

Web 2.0 의 RIA 인터페이스 개발을 위한 Flex

연세대학교 미디어시스템연구실

김 동 철 연구원

dckim@msl.yonsei.ac.kr

<http://www.t9t9.com>

contents

1. Web 2.0

- 1.1 웹의 진화와 전망 – Web2.0, OpenAPI, Mashup
- 1.2 Mobile Web 2.0 – iPhone, Blackberry, Google Android
- 1.3 RIA 주요 기술 - Ajax, Adobe Flex, MS Silverlight

2. Flex

- 1.1 Why Flex?
- 1.2 Flex Framework & Architecture
- 1.3 Flex 적용사례

3. Flex Tutorial

- 3.1 Hello flex
- 3.2 XML Data
- 3.3 Dynamic UX Design

웹의 진화와 전망 - Web 2.0

- Web 2.0 역사

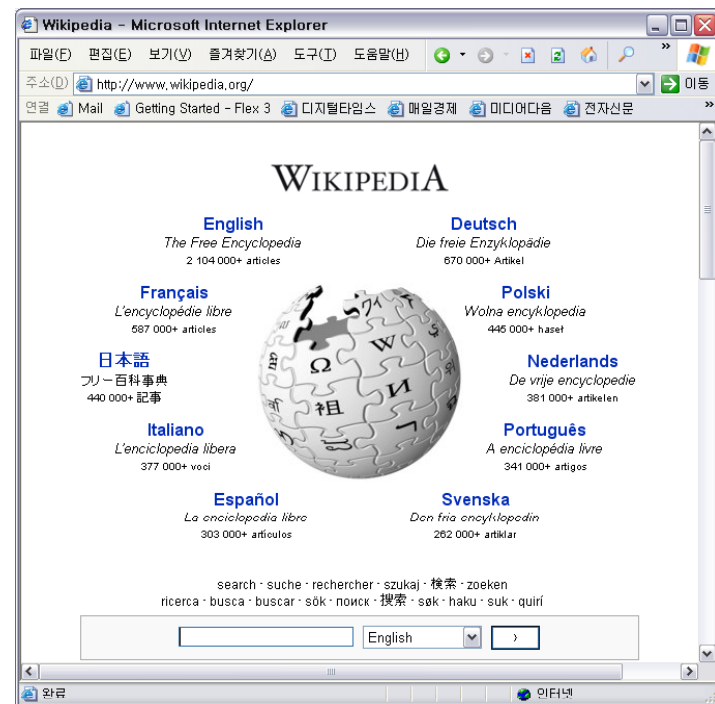
- O'Reilly사와 Media Live사가 2004년 IT Conference 를 위한 회의과정에서 Dail Dougherty 에 의해 처음 언급됨
- 닷컴 버블 과정에서 살아남은 기업들의 특징을 설명하고 차별화 하고자 Web 2.0 이란 단어를 사용

- Web 2.0 의 철학

- 참여
- 공유
- 개방

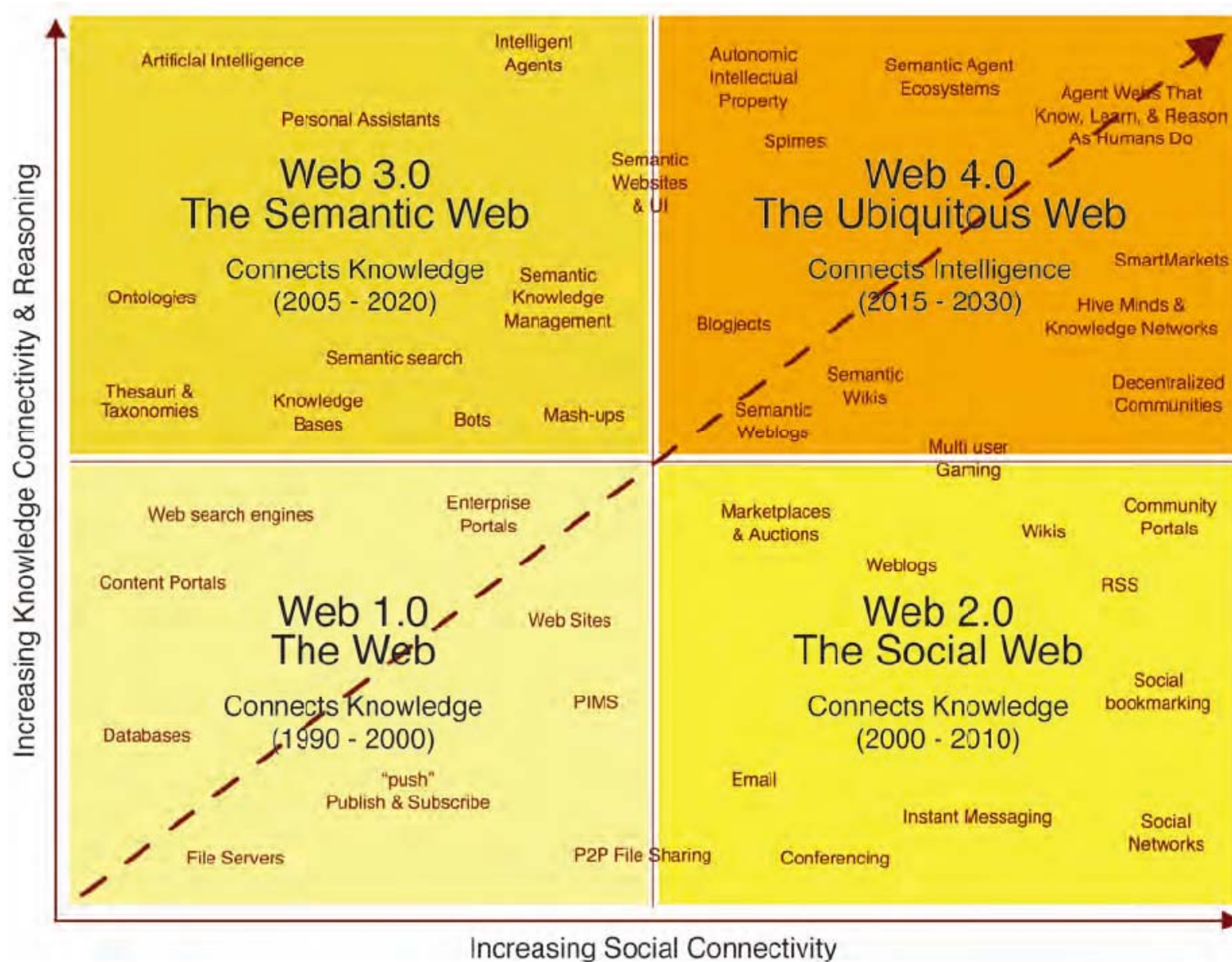
- 대표적인 사례

- Wikipedia.org
 - User 들이 **참여**하며 만들어 가는 백과사전
 - 지식을 **공유**하며
 - 누구나 수정할 수 있도록 **개방**



<http://www.wikipedia.org>

Internet Evolution



[Source: SEMTECH 2007]

http://www.semantic-conference.com/2007/semtech2007_brochure_WEB.pdf

웹의 진화와 전망 - Web 2.0

- Web 2.0 과 Web 1.0 비교

	Web 1.0	Web 2.0
특징	대표적 단어는 포탈(Portal) 포탈 위에 있는 서비스는 자신이 원하는 대로 할 수 없음	<u>플랫폼(Platform)으로서의 웹</u> 플랫폼 위에 있는 서비스는 자신이 원하는 대로 할 수 있음
	TV나 라디오처럼 정보와 서비스를 제공하기만 함 웹사이트에 자신이 올린 데이터/서비스하는 데이터 움직이거나 활용할 수 없음	누구나 데이터를 소유하지 않음 모든 사람들이 사용할 수 있음 더 나은 형태로 변경 가능함
	기술 중심	참여와 공유의 신문화 <u>사람이 중심</u>
기술	HTML, ActiveX 등	AJAX, Flax, XML, RSS, Tagging, LAMP 등
보안/ OS종속성	ActiveX 사용하며 보안취약, OS/브라우저 종속성이 있음	OS/브라우저에 상관없이 구현 가능
대표적 브라우저	인터넷 익스플로러 웹브라우저를 통해 서버에 대화요청, 단순한 뷰어역할	Fire Fox 수백 개 확장기능이 모두 일반 유저들에 의해 수정, 보완
사례	하이퍼링크 중심의 기본 웹사이트	아마존, 구글, 야후, 위키피디아

[Source: 웹 2.0 서비스 트렌드 및 비즈니스 모델, DIG커뮤니케이션]

웹의 진화와 전망 - Web 2.0

- Web 2.0 과 Web 1.0 비교

Web1.0	항목	Web 2.0
DoubleClick	광고	Google AdSense
Ofoto	사진공유	Flickr
Akamai	컨텐츠배포	BitTorrent
mp3.com	음악공유	Napster
Britannica Online	백과사전	Wikipedia
personal websites	출판	blogging
evite	스케줄	upcoming.org and EVDB
domain name speculation	마케팅	search engine optimization
page views	광고비	cost per click
screen scraping	연동	web services
publishing	컨텐츠 창작	participation
content management systems	컨텐츠 관리	wikis
directories (taxonomy)	분류	tagging ("folksonomy")
stickiness	홍보	syndication

[Source: Wikipedia, Web 2.0]
http://en.wikipedia.org/wiki/Web_2.0

웹의 진화와 전망 - Web 2.0 성공 모델

- Web 2.0 성공 모델 - M&A 사례
 - Google 이 인수한 Web 2.0 기업들

2001

[Deja](#) : usenet 사이트

[Outride, Inc.](#) : GUI 디자인에 강한 경험을 갖고 있다.

2003

[Pyra Labs](#) : "Blogger"를 만든 회사

[Neotonic Software](#) : CRM소프트웨어 기술

[Applied Semantics](#) : 텍스트 기반의 광고 기술

[Kaltix](#) : 개인화 서치 기술

[Sprinks](#) : AdWords의 기술

[Genius Labs](#) : Webblog 기술

2004

[Ignite Logic](#) : 법률관련 사이트구축 및 운영에 관한 기술

[Baidu](#)(나중에는 팔린다) : 아시아의 서치 포탈

[Picasa](#) : 이미지관련 기술

[Keyhole](#) : 지도 관련 기술, Google Earth 가 됨

[ZipDash](#) : 네비게이션 관련 기술

[Where2, LLC](#) : 지도 관련 기술

2005

[2Web Technologies](#) : 스프레드시트 관련 기술

[Urchin Software](#) : 웹 트래픽 통계관련 기술

[Dodgeball](#) : social 네트워크를 휴대용장치에서 가능한 기술

[Current Communications](#)(일부) : 초고속 광 통신망기술

[Akwan IT](#) : 남미의 IT기술관련 회사

[ReqWireless](#) : 모바일 웹브라우저와 이메일 기술.

[Android Inc.](#) : 모바일 소프트웨어 기술, Googled Android

[Time Warner/AOL](#) (일부)

2006

[dMarc Broadcasting](#) : 텍스트기반의 라디오/오디오 광고 기술

[MeasureMap](#) : 블로그 통계/트래픽 기술

[Writely](#) : 온라인 워드프로세싱 기술

[Sketchup](#) : 구글맵스에 개인의 아이콘이나 이미지 삽하는 기술

[Advanced Text Search](#) : 이스라엘 학생이 개발한 텍스트 서치 기술

[GTalkr](#) : 구글Talk를 위한 flash기반의 웹 IM client 기술

[Neven Vision](#) : 사진에서 정보를 빨리 서치하고 얻어낼수 있는 기술

[YouTube](#) : 온라인 비디오 sharing서비스

웹의 진화와 전망 - OpenAPI 와 Mashup

- OpenAPI & Mashup

- 웹 사이트가 자신들의 기능과 DB 를 이용할 수 있도록 일부 또는 전부를 공개
- 사용자는 공개된 여러 OpenAPI 를 이용 Mashup 하여 새로운 서비스를 구축 가능 (Web 포털들을 하나의 DB 또는 Function 으로 보는 관점도 생김)

- Mashup 대표 사례

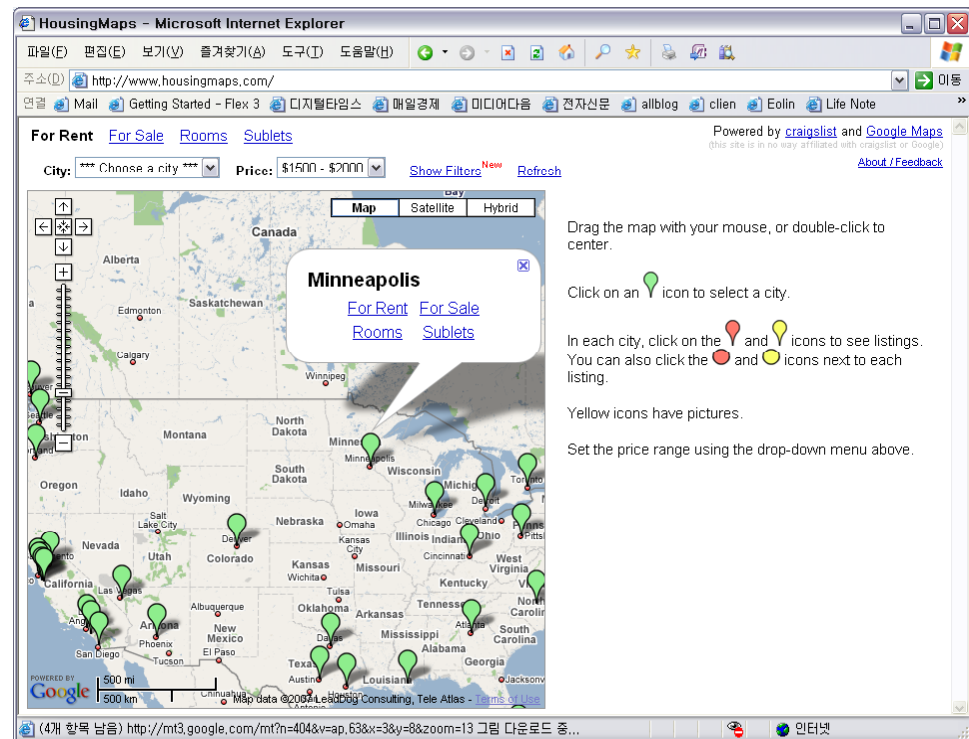
- <http://www.housingmap.com>
- 구글 맵과 부동산 정보를 결합
- 기존 부동산 중개업자들을 위협

- 국내 매시업 경진대회

- <http://mashupkorea.com>
- 매년 네이버, 다음 주최
- OpenAPI 를 활성화 중
- 입사 특전 부여

- 국내외 공개 OpenAPI

- <http://openapi.naver.com>
- <http://dna.daum.net>
- <http://code.google.com>



<http://www.housingmap.com>

웹의 진화와 전망 - 개발 플랫폼으로서의 웹

- Web Platform 의 등장

- 웹 2.0 서비스와 OpenAPI 의 확산으로 등장
- 웹 서비스들을 하나의 컴포넌드로 이용 가능
- 기존의 운영체제나 하드웨어에 플랫폼에 독립적
- 리치 인터넷 애플리케이션의 등장

- Web Platform 의 구성 요소

Client 기술	Server 기술	Content 기술	오픈 스탠더드
<ul style="list-style-type: none">• RIA 기술의 등장, 동적인 기능과 화려한 Web UI• Ajax, Flex, Silverlight, XUL, XAML, Widget	<ul style="list-style-type: none">• LAMP(Linux Apache Mysql Php) 와 같은 오픈 소스 소프트웨어 등장• LAMP, Ruby on Rails, OSS Framework	<ul style="list-style-type: none">• 웹 사이트들이 자사의 DB 를 공개• 웹을 하나의 DB 로 보는 관점	<ul style="list-style-type: none">• 웹 표준• XHTML, CSS, DOM, EC MASCript

[Source: 개발 플랫폼으로서의 웹의 진화와 향후 전망, 소프트뱅크]

웹의 진화와 전망 - RIA 기술 비교

구분	클라이언트/서버	웹	RIA(Flex)
UI Interface	자체 컴포넌트	HTML(jsp 등)	자체 컴포넌트
설치 프로그램	클라이언트 모듈	없음	플래시 플레이어
대용량 데이터 처리	가능	제한적	가능
오프라인 작업	가능	불가능	가능
실시간 모니터링	가능	제한적	가능
프로그램 배포	설치 프로그램	없음	자동 배포
컴포넌트 개발	제한적	템플릿화	가능
네트워크 용량	적음	높음	적음

[Source: Adobe 플렉스 2, 옥상훈]

Mobile Web 2.0

- iPhone
 - Multi Touch Interface
 - Full browsing Issue - Safari + Multi Touch
- Blackberry
 - E-mail service
 - Qwerty buttons
- Google Android
 - Open source Mobile Phone OSZ

Mac OS X

Blackberry OS

Android OS



Apple iPhone



RIM Blackberry

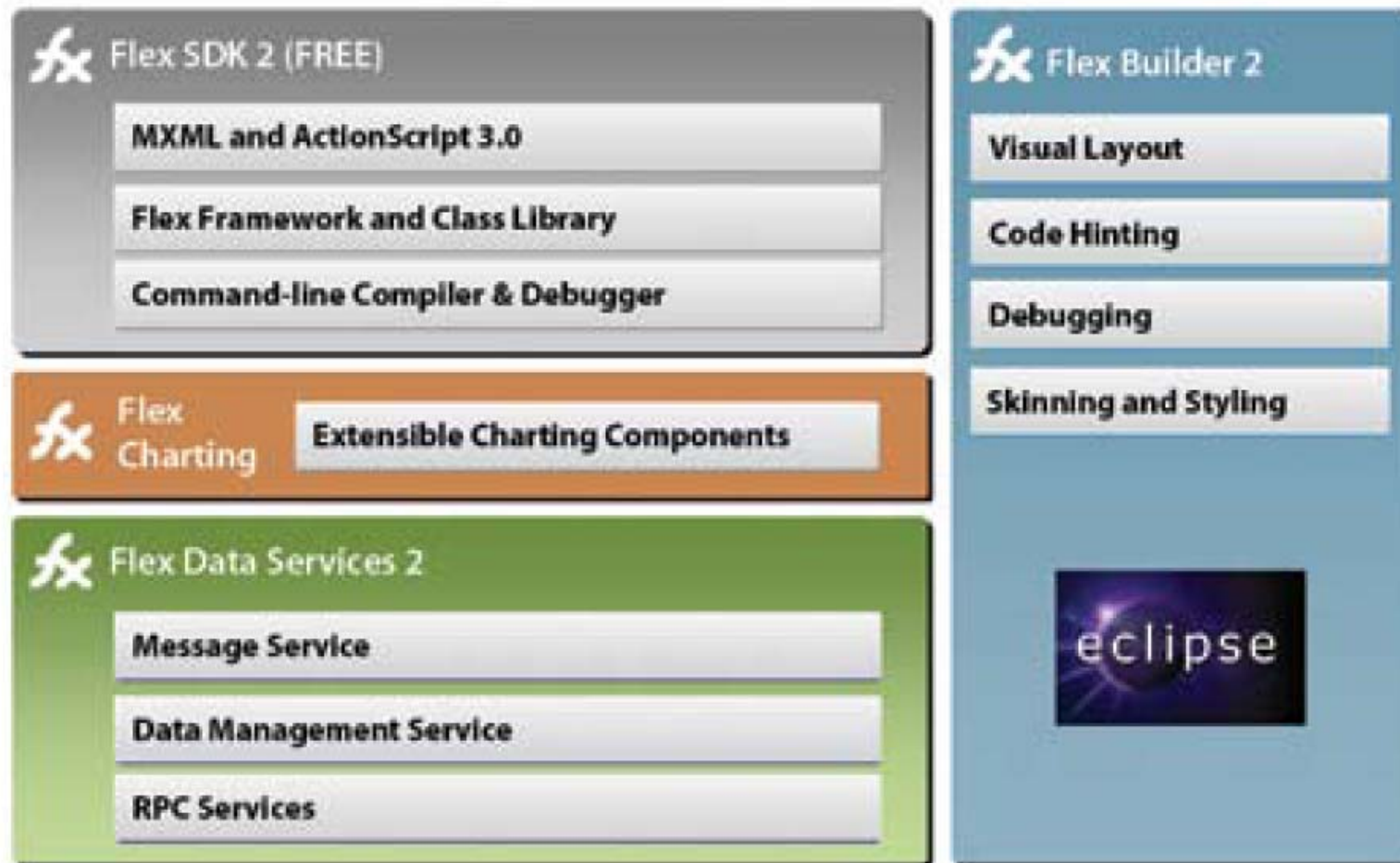


Google Android

RIA 주요기술

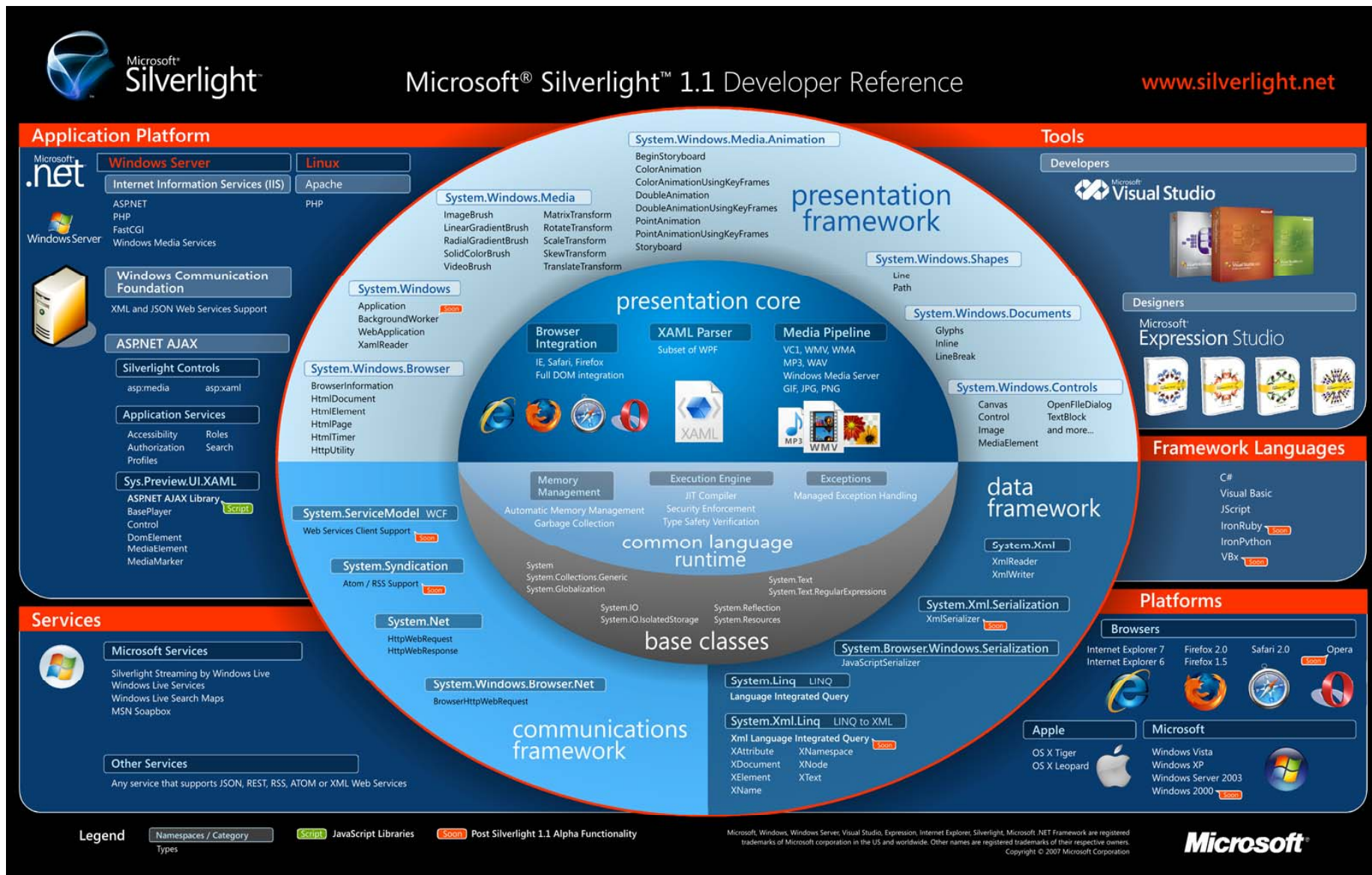
- RIA
 - Rich Internet Application 의 약자
 - 데스크톱 애플리케이션 수준의 기능, UI 를 갖는 웹
 - Rich User eXperience
- RIA 주요기술
 - Adobe Flex
 - Flash 기반, RIA 기술의 주류를 이룸
 - MXML, Actionscript,
 - AIR(Adobe Integrated Runtime)
 - 웹 애플리케이션과 데스크톱 애플리케이션을 융합
 - 다양한 디바이스에서 작동
 - Microsoft Silverlight
 - RIA 시장에 진입 시도
 - Open Laszlo
 - 오픈소스, Flash 및 Ajax 를 동시 지원
 - Ajax (Javascript)
 - 구글에서 주로 사용하면서 유명해짐
 - Google mail, Google maps

RIA 주요기술 - Flex



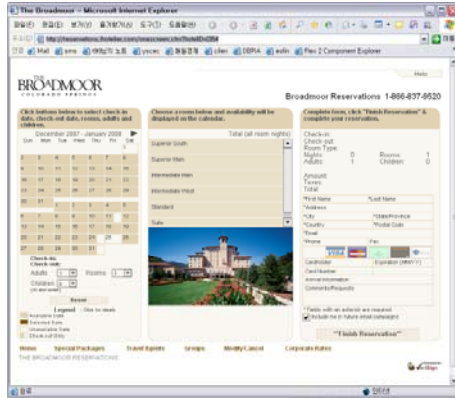
[Source: <http://flex.org>]

RIA 주요기술 - Silverlight



RIA 적용사례

1. Process 단축 UI

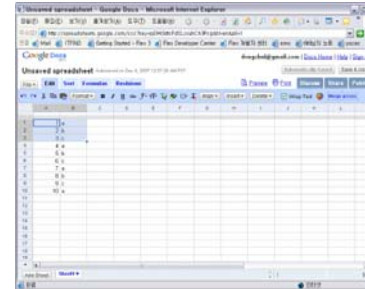


<http://reservations.ihotelier.com/onescreen.cfm?hotelID=2054> (추천)
by Flex

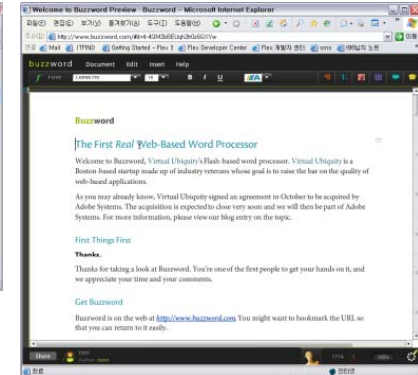


<http://www.gshop.co.kr>
by Flex

2. Web Office



<http://docs.google.com>
by Ajax



<http://www.buzzword.com>
by Flex

3. Multimedia

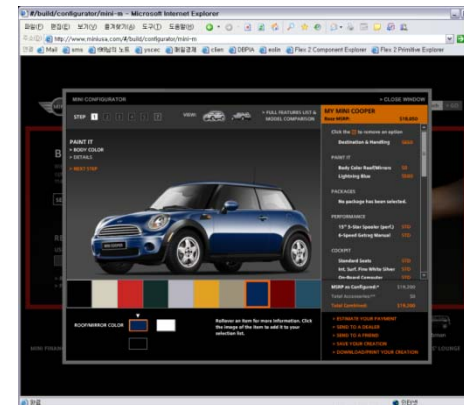


http://news.sbs.co.kr/section_news/news_read.jsp?news_id=N1000345080 (추천)
by Silverlight



<http://www.etonline.com/emmys>
by Silverlight

4. Interaction



<http://www.miniusa.com/#/build/configurator/mini-m>
by Flex

contents

1. Web 2.0

- 1.1 웹의 진화와 전망 – Web2.0, OpenAPI, Mashup
- 1.2 Mobile Web 2.0 – iPhone, Blackberry, Google Android
- 1.3 RIA 주요기술 - Ajax, Adobe Flex, MS Silverlight

2. Flex

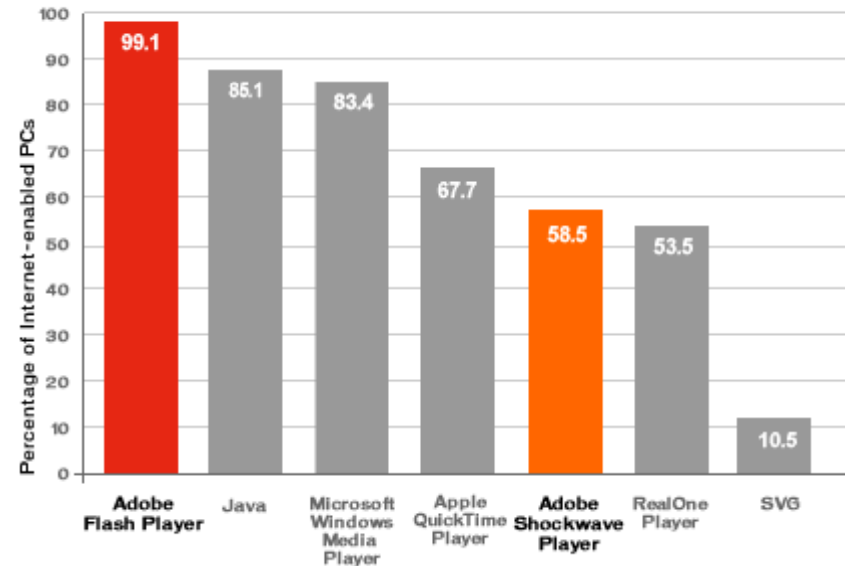
- 1.1 Why Flex?
- 1.2 Flex Framework & Architecture
- 1.3 Flex 적용사례

3. Flex Tutorial

- 3.1 Hello flex
- 3.2 XML Data
- 3.3 Dynamic UX Design

Why Flex?

- Platform Independent
 - One Source Multi Use
- 전 세계 PC 에서 작동
 - 전세계 99.1% 컴퓨터에 Flash Player 탑재
- Mobile Phone, MP3, PMP 에서 작동
 - Flash Lite 엔진 탑재
- Benefit
 - High Productivity
 - Light weight



인터넷을 사용하는 PC 들에 설치된 Flash Player
 [Source: http://www.adobe.com/products/player_census/flashplayer/]

Worldwide Ubiquity of Adobe Flash Player by Version - September 2007

	Flash Player 6	Flash Player 7	Flash Player 8	Flash Player 9
Mature Markets ¹	99.1%	99.1%	98.4%	93.3%
US/Canada	99.0%	99.0%	98.5%	94.1%
Europe ²	99.2%	99.2%	98.2%	93.7%
Japan	99.5%	99.5%	99.0%	93.7%
Emerging Markets ³	Not surveyed in this wave			

[Source: http://www.adobe.com/products/player_census/flashplayer/version_penetration.html]

Why Flex?

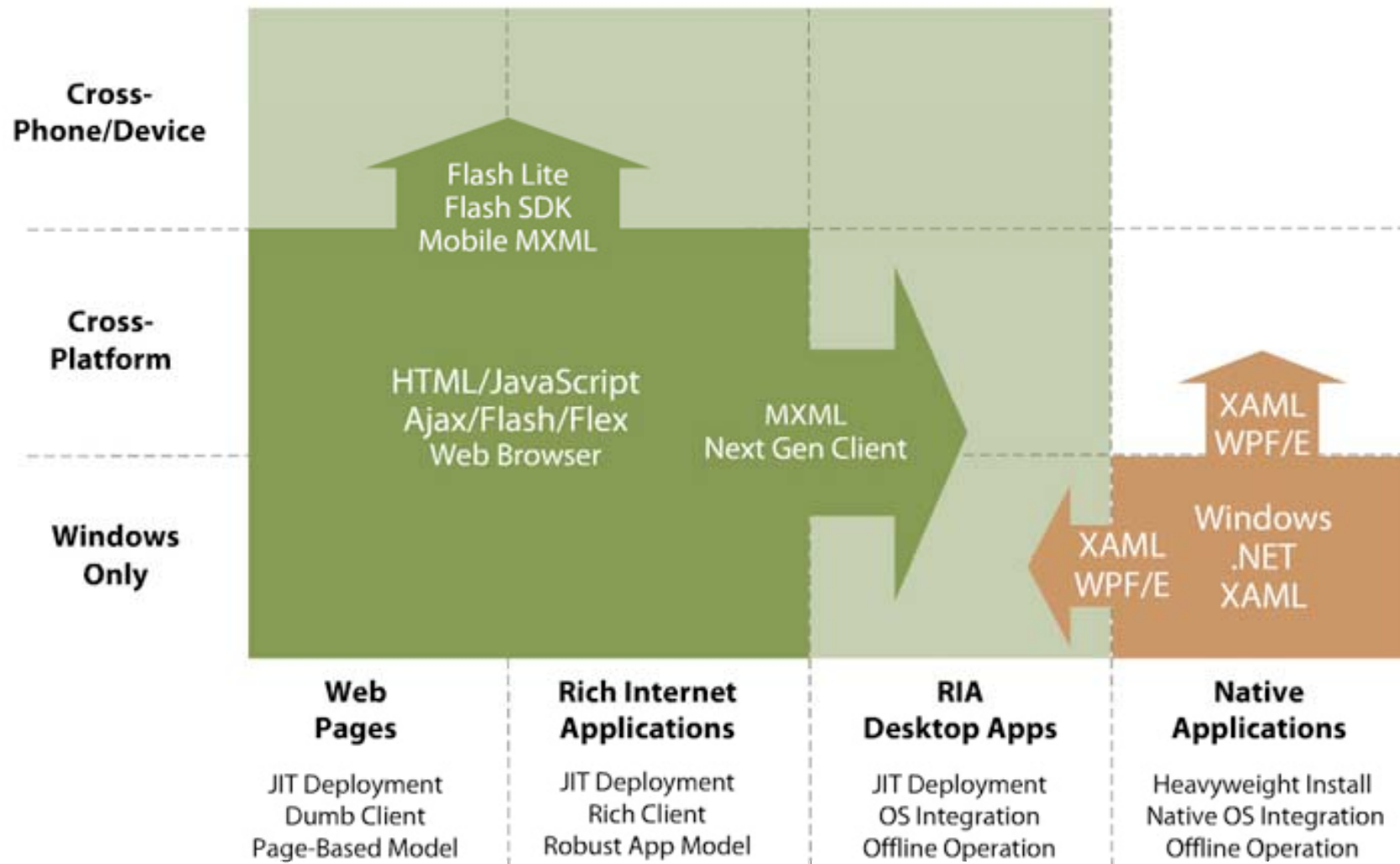
- Oracle 에서 Flex 채택, 2007/11/12
 - 최근 개최된 Oracle OpenWorld 2007
 - Enterprise Manager, Sales Campaign, BI Publisher 등등 자사의 제품들에 Flex 를 채택한 데모들을 선보임

[Source: <http://www.jamesward.org/wordpress/2007/11/12/oracle-chooses-flex-part-1/>]
- 삼성 X-Internet 표준 Flex 채택, 2006/06/30

[Source: <http://www.zdnet.co.kr/itbiz/press/enterprise/dev/0,39032022,39148958,00.htm>]
- 인터넷 뱅킹 Flex 도입 사례
 - 송수신 량이 평균 62.3% 정도 감소하여 화제
 - 월말에 폭주하는 사용자 수를 수용하기 위해 하드웨어 투자비용이 필요했었음
 - 도입 후 획기적으로 절감
- 서비스 제공업체의 연간 서버 투자비용
 - 기존 1년 1대당 4000불(400만원)
 - RIA 기술 도입후 2010년에는 200불(20만원) 정도로 낮아질 것

[Source: Next Internet, David & Danny]

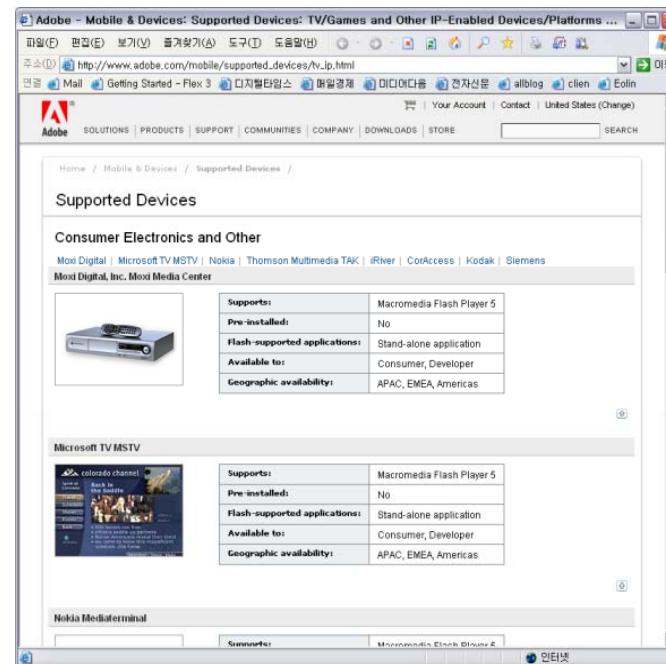
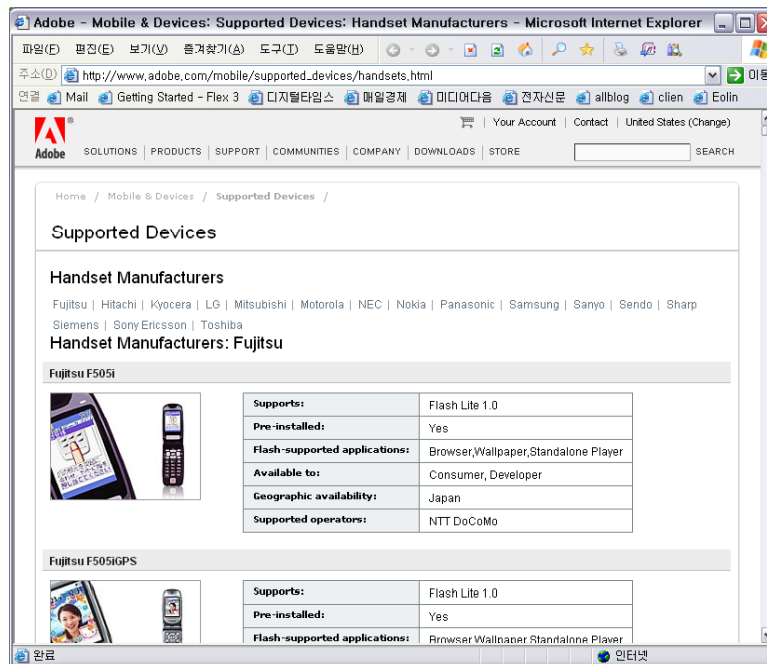
Why Flex?



[Source: An introduction to Flex and Apollo, Duane Nickull, Adobe]

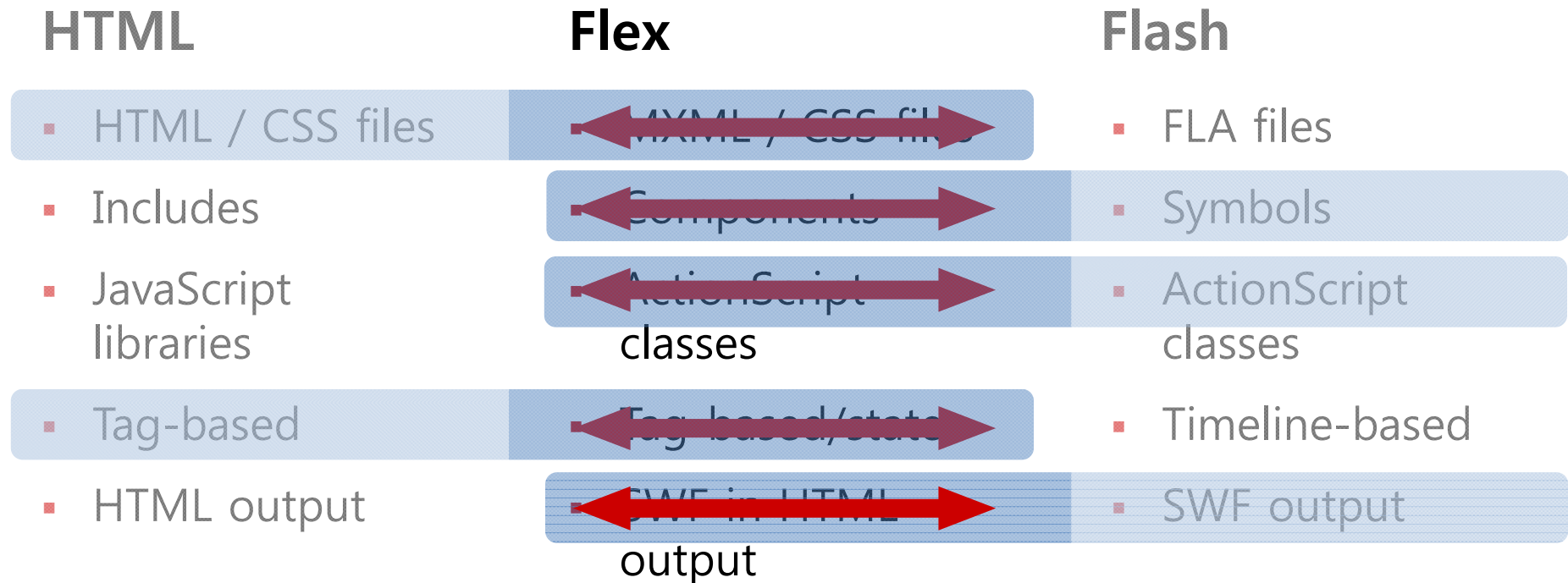
Why Flex? – Flash enable devices

- Handset Manufacturers
 - Fujitsu, Hitachi, Kyocera, LG, Mitsubishi, Motorola, NEC, Nokia, Panasonic, Samsung, Sanyo, Sendo, Sharp, Siemens, Sony Ericsson, Toshiba
- Operators
 - KDDI, NTT DoCoMo, Verizon Wireless, SoftBank (Vodafone KK)
- PDAs
 - Microsoft Pocket PC, Sony CLIE
- Electronics and Other
 - Moxi Digital, Microsoft TV MSTV, Nokia, Thomson Multimedia, TAK, iRiver, CorAccess, Kodak, Siemens



[Source: http://www.adobe.com/mobile/supported_devices/handsets.html]

HTML vs Flex vs Flash



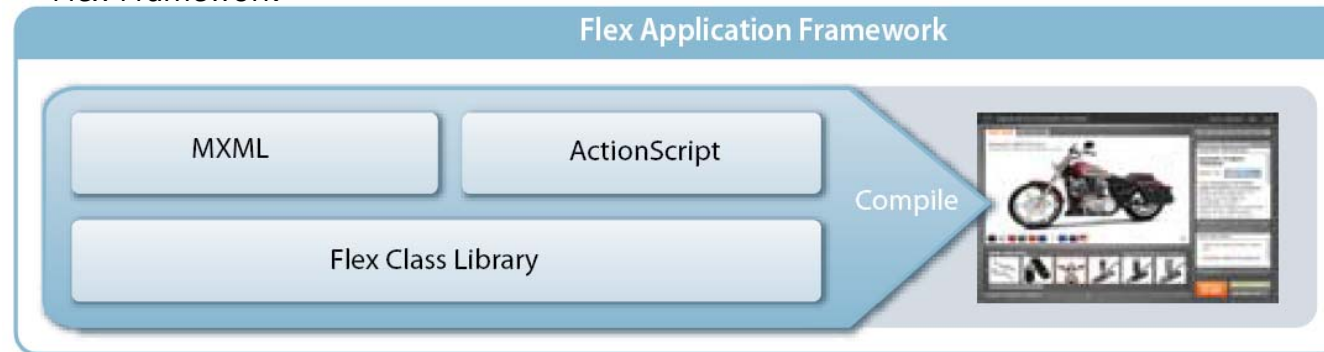
[Source: An introduction to Flex and Apollo, Duane Nickull, Adobe]

- Unlike HTML, Flex was designed for application development (no more creating navibars by slicing up images!)

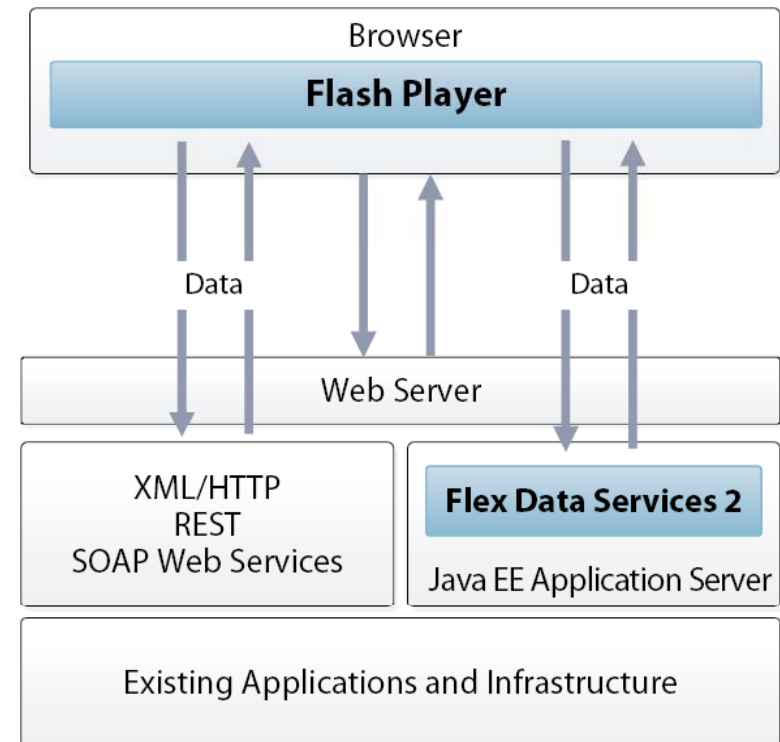
Flex Framework & Architecture



Flex Framework



Flex Architecture



Flex Framework - MXML

- Macromedia eXtensible Markup language.
 - XML language
 - Declarative programming (objects / properties are declared)

```
<?xml version="1.0" encoding="utf-8"?>
<mx:Application xmlns:mx="http://www.adobe.com/2006/mxml" xmlns="" layout="vertical">

    <mx:WebService id="webService"
        wsdl="http://localhost:8700/f2ria/services/SimpleWebService?wsdl"
        load="webService.outMethod()"
        showBusyCursor="true"/>

    <mx:Label text="{webService.outMethod.result}"/>

    <mx:Button label="Get Static Data"
        click="webService.arrayOutMethod()"/>

    <mx:DataGrid dataProvider="{webService.arrayOutMethod.result}"/>

    <mx:Button label="Get Dynamic Data"
        click="webService.getSandwichList()"/>

    <mx:DataGrid dataProvider="{webService.getSandwichList.result}"/>

</mx:Application>
```

Flex Framework - ActionScript


- AS : Action Script
 - Action Script 3.0

```
<mx:Script>
  <![CDATA[
    import mx.controls.Alert;
    import mx.managers.CursorManager;
    import mx.collections.ArrayCollection;
    import mx.utils.StringUtil;

    private var fr:FileReferenceList;
    private var nTotSize:int = 0;
    private var nCurStep:int = 0;
    [Bindable]
    private var arrColl:ArrayCollection;

    private function init():void {
        arrColl = new ArrayCollection();
        ExternalInterface.addCallback("myFlexFunction",Upload_SEQ);
    }
  ]]>
</mx:Script>
```



Flex Framework - Class Library



PACKAGES
mx.core
mx.effects
mx.formatters
mx.validators
mx.managers

mx.core.ext
mx.managers.layoutClasses

mx.events
mx.utils
mx.styles
mx.servicetags
mx.preloaders

Flex Class Library Reference

mx.core

- Application**
The class that manages the application's lifecycle and provides a central location for application-wide data and services.
- DragSource**
The class that manages the drag-and-drop operation.
- ExternalContent**
The class that manages external content.
- MXMLElement**
The class that represents an XML element.
- MXMLElementAdapter**
The class that adapts an MXMLElement to a data provider.
- Repeater**
The class that manages a list of data items.
- ScrollView**
The class that manages a scrollable view.
- UIComponent**
The class that represents a user interface component.
- UIObject**
The class that represents a user interface object.
- UIObjectDescriptor**
The class that describes a UIObject.
- View**
The class that represents a view.

mx.core.ext

- FlexExtensions**
The class that provides extensions to the Flex framework.
- UIComponentExtensions**
The class that provides extensions to the UIComponent class.

mx.effects

- Sequence**
The class that manages a sequence of effects.
- CompositeEffect**
The class that manages a composite effect.
- Effect**
The class that represents an effect.
- EffectManager**
The class that manages a collection of effects.
- Fade**
The class that represents a fade effect.
- MaskEffect**
The class that represents a mask effect.
- Move**
The class that represents a move effect.
- Parallel**
The class that manages a parallel sequence of effects.
- Pause**
The class that represents a pause effect.
- Resize**
The class that represents a resize effect.

mx.events

- EventDispatcher**
The class that manages event dispatching.
- LowLevelEvents**
The class that manages low-level events.
- UIEventDispatcher**
The class that manages UI events.

mx.utils

- ArrayUtil**
The class that provides utility methods for arrays.
- ClassUtil**
The class that provides utility methods for classes.
- Delegate**
The class that provides a delegate pattern.

mx.styles

- ObjectCopy**
The class that provides utility methods for object copying.
- URLUtil**
The class that provides utility methods for URLs.
- XMLUtil**
The class that provides utility methods for XML.

mx.formatters

- CurrencyFormatter**
The class that formats currency values.
- DateBase**
The class that provides a base class for date formatters.
- DateFormatter**
The class that formats date values.
- Formatter**
The class that provides a base class for formatters.
- NumberBase**
The class that provides a base class for number formatters.
- NumberFormatter**
The class that formats number values.
- PhoneFormatter**
The class that formats phone numbers.
- StringFormatter**
The class that formats string values.
- SwitchSymbolFormatter**
The class that formats switch symbols.
- ZipCodeFormatter**
The class that formats zip codes.

mx.validators

- CreditCardValidator**
The class that validates credit card numbers.
- DateValidator**
The class that validates date values.
- EmailValidator**
The class that validates email addresses.
- NumberValidator**
The class that validates number values.
- PhoneNumberValidator**
The class that validates phone numbers.
- SocialSecurityValidator**
The class that validates social security numbers.
- StringValidator**
The class that validates string values.
- ValidationFailedEvent**
The class that represents a validation failure event.
- Validator**
The class that provides a base class for validators.
- ZipCodeValidator**
The class that validates zip codes.

mx.managers


- CursorManager**
The class that manages the cursor.
- DepthManager**
The class that manages the depth of UI components.
- DragManager**
The class that manages drag-and-drop operations.
- FocusManager**
The class that manages focus.
- HistoryManager**
The class that manages the application history.
- LayoutManager**
The class that manages the layout of UI components.
- OverlappedWindows**
The class that manages overlapped windows.
- PopUpManager**
The class that manages pop-up windows.
- SystemManager**
The class that manages system resources.
- ToolTipManager**
The class that manages tool tips.

mx.managers.layoutClasses

- PriorityQueue**
The class that provides a priority queue.

mx.preloaders

- DownloadProgressBar**
The class that provides a download progress bar.



Downloaded from www.adobe.com

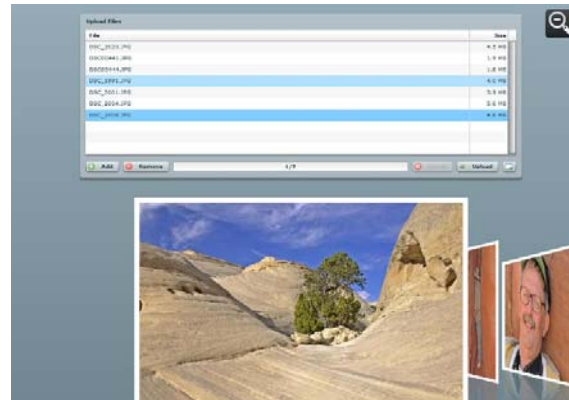
created by wencle | contact: edae@mg.de | www.mg.de | download actual version from: www.mg.de/flex | version 1.0

25

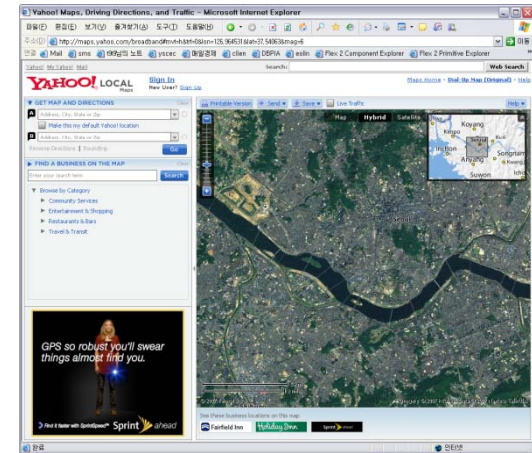
Flex 적용사례



<http://flex.org/showcase> (추천)



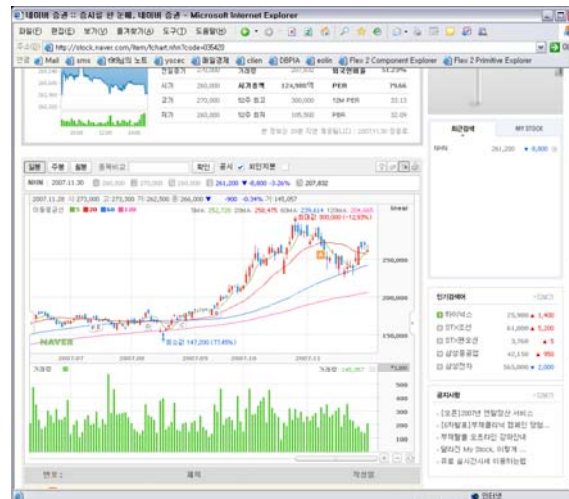
<http://www.friendlytek.com/upload/>



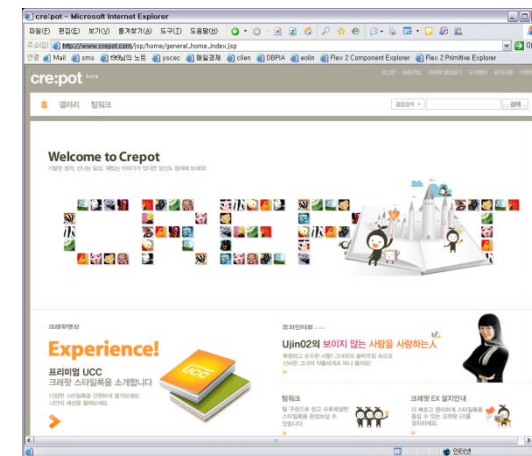
<http://map.yahoo.com>



CGV – 빠른 예매 시스템
<http://www.cgv.co.kr>



네이버 증권 – <http://stock.naver.com>



<http://www.crepot.com> (추천)

contents

1. Web 2.0

- 1.1 웹의 진화와 전망 – Web2.0, OpenAPI, Mashup
- 1.2 Mobile Web 2.0 – iPhone, Blackberry, Google Android
- 1.3 RIA 주요기술 - Ajax, Adobe Flex, MS Silverlight

2. Flex

- 1.1 Why Flex?
- 1.2 Flex Framework & Architecture
- 1.3 Flex 적용사례

3. Flex Tutorial

- 3.1 Hello flex
- 3.2 XML Data
- 3.3 Dynamic UX Design

Flex Tutorial

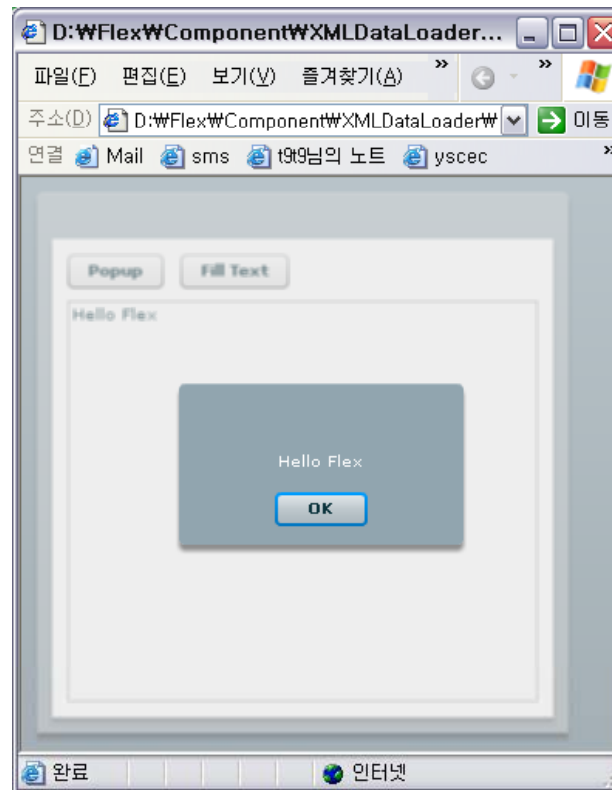
- Hello Flex
 - 기본적인 Flex 컴파일 과정을 살펴본다.
- XML Data Loader
 - 데이터 처리의 기본인 XML 데이터 처리를 배운다.
- Dynamic UX Design
 - RIA의 핵심이라고 할 수 있는 Dynamic 하게 인터페이스가 변화하는 것을 살펴 본다.

Flex Tutorial - Hello Flex

HelloFlex.mxml

```
<?xml version="1.0" encoding="utf-8"?>
<mx:Application xmlns:mx="http://www.adobe.com/2006/mxml" layout="absolute">
    <mx:Panel x="10" y="10" width="250" height="350" layout="absolute">
        <mx:Button x="10" y="10" label="Popup" click="mx.controls.Alert.show('Hello
Flex')"/>

        <mx:Button x="83" y="10" label="Fill Text" click="TextBox.text='Hello Flex'"/>
        <mx:TextArea x="10" y="40" width="210" height="260" id="TextBox"/>
    </mx:Panel>
</mx:Application>
```



<http://flex.t9t9.com/HelloFlex.html>

Flex Tutorial - XML Data Loader

XMLDataLoader.mxml

```
<?xml version="1.0" encoding="utf-8"?>
<mx:Application xmlns:mx="http://www.adobe.com/2006/mxml" layout="vertical">
    <mx:Script>
        <![CDATA[
            import mx.collections.ArrayCollection;
            import mx.utils.ArrayUtil;

        ]]>
    </mx:Script>

    <!-- xml 을 Model 객체로 읽어 들이기 -->
    <mx:Model id="xmlData" source="data.xml" />

    <!-- Model 객체의 데이터를 ArrayCollection을 이용하여
        ArrayCollection의 source값으로 읽어 들임 -->
    <mx:ArrayCollection id="xmlAC" source="{ArrayUtil.toArray(xmlData.item)}" />

    <mx:Panel title="XML Data" width="100%">
        <mx:List dataProvider="{xmlAC}" width="100%" editable="false" fontSize="11"/>
        <mx:ControlBar />
    </mx:Panel>

    <mx:Panel title="DataGrid" width="100%" height="242" fontSize="11" layout="vertical">
        <mx>DataGrid dataProvider="{xmlAC}" editable="false" width="100%"
borderStyle="solid" height="100%"/>
        <mx:ControlBar/>
    </mx:Panel>

</mx:Application>
```

Flex Tutorial - XML Data Loader

data.xml

```
<?xml version="1.0" encoding="utf-8" ?>
<items>
  <item label="맑음" date="2007-07-08" />
  <item label="비온뒤 맑음" date="2007-07-09" />
  <item label="흐림" date="2007-07-10" />
  <item label="화창" date="2007-07-11" />
</items>
```



<http://flex.t9t9.com/XMLData.html>

Flex Tutorial - Dynamic UX Design (1/2)

DynamicUX.mxml

```
<?xml version="1.0" ?>
<!-- transitions/DefiningTrans.mxml -->
<mx:Application xmlns:mx="http://www.adobe.com/2006/mxml" >
  <!-- http://flexdocs.kr/docs/flex2/docs/00000938.html -->
  <!-- Define the two view states, in addition to the base state.-->
  <mx:states>
    <mx:State name="One">
      <mx:SetProperty target="{p1}" name="x" value="110"/>
      <mx:SetProperty target="{p1}" name="y" value="0"/>
      <mx:SetProperty target="{p1}" name="width" value="200"/>
      <mx:SetProperty target="{p1}" name="height" value="210"/>
      <mx:SetProperty target="{p2}" name="x" value="0"/>
      <mx:SetProperty target="{p2}" name="y" value="0"/>
      <mx:SetProperty target="{p2}" name="width" value="100"/>
      <mx:SetProperty target="{p2}" name="height" value="100"/>
      <mx:SetProperty target="{p3}" name="x" value="0"/>
      <mx:SetProperty target="{p3}" name="y" value="110"/>
      <mx:SetProperty target="{p3}" name="width" value="100"/>
      <mx:SetProperty target="{p3}" name="height" value="100"/>
    </mx:State>
    <mx:State name="Two">
      <mx:SetProperty target="{p2}" name="x" value="110"/>
      <mx:SetProperty target="{p2}" name="y" value="0"/>
      <mx:SetProperty target="{p2}" name="width" value="200"/>
      <mx:SetProperty target="{p2}" name="height" value="210"/>
      <mx:SetProperty target="{p3}" name="x" value="0"/>
      <mx:SetProperty target="{p3}" name="y" value="110"/>
      <mx:SetProperty target="{p3}" name="width" value="100"/>
      <mx:SetProperty target="{p3}" name="height" value="100"/>
    </mx:State>
  </mx:states>
</mx:Application>
```

Flex Tutorial - Dynamic UX Design (2/2)

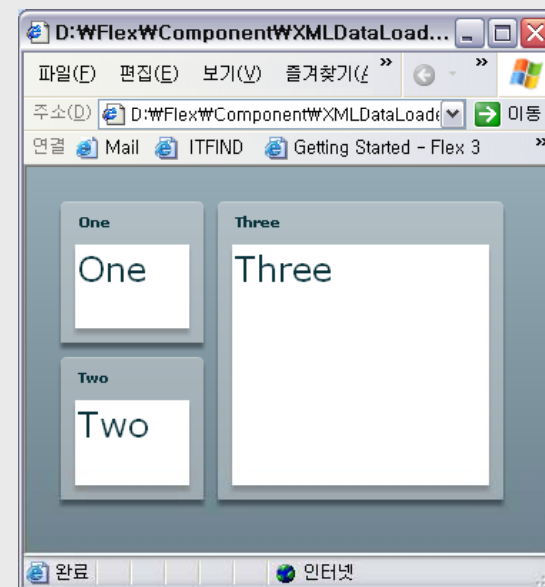
DynamicUX.mxml

```
<!-- Define Transition array with one Transition object.-->
<mx:transitions>
    <!-- Define a transition for changing from any state to any state.
    -->
    <mx:Transition id="myTransition" fromState="*" toState="*">
        <!-- Define a Parallel effect as the top-level effect.-->
        <mx:Parallel id="t1" targets="{[p1,p2,p3]}">
            <!-- Define a Move and Resize effect.-->
            <mx:Move duration="400"/>
            <mx:Resize duration="400"/>
        </mx:Parallel>
    </mx:Transition>
</mx:transitions>

<!-- Define the Canvas container holdig the three Panel containers.-->
<mx:Canvas id="pm" width="100%" height="100%" >
    <mx:Panel id="p1" title="One"
        x="0" y="0" width="100" height="100"
        click="currentState='One'" >
        <mx:Label fontSize="24" text="One"/>
    </mx:Panel>

    <mx:Panel id="p2" title="Two"
        x="0" y="110" width="100" height="100"
        click="currentState='Two'" >
        <mx:Label fontSize="24" text="Two"/>
    </mx:Panel>

    <mx:Panel id="p3" title="Three"
        x="110" y="0" width="200" height="210"
        click="currentState=''" >
        <mx:Label fontSize="24" text="Three"/>
    </mx:Panel>
</mx:Canvas>
</mx:Application>
```



<http://flex.t9t9.com/DynamicUX.html>

Reference

- Adobe Flex 개발자 센터 - 빠른 시작 가이드
 - <http://www.adobe.com/kr/devnet/flex/?tab:quickstart=1>
- 네이버 Flex 카페 - <http://cafe.naver.com/flexcomponent>
- Flex 커뮤니티 - <http://www.flex.org>
- Download Flex - <http://www.adobe.com/products/flex/>
- Free Compiler - <http://www.adobe.com/products/flex/sdk/>
- Free online compiler - <http://try.flex.org/>
- Flex Forums
 - <http://www.adobe.com/cfusion/webforums/forum/index.cfm?forumid=60>
 - <http://forums.flexonline.com/ib/ikonboard.cgi>
 - <http://groups.yahoo.com/group/flexcoders/>
 - <http://www.dopejam.com/FlexCoders/>
- Flex Component Explorer
 - <http://examples.adobe.com/flex2/inproduct/sdk/explorer/explorer.html>
- Flex Style Explorer
 - <http://examples.adobe.com/flex2/inproduct/sdk/explorer/explorer.html>