술을 적용하여 각 언어별로 지도 서비스에서 가장 많이 사용하는 기능을 구현하였다. 각 웹 브라우저별로 사용자가 느끼는 속도에 대해서 실험 및 분석을 실시하였으며, 분석결과 JavaScript, Silverlight, Flex로 구현된 기능마다의 특징이 발견되었다. 실제 지도 서비스중인 서울시GIS포털시스템의 생활지도에서 직접 테스트하였다. 종합적인 결과 본 연구의 환경에서는 Silverlight의 성능이 다소 우수한 것으로 나왔다.

키워드 : RIA, 지도 데이터 매쉬업, 웹 지도 서비스

Abstract Recently, Web map service is actively accomplished in both private companies and public offices. As a platform, it exists variously from desktop to smartphone. The technology being used has a trend to develop continuously. Also, Web map system provides various open API for people who use geospatial service and data mashup. In this paper, RIA technology which is popularly used recently in web map service will be applied to introduce the functions different language map services are mostly using. Based on users' feeling about different web browser's speed, test and analysis have been accomplished. The result is that there are different characteristics according to different functions such as JavaScript, Silverlight, Flex. Actual test has been personally carried out on map service in Seoul GIS portal system. The comprehensive conclusion is that Silverlight has more outstanding function compared with other RIA techniques under the test environment.

Keywords : RIA, Map Data Mashup, Web Map Service

1. 서 론

웹 지도 서비스는 기존의 클라이언트 서버 시스템이 가지는 공간적 한계를 벗어나 인터넷을 통하여 어디서든지 자유롭게 지도 및 각종 지도와 관련한 정보를 검색하고 활용할 수 있도록 지원해준다. 특히 인터넷 사용자들은 지도 구동을 위한 GIS 엔진을 구매하지 않고도 웹 브라우저 등을 통해 지도 관련 애플리케이션을 사용할 수 있을 뿐만 아니라, 이러한 지도서비스를 기반으로 다양한 서비스가 제공되고 있어서, 향후 웹 지도 서비스의 활용은 더욱

가속화 될 전망이다[9, 10].

그림 1은 매쉬업에 대해서 유명한 웹 페이지인 www.programmableweb.com에서 제공한 결과이며, 매쉬업 순위 중 지도(mapping)를 이용한 경우가 33%이상인 것을 보여준다. 최근 민간 및 관공서에서 지도 서비스는 급속히 대중화된 인터넷 서비스 환경을 이용하여 교통, 문화, 관광 등의 콘텐츠와 결합하여 함께 서비스 중이며, 전 세계적으로 지도와 연계한 서비스가 주를 이루고 있는 실정이다 [16].

* 서울시립대학교 공간정보공학과 박사과정 gis@seoul.go.kr(교신저자)
** 서울시립대학교 공간정보공학과 교수 jhkoh@uos.ac.kr



그림 1. 지도를 이용한 매쉬업 순위

1.1 국내외 지도 포털사에서의 RIA 이용현황

현재 인터넷 서비스의 발전과 클라이언트의 데스크 탑 성능이 향상되면서, 서비스가 어플리케이션 중심이 되어 편리한 사용자 인터페이스와 빠른 응답성을 요구로 하고 있다. RIA는 클라이언트에서 데이터의 가공과 사용자 인터페이스를 처리함으로써 빠른 응답속도 및 편리한 사용자 인터페이스를 제공한다. 국내인 경우 대표적으로 다음지도에서는 현재 JavaScript 지도 API를 네이버지도에서는 JavaScript와 Flash API를, 해외인 경우 대표적으로 Google Map에서는 JavaScript, Flash, Google Earth API를 MS사의 Bing Map에서는 Silverlight와 Ajax API등을 제공 중에 있다. 그림 2는 Google Map과 네이버지도에서 제공하는 RIA 관련 지도API를 보여준다.



1.2 선행연구 내용

1.2.1 국내 연구동향

RIA기술을 이용한 국내 연구로써 이재일 외 2인은 무선 인터넷을 사용하는 모바일 환경에서 RIA 기반 지도데이터 서비스를 실험 및 분석하였다. 기존의 웹 환경인 WMS(Web Map Service)와 RIA 기술인 Flex를 사용하여 지도정보데이터 서비스에 대해서 비교 성능을 실시하였다. Flex로 개발하여 테스트하였으며, 클라이언트가 웹 서버로 지도정보 데이터를 요청하면 웹 어플리케이션 자체에 사용자가 요구한 지도정보데이터를 포함하여 한 번에 전송하는 지도정보데이터 서비스를 제안하였다. 하지만, 프로토타입의 환경에서 서울시 강동구와 강남구의 지역에 대해서 Flex 기술을 적용하였다[11].

1.2.2 해외 연구동향

Liu Yang 외 2인은 REST기반의 웹 GIS와 Silverlight를 결합하여 상해지역에 대한 전자지도(Electronic Map) 페이지를 개발하였다. 사용자에게 Silverlight의 특징인 고급스럽게 버튼 이벤트가 나타나거나 사라지고, 부드럽게 사진이나, 동영상 이 재현되게 구현하는 테스트를 하였다. 중국의 바이두 지도(Baidu Map)나 구글맵과는 달리 공간질의, 객체 그리기, 버퍼 분석시에 리 플래쉬하는 과정이 필요 없어서, 사용자가 기다리는 시간을 줄여주어서 만족감을 준다는 결론을 도출하였으나, 테스트서버 환경에서 단순한 기능에 대해서 Silverlight로 구현하여 실험을 하였다[2].

Tian외 2인은 비디오감시시스템을 웹 GIS와 ArcGIS Server Flex API 기술을 결합하여 개발한

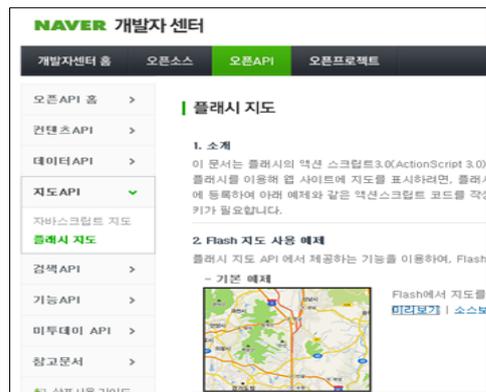


그림 2. Google Map(좌)과 네이버 지도(우)에서 제공중인 RIA 관련 지도API

연구를 하였다. 기존의 웹 GIS의 기술력과 비교하여서 Flex는 웹 애플리케이션에서 사용자에게 만족스러운 느낌을 제공하였고 모니터와 지도가 결합한 시스템에서 각종 기능에 대해서 조작이 쉽다는 연구결과가 나왔다. 하지만 웹 지도의 전반적인 기능이 아닌 비디오감시시스템을 위한 동영상 기능에 대한 분석을 하였다[3].

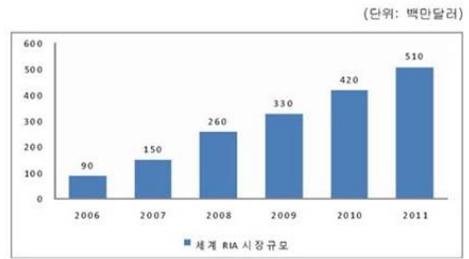
1.3 개선방안

기존에 연구된 방법들을 분석한 결과 대부분이 실서버가 아닌 프로토타입의 환경에서 실험을 하였으며, RIA의 일부 언어로 지도서비스에 대한 간단한 기능을 구현하여 테스트한 점이 아쉬웠다. 이에 본 연구에서는 실제로 운영중인 서울시GIS포털서비스의 생활지도의 환경에서 RIA의 주요 기술인 JavaScript, Flex, Silverlight로 웹 지도 서비스에서 많이 사용하는 기능등을 구현하였다. 또한 구현된 기능 등의 특징을 웹 브라우저 별로 분석하여, 신기술인 RIA를 사용하여 웹 지도 서비스를 구축하고자 하는 사업에서 참조할 수 있는 객관적인 실험결과를 제공하는데 있다.

2. RIA관련 기술조사

인터넷 애플리케이션(RIA, Rich Internet Application)이라는 용어는 2002년도 매크로미디어사에서 배포한 백서에서 세계 최초로 그 개념이 소개되었다[1, 14]. 별도의 설치가 필요 없는 웹 브라우저 기반의 애플리케이션 배포의 장점과 서버 측 웹 서비스와의 연동, 마크업 언어 기반의 선연적 애플리케이션 구성 등은 유지하면서 데스크 탑 애플리케이션과 대등한 사용자 경험을 주는 것을 목표로 하는 기술이다[8]. 즉, 웹이라는 통신의 한계와 HTML로 표현할 수 없는 한계를 보완해 주기 위한 기술로 이해하면된다. RIA는 자바 애플릿, AJAX(Asynchronous JavaScript and XML), 어도브사의 플래쉬(Flash), 마이크로소프트사의 실버라이트(Silverlight) 등을 주로 사용하여 개발한다. 그림 3은 세계적으로 사용되는 RIA의 세계 시장규모를 보여주며, 날로 RIA기술의 사용이 늘어남을 알 수 있다[6, 21].

RIA를 통해서 사용자는 풍부한 UI를 제공받고, 서버가 아닌 클라이언트의 자원을 사용하여 페이지의 필요한 부분의 데이터만을 서버에 요청하기 때



구분	2006	2007	2008	2009	2010	2011
세계 RIA 시장규모	90	150	260	330	420	510

자료: Zapthink, 2006

그림 3. RIA의 세계 시장규모

문에 불필요한 네트워크 부하를 막을 수 있다. 표 1은 기존 HTML기술과 RIA기술의 특징을 설명한다.

표 1. 기존 HTML과 RIA의 기술특징

기존 HTML 기반 구성의 한계	기존 Flash 구성의 한계
페이지 전체의 구성요소를 로딩해야 함	파일을 한번에 로딩하기 때문에 시간이 많이 걸림
프로세스에 따라 연결된 페이지들을 모두 로딩해야 함	검색이나 DB연동이 되지않는 단순 멀티미디어로 사용됨
멀티미디어 콘텐츠의 수용에 제한이 있음	HTML에 비해 제작방법이 복잡해 유지보수가 까다로움
Rich Internet Application 기술특징	
플래시 인터페이스를 사용해 전체페이지가 아닌 필요한 부분만 로딩해 페이지 부하가 적고 빠른 비주얼적 다이나믹 검색과 DB연동이 가능하고 콘텐츠를 재가공 없이 모바일에서 사용가능 복잡한 단계별 페이지 구조를 단일 인터페이스에서 처리 가능	

표 2는 RIA 기반 웹 사이트와 인터랙션 과정에서 사용자 경험 중 사용자가 어떻게 느끼는가에 관한 질적인 측면에서 긍정적 경험과 부정적 경험을 설명한다[7].

표 2. 긍정적 및 부정적 사용자 경험

긍정적 경험	<ul style="list-style-type: none"> 인간과 환경이 활발하게 인터랙션하고 후속되는 경험을 촉진함으로써 계속되는 성장을 조장하는 경험 만족, 충성도 향상, 성공
부정적 경험	<ul style="list-style-type: none"> 상호작용이 활발하지 못하고 후속 경험이 성장을 방해하는 경험 목적에 이루지 못함, 부정적 반응

본 연구에서는 기존부터 Web GIS 지도서비스에서 많이 사용하는 JavaScript와 RIA 기술인 Flex 및 Silverlight에 대한 원리 연구와 성능실험을 하였다. 본 장에서는 이와 관련된 기술을 조사하여 설명하였다.

2.1 JavaScript

JavaScript 기반의 웹 지도 서비스는 가장 구현하기 쉽고, 가장 널리 사용되어지고 있다. HTML과 같이 웹 브라우저에서 직접 해독이 가능하기 때문에 소스의 보안이 어렵다는 단점이 있지만 보다 많은 사람들이 사용할 수 있어 사용자의 폭을 넓힌다는 장점이 있다. 그림 4는 JavaScript 기반의 웹 지도 서비스가 실행되는 원리를 보여준다. 하지만 RIA처럼 사용자 경험을 제공하지 못한다. JavaScript로 구현된 지도 서비스는 사용자(Client)가 웹 브라우저를 통해서 지도를 호출하면 웹브라우저는 JavaScript / Html을 바로 해석하고 REST서비스를 이용하여 GIS Server와 통신하고 지도를 컨트롤 할 수 있도록 해준다[15].

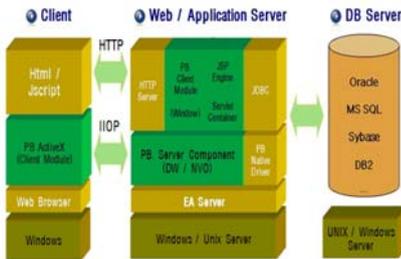


그림 4. JavaScript의 실행 원리

2.2 Silverlight

Silverlight는 최근 웹 시장에서 Microsoft사가 Flex를 따라잡기 위해서 야심차게 내어 놓은 최신 RIA 기술로써, 강력하고 효율적인 미디어 기능과 .Net Framework의 기능들을 사용하여 보다 높은 품질의 코드 작성이 가능하다. 하지만 이미 넓은 시장과 다양한 리소스를 보유하고 있는 Flex에 비해서 디자이너와 프로그래머들의 개발 경험이 짧아, 개발을 진행함에 있어서 참고 할 만한 리소스가 다소 부족한 상황이다. 그림 5는 Silverlight 기반의 웹 지도 서비스가 실행되는 원리를 보여준다[20].

Silverlight로 구현된 지도 서비스로 사용자가 접

속하면 Web/Application 서버에서 XAML (Extensible Application Markup Language, 확장 응용 프로그램 마크업 언어) 코드를 xap로 컴파일해서 사용자 PC에 전송되어 Silverlight 런타임이 이를 보여주고 REST서비스를 이용하여 GIS Server와 통신하여 지도를 컨트롤 할 수 있도록 해준다.



그림 5. Silverlight의 실행 원리

2.3 Flex

Flex의 개발 툴인 Flex Builder가 이클립스 기반이라 자바나 서버 개발자들이 자연스럽게 활용할 수 있으며 XML태그(xml)와 액션스크립트로 코딩되므로 html 코딩을 이해하는 수준정도면 쉽게 배울 수 있는 장점이 있다. 다양한 유저 인터페이스를 기본 컴포넌트로 제공하므로 개발자의 코딩량이 줄어들어 개발속도를 향상할 수 있고, 다양한 방법으로 컴포넌트를 만들 수 있어 코드의 재사용성이 높아 개발속도를 향상할 수 있다[17]. 또한, 기존 시스템 및 다른 애플리케이션과 연동할 수 있는 다양한 방법을 제공한다. 그림 6은 Flex 기반의 웹 지도 서비스가 실행되는 원리를 보여준다[5].

Flex로 구현된 지도 서비스로 접속하면 웹 어플리케이션 서버에서 설치된 Flex 서버는 2004년 3월 매크로미디어사가 도입한 XML 기반의 사용자 인터페이스 생성 언어인 MXML(Magic eXtensible Markup Language) 코드를 SWF로 컴파일해서 사용자 PC에 전송되어 Flash 플레이어가 이를 보여주고 REST서비스를 이용하여 GIS Server와 통신하여 최종적으로 지도를 컨트롤 할 수 있도록 해준다.

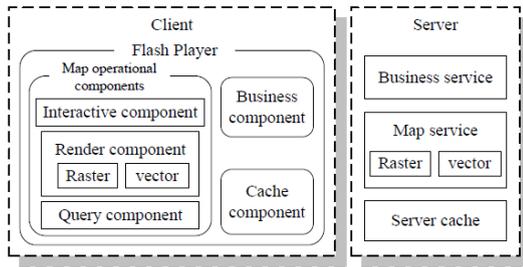


그림 6. Flex의 실행 원리

3. 연구내용

3.1 연구 대상지

기존부터 웹 지도 서비스에 많이 사용되는 Java Script와 RIA 기반의 Silverlight 및 Flex의 성능 비교와 국내외 선행연구와의 객관적인 성능 비교분석을 위해서 실제로 운영중인 서울시GIS포털시스템(gis.seoul.go.kr)의 생활지도를 대상으로 시험을 하였다. 그림 7은 본 연구에서 사용되는 서울시 및 경기도 지역에 대한 지도 서비스의 모습을 보여준다 [13].



그림 7. 서울시GIS포털시스템의 생활지도와 연구대상지

실험에 사용되는 생활지도의 레이어에 대한 명칭 및 개수는 아래 표 3과 같으며, 데이터의 크기는 총 2.69GB이다.

3.2 지도서비스를 위한 시스템 환경의 구성

웹 서버는 데이터티어, 애플리케이션티어, 프리젠테이션티어로 모두 3단으로 구성되어 있다.

표 3. 레이어의 명칭 및 개수

레이어명	개수
명칭관련	51
도로관련	31
경계관련	6
지하철관련	8
건물관련	3
공원시설	3
자연관련	13

서울시 GIS포털시스템은 ESRI사의 ArcGIS Server 9.3 제품을 사용 하였으며, 그림 8과 같이 데이터티어에서는 ArcGIS Desktop을 이용하여 지도데이터를 제작 하였다. 애플리케이션티어는 웹 서버와 GIS서버로 구성되어 지도서비스를 호스팅하게 된다. 프리젠테이션티어는 애플리케이션티어에서 제공되는 지도 서비스를 각종 디바이스로 사용하게 된다. 이때 각각의 언어로 개발된 시스템은 ArcGIS Server에서 제공하는 REST서비스를 이용하여 지도와 통신하여 사용자들이 지도 서비스를 이용할 수 있다[4].

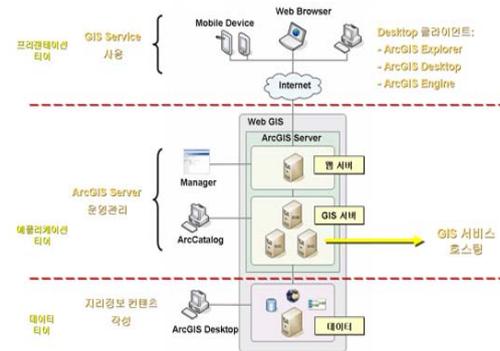


그림 8. 지도서비스를 위한 시스템 환경

3.3 서버의 스펙

본 실험에서 사용된 웹 서버와 GIS엔진이 탑재된 GIS 서버에 대한 자세한 스펙은 아래 표 4와 같으며, 이중화로 구성이 되어있다.

3.4 연구방법

본 연구에서는 JavaScript방식과 RIA방식인 Silverlight 및 Flex와의 성능을 비교 및 분석하는 실험으로써, 성능분석을 위하여 같은 대상지역에 대해서

표 4. 서버의 스펙

	웹서버	GIS서버
MODEL	HP ML570	HP ML380
CPU	3.66G*4	DUAL 3.0G
RAM	4G	4G

동일한 서버에 대해서 실제로 운영중인 서울시GIS 포털시스템의 생활지도 서비스를 대상으로 실험 하였다. 테스트하는 기능은 모두 9가지로써 이미지 지도(Tiled Map), 기본 컨트롤인 확대(Zoom In), 축소(Zoom Out), 전체 화면확대(Full Extent), 이동(Pan), 네비게이션 스타일, 지도상에 정보 창보기, 지도상에 해당위치에 마우스 오버시 정보표시, 검색창 기능이다. 표 5는 본 실험에서 사용되는 기능에 대한 설명을 보여준다.

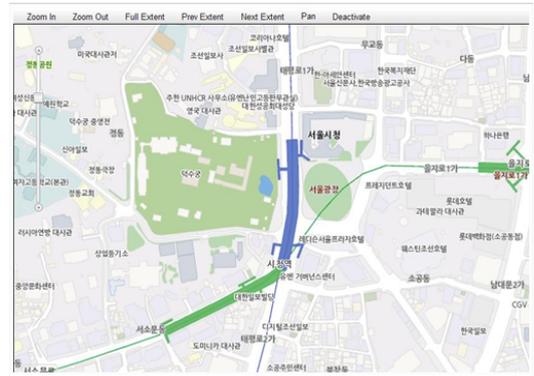
표 5. 각 기능에 대한 설명

기능	설명
Tiled Map	Tiled Map을 사용하여 지도를 표시
기본 컨트롤	확대, 축소, 이동, 전체화면을 사용하여 지도를 표시
네비게이션 스타일	네비게이션 스타일을 변경
지도상에 정보 창보기	해당좌표에 정보를 보기
지도상에 해당위치에 마우스 오버시 정보표시	지도상에서 사용자에게 의해서 일어나는 마우스 오버시 해당 위치의 정보를 조회
검색창 기능	지도 레이어의 내용을 조회하여 검색

각 API로 구현된 기능에 대해서는 사용자들은 다양한 웹 브라우저를 사용하며, 이에 따라서 서로 다른 지도 서비스를 체감한다. 최근 사용자들은 다양한 웹 브라우저를 사용하고 있으며, 보다 객관적인 결과제공을 위해 본 실험에서 사용된 웹 브라우저로는 Internet Explorer 8.0.6, Mozilla Firefox 3.6.7, Google Chrome 10.0.6이며, 각 기능에 대해서 로딩 속도 측정 실험을 하였다.

그림 9, 그림 10, 그림 11은 각각 ArcGIS Server 9.3에서 제공하는 JavaScript API, Silverlight API, Flex API를 사용하여 개발된 기본 컨트롤(확대, 축소, 전체화면, 이동)에 대한 기능 구현의 모습과 작

성된 코드를 보여준다. 이외 다른 기능에 대한 구현은 서울시GIS포털시스템에 자세히 설명되어있다.



```

<script type="text/javascript"
src="http://serverapi.arcgisonline.com/jsapi/arcgis/?v=1.6">
</script>
<script type="text/javascript">
dojo.require("esri.map"); dojo.require("esri.toolbars.navigation");
dojo.require("dijit.form.Button"); dojo.require("dijit.Toolbar");
var map, navToolbar;
function init() {esriConfig.defaults.map.sliderLabel = null;
//지도 생성 map = new esri.Map("map");
map.addLayer(new esri.layers.ArcGISTiledMapServiceLayer
("http://115.84.164.197/ArcGIS/rest/services/SeoulLife/MapServer"));
//툴바 네비게이션
navToolbar = new esri.toolbars.Navigation(map);
dojo.connect(navToolbar,
"onExtentHistoryChange", extentHistoryChangeHandler);
} function extentHistoryChangeHandler() {
-----중략-----
} dojo.addOnLoad(init); </script> </head>
//지도 툴바 버튼 생성
<div id="navToolbar" dojoType="dijit.Toolbar">
<div dojoType="dijit.form.Button" id="zoomin"
iconClass="zoominIcon"
onClick="navToolbar.activate(esri.toolbars.Navigation.ZOOM_IN);"
>확대</div>
<div dojoType="dijit.form.Button" id="zoomout"
iconClass="zoomoutIcon"
onClick="navToolbar.activate(esri.toolbars.Navigation.ZOOM_OUT)"
/>축소</div>
<div dojoType="dijit.form.Button" id="zoomfullent"
iconClass="zoomfullentIcon"
onClick="navToolbar.zoomToFullExtent();">전체</div></div>
//지도 디스플레이 부분
<div id="map" style="width:1024px; height:512px; border:1px solid
#000;"></div>
    
```

그림 9. JavaScript API를 사용한 기본 컨트롤 구현



```
//지도 생성 부분
<esri:Map x:Name="MyMap"
ExtentChanged="MyMap_ExtentChanged" IsLogoVisible="False">
<esri:ArcGISTiledMapServiceLayer
ID="StreetMapLayer"
Url="http://115.84.164.196/ArcGIS/rest/services/SeoulLifeAPI/MapService
r"
Token="QTAKxZxWwNMlutz2EfoK239WVmS0eyGOg0-CX8e1tsNchK
NPnBLl1KJZApW14gmRVXQBROVHL4h48BUT1xzNFEg.." />
</esri:Map>
-----중략-----
<esri:ToolBar x:Name="MyToolBar" MaxItemHeight="80"
MaxItemWidth="80" VerticalAlignment="Top"
HorizontalAlignment="Center" Loaded="MyToolBar_Loaded"
ToolBarItemClicked="MyToolBar_ToolBarItemClicked"
ToolBarIndexChanged="MyToolBar_ToolBarIndexChanged"
Width="500" Height="80">

//지도 툴바 버튼 생성 부분
<esri:ToolBar.Items> <esri:ToolBarItemCollection>
<esri:ToolBarItem Text="Zoom In">
<esri:ToolBarItem.Content> </esri:ToolBarItem.Content>
</esri:ToolBarItem> <esri:ToolBarItem Text="Zoom Out">
<esri:ToolBarItem.Content> </esri:ToolBarItem.Content>
</esri:ToolBarItem>
<esri:ToolBarItem Text="Pan">
<esri:ToolBarItem.Content>
</esri:ToolBarItem.Content>
</esri:ToolBarItem>
<esri:ToolBarItem Text="Previous Extent" >
<esri:ToolBarItem.Content>
</esri:ToolBarItem.Content>
```

그림 10. Silverlight API를 사용한 기본 컨트롤 구현



```
<?xml version="1.0" encoding="utf-8"?>
<mx:Application
xmlns:mx="http://www.adobe.com/2006/mxml"
styleName="plain"
>
<mx:Script>
<![CDATA[//지도 툴바 메소드 호출 부분
private function
itemClickHandler(event:ItemClickEvent):void
{ -----중략----- }
]]>
</mx:Script>
<esri:Navigation id="navToolBar" map="{myMap}"/>
<!--지도 컨트롤 버튼 생성 부분 -->
<mx:ControlBar paddingTop="0" paddingBottom="0"
width="100%" horizontalAlign="center">
<mx:ToggleButton id="tbb"
creationComplete="tbb.selectedIndex == -1"
valueCommit="if (tbb.selectedIndex == -1)
{ navToolBar.deactivate() }"
itemClick="itemClickHandler(event)"
toggleOnClick="true">
<mx:dataProvider>
<mx:Array>
<mx:String>확대</mx:String>
<mx:String>축소</mx:String>
<mx:String>이동</mx:String>
</mx:Array>
</mx:dataProvider>
</mx:ToggleButton>
<mx:ControlBar horizontalGap="0" paddingBottom="0"
paddingTop="0">
</mx:ControlBar>
<mx:Button label="전체" click="navToolBar.zoomToFullExtent()"/>
</mx:ControlBar>
<!--지도 생성 -->
<esri:Map id="myMap" logoVisible="false" >
<esri:ArcGISTiledMapServiceLayer url="{MAP주소}"/>
</esri:Map>
<mx:TraceTarget level="{LogLevel.DEBUG}"
includeCategory="true" includeLevel="true" includeTime="true"/>
</mx:Application>
```

그림 11. Flex API를 사용한 기본 컨트롤 구현

4. 연구결과

로딩시간 분석을 위해서 Internet Explorer와 Mozilla Firefox는 영국 Simtec사의 HttpWatch Basic 7 분석 툴을 사용하였고, Google Chrome은 웹 브라우저에서 자체적으로 제공하는 개발자도구인 Network 분석툴을 사용하였다[12].

4.1 Internet Explorer

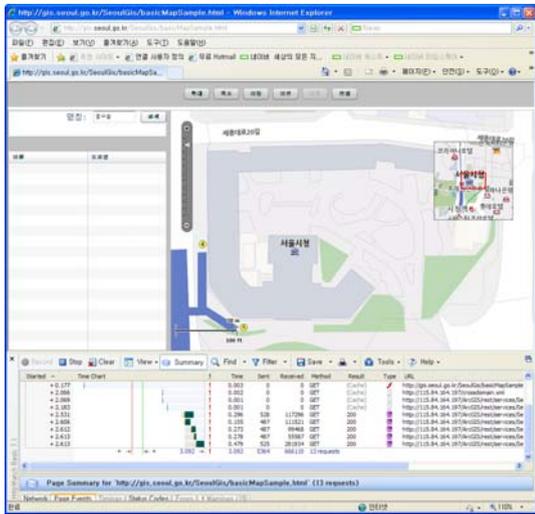


그림 12. HttpWatch Basic툴을 사용한 Internet Explorer에서의 Flex지도 분석 예

표 6. Internet Explorer에서의 분석결과

(단위:초)

기능 \ 개발언어	JavaScript	Silverlight	Flex
Tiled Map	0.515	0.941	1.918
Zoom In	0.137	0.130	0.017
Zoom Out	0.103	0.208	0.011
Full Extent	0.029	0.024	0.004
Pan	0.097	0.054	0.061
네비게이션 스타일	0.376	1.058	2.648
지도상에 정보 창보기	0.388	2.626	2.102
지도상에 해당위치에 마우스 오버시 정보표시	0.428	1.535	1.993
검색창 기능	1.878	1.362	2.456

4.2 Mozilla Firefox

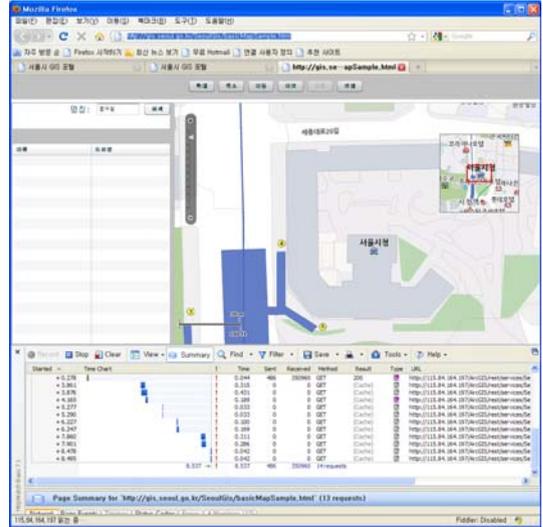


그림 13. HttpWatch Basic툴을 사용한 Firefox에서의 Flex지도 분석 예

표 7. Firefox에서의 분석결과

(단위:초)

기능 \ 개발언어	JavaScript	Silverlight	Flex
Tiled Map	1.137	1.349	2.313
Zoom In	0.174	0.188	0.121
Zoom Out	0.117	0.201	0.087
Full Extent	0.138	0.063	0.043
Pan	0.654	0.552	0.072
네비게이션 스타일	0.467	1.363	2.127
지도상에 정보 창보기	0.704	1.261	2.532
지도상에 해당위치에 마우스 오버시 정보표시	0.865	1.754	2.080
검색창 기능	1.958	1.657	2.405

- [4] VMware, 2009, "ESRI ArcGIS Server 9.3 for VMware Infrastructure," pp. 6.
- [5] Wang Linlin, Hu Dehua, 2009, "Research and Realization of RIA WebGIS Based on Flex," 2009 Intelligent Systems and Applications International Workshop. pp. 1-4.
- [6] 김문기 외 2인, 2010, "서울시 지도 Open API를 활용한 RIA기술 적용에 대한 연구," 2010(사)한국지리정보학회 추계학술대회 논문집, pp. 186-191.
- [7] 김혜선, 윤정식, 2010, "RIA 기반 웹사이트의 사용자 경험 분석," 한국인포디자인학회 인포디자인이슈 20호, pp. 53-62.
- [8] 박찬, 유관희, 2008, "차세대 웹을 위한 RIA(Rich Internet Application) 기술의 현황 및 전망," 한국콘텐츠학회 제6권 제4호, pp. 10-14.
- [9] 오윤석, 엄우학, 김병국, 2003, "서해연안 환경정보 관리를 위한 Web GIS 구축에 관한 연구," 한국GIS학회 2003년 춘계학술대회, pp. 419-424.
- [10] 우제윤 외 3인, 2001, "Web GIS를 이용한 건설공사 정보관리 시스템 구축," 한국공간정보시스템학회지, 제3권, 제2호, pp. 45-51.
- [11] 이재일, 김지호, 오병우, 2009, "모바일 환경을 위한 RIA기반 공간 데이터 서비스," 한국공간정보시스템학회지, 제11권, 제4호, pp. 56-64.
- [12] 로딩속도 분석에 사용된 프로그램
<http://www.httpwatch.com/>
- [13] 서울시GIS포털시스템, gis.seoul.go.kr
- [14] 위키백과, 리치인터넷 애플리케이션,
<http://ko.wikipedia.org/wiki>
- [15] 자바스크립트 기초강좌
<http://cafe.naver.com/home051/40>
- [16] 프로그래머블웹,
<http://www.programmableweb.com>
- [17] 플렉스 소개, <http://gstarclub.springnote.com/pages/938124>
- [18] HTML5 가 만드는 RIA 개발 로드맵,
<http://blog.naver.com/marketview/110104551174>
- [19] RIA 시장 활성화 동인 키워드와 RIA 제품 개발 방향, <http://blog.naver.com/marketview?Redirect=Log&logNo=110083128261>
- [20] Silverlight 아키텍처,
<http://msdn.microsoft.com/ko-kr/li>

[brary/bb404713\(v=vs.95\).aspx](http://library/bb404713(v=vs.95).aspx)

- [21] Web2.0시대 RIA와 UXdesign,
<http://blog.naver.com/lyjjessica/40114682953>

논문접수 : 2011.09.08

수정일 : 1차 2011.11.21 / 2차 2011.12.27

심사완료 : 2012.01.13



김 문 기

2005년 인하대학교 대학원 지리정보공학
학공학석사

2007년~2009년 한국건설기술연구원
연구원

2009년~현재 서울시청 정보화기획단
지리정보담당관 주무관(서울시GIS포털시스템, 공간데
이터웨어하우스 운영)

2011년~현재 서울시립대학교 대학원 공간정보공학과
박사과정

관심분야는 WebGIS, Web Map Service



고 준 환

1990년 국토개발(도시계획) 기술사

1996년 서울시립대학교 대학원 도시공
학과 공학박사

2010년~2011년 한국공간정보학회 통
합 초대회장

2011년~현재 서울시립대학교 도시과학대학장/공간정
보공학과 교수

관심분야는 도시정보시스템, PPGIS, NSDI