

복습

■ 왜, 웹 표준을 써야 하나?

- 다양한 OS, 웹 브라우저, 디바이스 사용자를 위해
- 노약자, 어린이, 장애인을 위해
- 에이전트 및 검색 엔진 로봇 등 기계를 위해

■ 방법론

- 구조와 표현을 완전히 분리할 것
- 구조(X/HTML), 표현(CSS)

■ Structural Markup

- HTML에서 외양(Style)에 관계된 부분은 모두 CSS로 빼 냄으로서 데이터만 남도록 하는 형태
- Ex. 과 의 다양한 사용 방법

■ Semantic Markup

- HTML에서 가급적 의미를 잘 살린 태그를 사용하여 문서의 정보를 정확히 전달하도록 한다.
- Ex.) vs. , <i> vs.

DOM 스크립팅

문서 객체 모델(DOM)

■ Document / 문서

- 웹 브라우저에서 읽을 수 있는 모든 형태의 콘텐츠
- HTML 요소, 속성, 텍스트, CSS 프로퍼티 등

■ Object / 객체

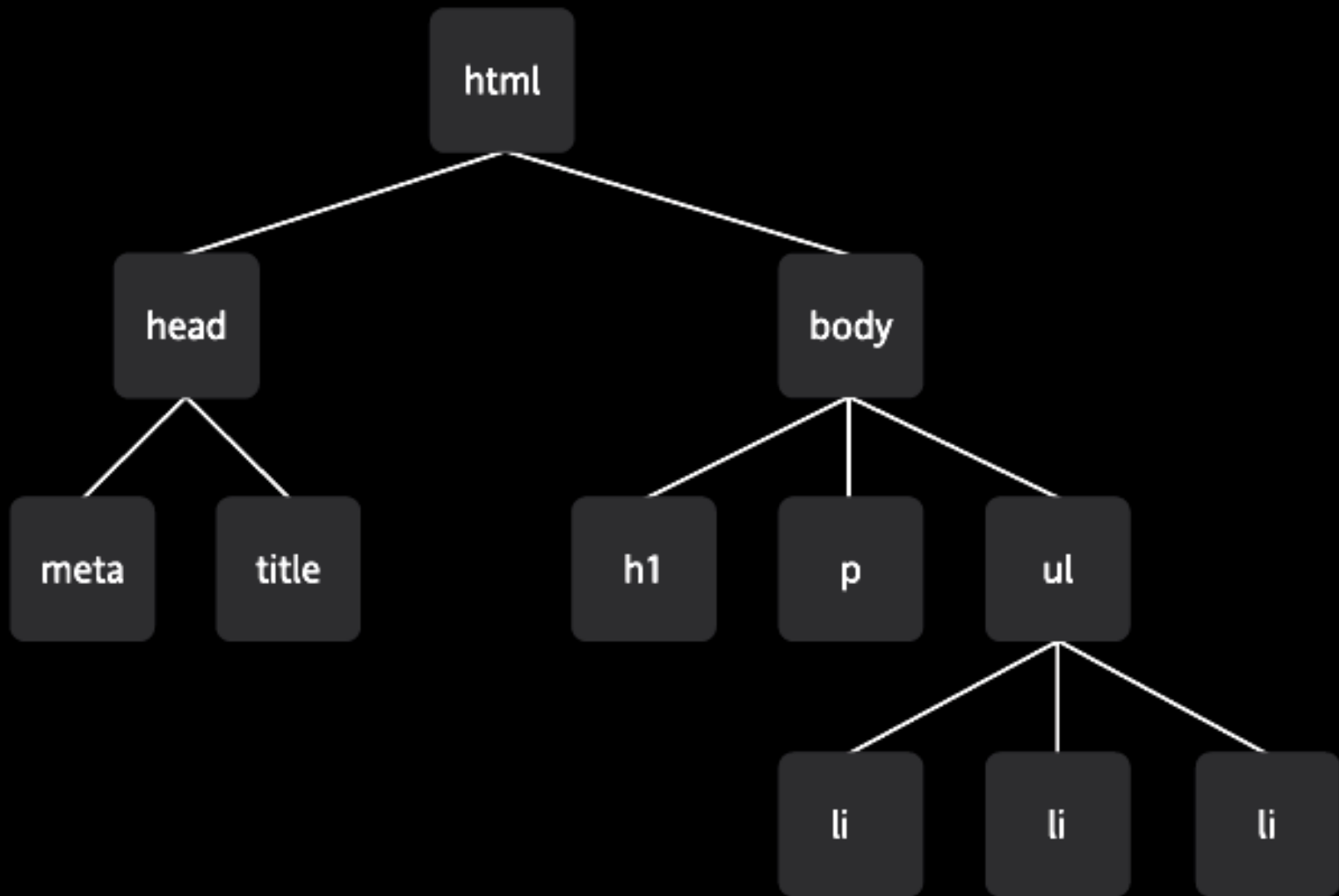
- 객체는 메소드와 프로퍼티로 구성됨
- 메소드: `document.form`
- 프로퍼티: `document.form.value`

■ Model / 모형

- 문서 객체를 표시하는 방법 (트리 구조)
- 지도나 모형과 같음. 자바 스크립트로 접근 가능

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN"
"http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en">
  <head>
    <meta http-equiv="content-type" content="text/html;
■ charset=utf-8" />
    <title>쇼핑 목록</title>
  </head>
  <body>
    <h1>쇼핑 목록</h1>
    <p title="a gentle reminder">이 물건 사는 것을 잊지 마세요.</p>
    <ul id="purchases">
      <li>땅콩</li>
      <li>치즈</li>
      <li>우유</li>
    </ul>
  </body>
</html>
```

객체 지도(Object Tree)



요소 노드

p

title =
"a gentle
reminder"

속성 노드

이 물건 사는 것을
잊지마세요!

텍스트 노드

자바스크립트

■ 자바 스크립트

- 웹 브라우저가 VM이 되는 해석기 언어
- 90년대 Jscript, VBScript 등 다양한 기술 확장
- 2000년대 ECMAScript 표준화가 됨.
- Flash에서 ActionScript와 같은 유사 언어 만들어짐

■ 자바 스크립트 사용법

- 자바 스크립트가 주요 기능이 아니라 문서를 보조해 주는 수단으로 사용해야 함.
- 자바 스크립트는 문서 객체 모델(DOM)을 통해 사용자 경험 및 기능적 요소 제공.

DOM 스크립팅

■ 원칙 1. 단계적 기능 향상/축소

- 자바 스크립트 기능을 끄더라도 웹 문서의 주요 기능은 가능해야 함

■ 원칙 2. 스크립트 분리

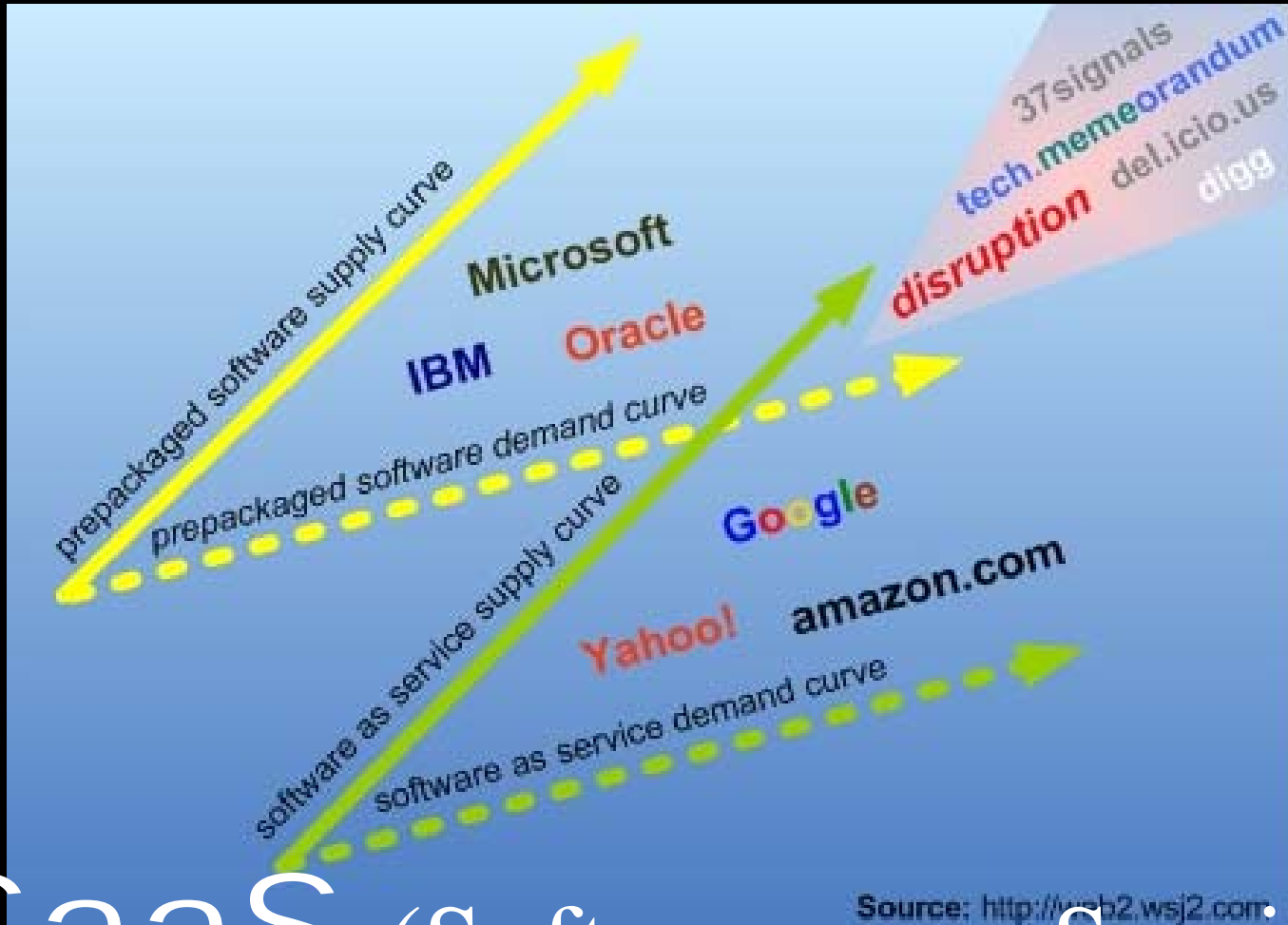
- 가급적 마크업에 스크립트를 직접 쓰지 말 것
- 별도 파일로 함수를 분리할 것

■ 원칙 3. 하위 호환성

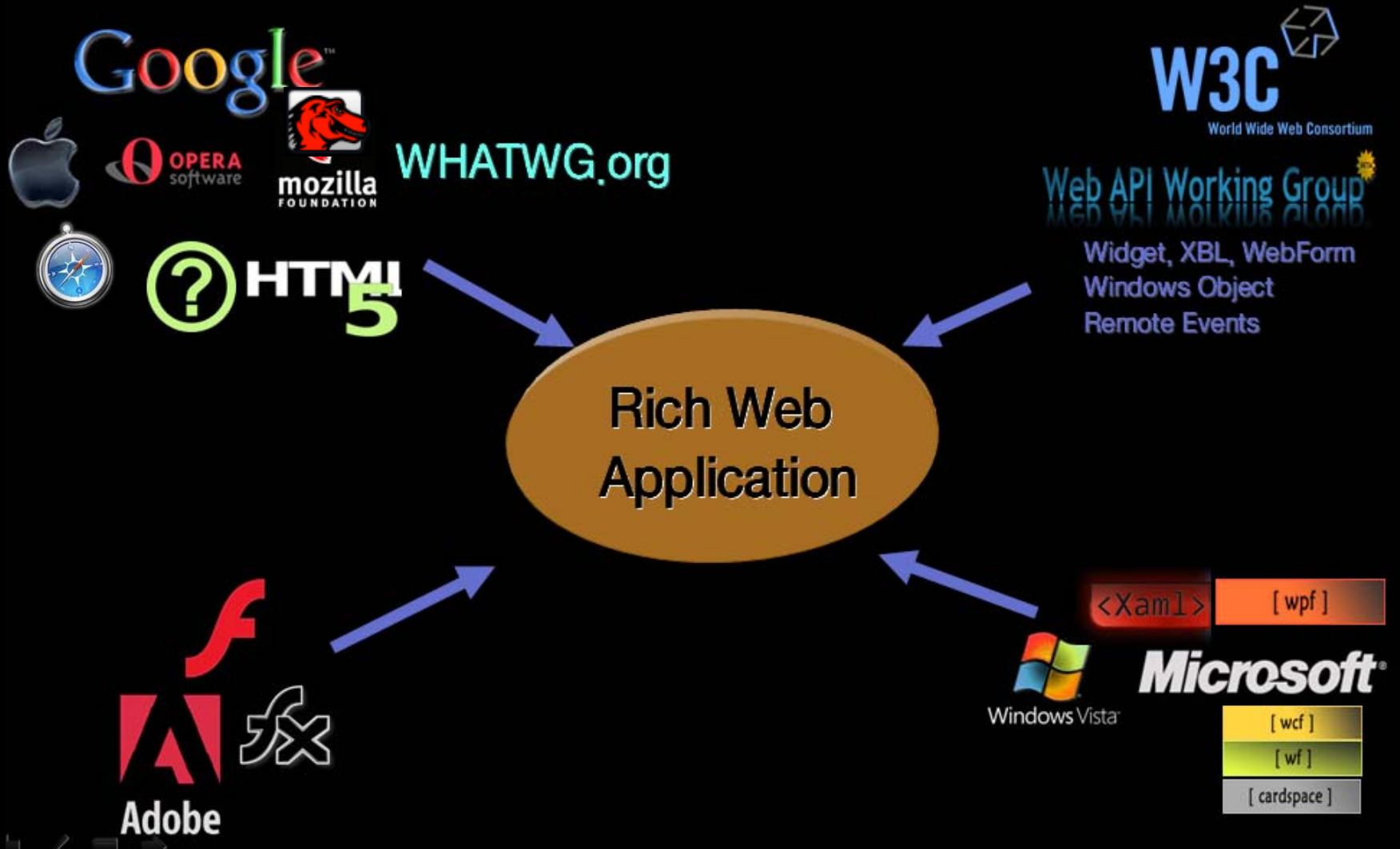
- 다양한 웹 브라우저 사용자를 위한 배려를 위해 객체 판별법 같은 방식으로 하위 버전 지원

웹 애플리케이션 기술

Trend 4: 소프트웨어 서비스



SaaS (Software as a Service)





- XML(MXML)+ ECMAScript (ActionScript) + SWF

빠른예매 | Ticketing

RESET CLOSE

영화선택

영화장르

(R) (더빙) 천년여우 여우비

(18) (디지털)300

(18) (IMAX DMR 2D)300

(15) 1번가의 기적

(18) 300

(12) 그 여자 작사, 그 남자 작곡

(12) 그놈 목소리

(15) 나비 효과 2

(12) 더 퀸

(12) 동경심판

(12) 드림 걸즈

(12) 리틀 러너

(15) 마강호텔

(12) 마미야 형제

(18) 바람피기 좋은 날

(18) 바벨

(12) 복면달호

(R) 비밀의 숲 테라비시아

극장선택

지역순

선택극장 : CGV 제주

서울

인천/경기

대전/충청

광주/전라

부산/경상

제주

CGV 제주

SWEET BOX

GOLD CLASS

EURO CLASS

STAR

IMAX

날짜/시간

날짜선택안내

선택하신 날짜 :

2007년 3월

일	월	화	수	목	금	토
				1	2	3
4	5	6	7	8	9	10
11	12	13	14	15	16	17
18	19	20	21	22	23	24
25	26	27	28	29	30	31

인원선택

예매정보

15 1번가의 기적

평점 ★★★★★

예매율 순위 4 위 / -5

관람객 순위 1 위 / -1

상영시간 113분

관람극장 CGV 제주

관람일자

관람시간

관람인원

예매좌석

좌석선택

Adobe Flex®

My First Flex App

Coffee Blends

Red Sea

▼

Description:
Smooth and
fragrant

Add to Cart

Blue Mountain

```
<?xml version="1.0"?>
<mx:Application xmlns:mx="http://www.macromedia.com/2003/mxml" >
  <!-- data model -->
  <mx:Array id="coffeeArray">
    <mx:Object>
      <label>Red Sea</label>
      <data>Smooth and fragrant</data>
    </mx:Object>
    <mx:Object>
      <label>Andes</label>
      <data>Rich and pungent</data>
    </mx:Object>
    <mx:Object>
      <label>Blue Mountain</label>
      <data>Delicate and refined</data>
    </mx:Object>
  </mx:Array>
  <mx:Script>
    <![CDATA[
function addToCart():Void {
  cart.addItem(coffeeCombo.selectedItem.label,coffeeCombo.selectedItem.data);
}
]]>
  </mx:Script>
  <!-- view -->
  <mx:Panel title="My First Flex App">
    <mx:Label text="Coffee Blends"/>
    <mx:ComboBox id="coffeeCombo" dataProvider="{coffeeArray}"/>
    <mx:Text text="Description: {coffeeCombo.selectedItem.data}"/>
    <mx:Button label="Add to Cart" click="addToCart()"/>
    <mx>List id="cart"/>
  </mx:Panel>
</mx:Application>
```

XUL Application for Firefox

- XUL + CSS + JavaScript (XPCOM)
- 수 백 개의 Firefox Extensions 개발 중

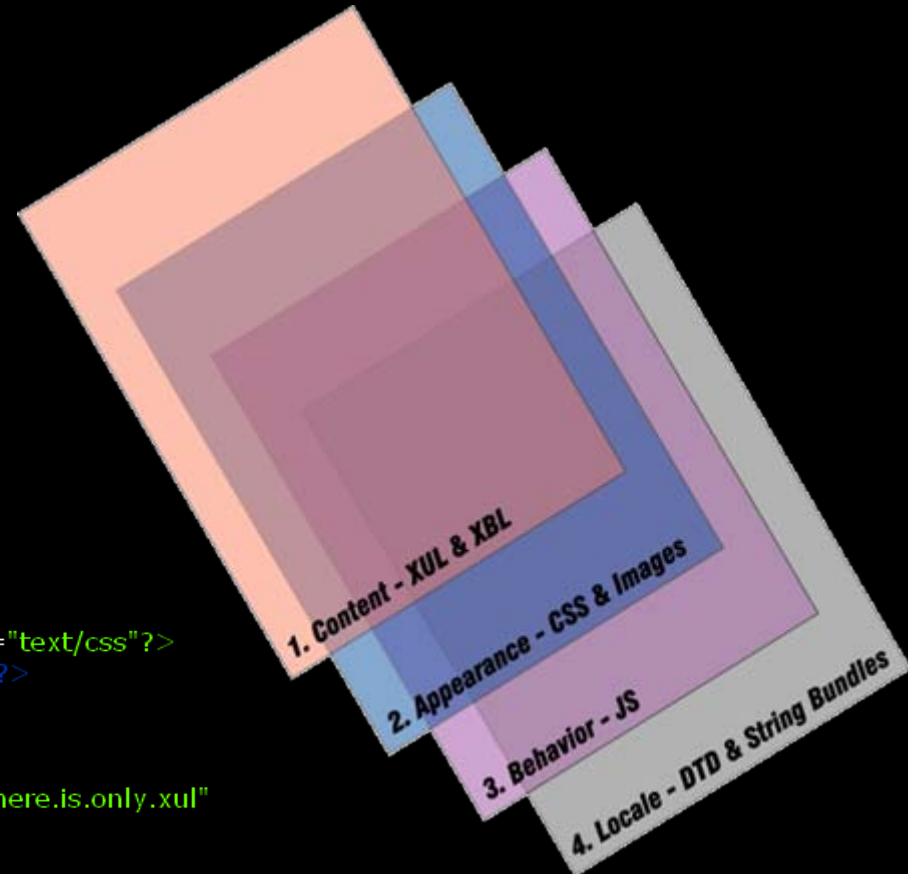


```
<?xml version="1.0"?>
<?xml-stylesheet href="chrome://mypackage/skin/" type="text/css"?>
<?xul-overlay href="chrome://mypackage/content/myOverlay.xul"?>

<window id="main-window"
  xmlns:html="http://www.w3.org/TR/REC-html40"
  xmlns="http://www.mozilla.org/keymaster/gatekeeper/there.is.only.xul"
  onload="HandleOnLoad()">

  <script language="JavaScript" src="chrome://mypackage/content/demo.js"/>
  <box orient="vertical">
    <button value="Spam" flex="2" oncommand="DoIt()" />
    <button value="Foo" flex="1" />
    <button value="Eggs" flex="1" />
  </box>

</window>
```



YAHOO! WIDGETS (aka Konfabulator)

- Graphic + JavaScript (Mozilla) Engine
- Widget: Javascript + HTML + CSS

Includes these helpful Yahoo! Widgets



SEARCH
Start your search right from the desktop



WEATHER
Get up-to-the-minute weather conditions.

TODAY SAT



Last Updated at 10:31A

INDU	10603.81	-23
Dow Jones Industrial Average		
COMPX	2169.45	-5
Nasdaq Combined Composite		
YHOO	33.42	0
Yahoo! Inc.		

FINANCE
Watch your portfolio in the stock ticker.

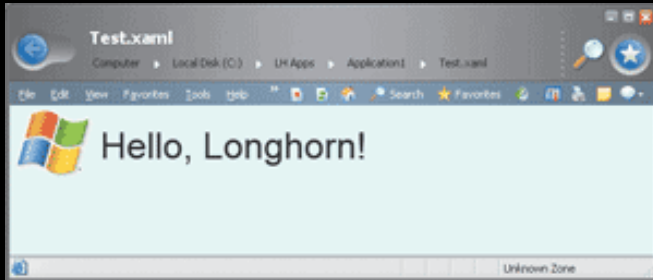


PICTURE FRAME
Display your photos and share them using Flickr.

The Widget Gallery



XAML on Microsoft Silverlight®



- Windows Vista 아발론 엔진 사용
- XAML과 간단한 C#코드로 RIA 개발 가능

```
<Canvas xmlns="http://schemas.microsoft.com/2003/xaml" Background="LightCyan"
Width="100%" Height="100%">
  <Image Source="lh.bmp" Canvas.Left="5" Canvas.Top="5" />
  <Text Canvas.Left="90" Canvas.Top="20" FontSize="36">Hello, Longhorn! </Text>
</Canvas>
<FlowPanel>
  <Button Background="Green" Click="EventY">Yes</Button>
  <Button Background="Red" Click="EventN">No</Button>
</FlowPanel>
```

.net Code

```
void EventY(...)
{
  MessageBox.Show("Agree");
}
```

C# Code

```
void EventN(...)
{
  MessageBox.Show("Disagree");
}
```

WHATWG



OPERA
software



mozilla

Google™



Web Applications 1.0

Working Draft — 12 March 2006



You can take part in this work. [Join the working group's discussion list.](#)

This version:

<http://www.whatwg.org/specs/web-apps/current-work/>

Editor:

Ian Hickson, Google, ian@hixie.ch

© Copyright 2004-2006 Apple Computer, Inc., Mozilla Foundation, and Opera Software ASA.
You are granted a license to use, reproduce and create derivative works of this document.

Web Forms 2.0

Working Draft — 3 March 2006

This version:

<http://www.whatwg.org/specs/web-forms/current-work/>



Rich Web Client Activity Proposal

1. Executive Summary

The Rich Web Client Activity manages the W3C's efforts in two Working Groups: [Web Applications](#) and [Compound Document Formats](#).

The mission of the [Web Applications Working Group](#) is to develop specifications that enable improved client-side application development, by building on existing standards and implementations. The [Compound Document Formats Working Group](#) is developing a framework for combining languages in a single document and a profile that uses the framework to combine XHTML and SVG.

This Activity subsumes the existing Compound Document Formats Activity and therefore the CDF Working Group.

Web Application Working Group Charter

Mission

The mission of the W3C Web Application Working Group is to develop specifications that enable improved client-side application development on the Web. This includes the development of languages for applications, especially user interfaces, and definition of programming interfaces to be made available in a Web client.

The target platforms for this Working Group includes desktop and mobile browsers as well as many speciality, browser-like environments that use Web client technologies. The goal is to promote universal access both for users and devices, including those with special needs.

Additionally, the Working Group has the goal to improve client-side application development through education, outreach and interoperability testing.

리치 웹의 전망

User Experience – 사용자 경험

Invisible Web – 브라우저 밖 웹

Software Web – 소프트웨어 웹

복습. DOM 스크립팅

■ 원칙 1. 단계적 기능 향상/축소

- 자바 스크립트 기능을 끄더라도 웹 문서의 주요 기능은 가능해야 함

■ 원칙 2. 스크립트 분리

- 가급적 마크업에 스크립트를 직접 쓰지 말 것
- 별도 파일로 함수를 분리할 것

■ 원칙 3. 하위 호환성

- 다양한 웹 브라우저 사용자를 위한 배려를 위해 객체 판별법 같은 방식으로 하위 버전 지원

복습: 웹애플리케이션

■ 리치 웹 어플리케이션

- User Interface: XML
- Presentation: CSS
- Behavior: JavaScript (script language...)
- Data: Web APIs (XML)

■ 주요 변화

- 벤더 중심: Microsoft, Adobe, Sun
- 브라우저 중심: W3C, 웹표준

웹 개발 프레임워크

Front-end

Back-end

Web Services

■ 스크립트 언어

- 전통적인 CGI 방식에서 빠른 개발로 변화
- ASP, JSP, PHP, Python

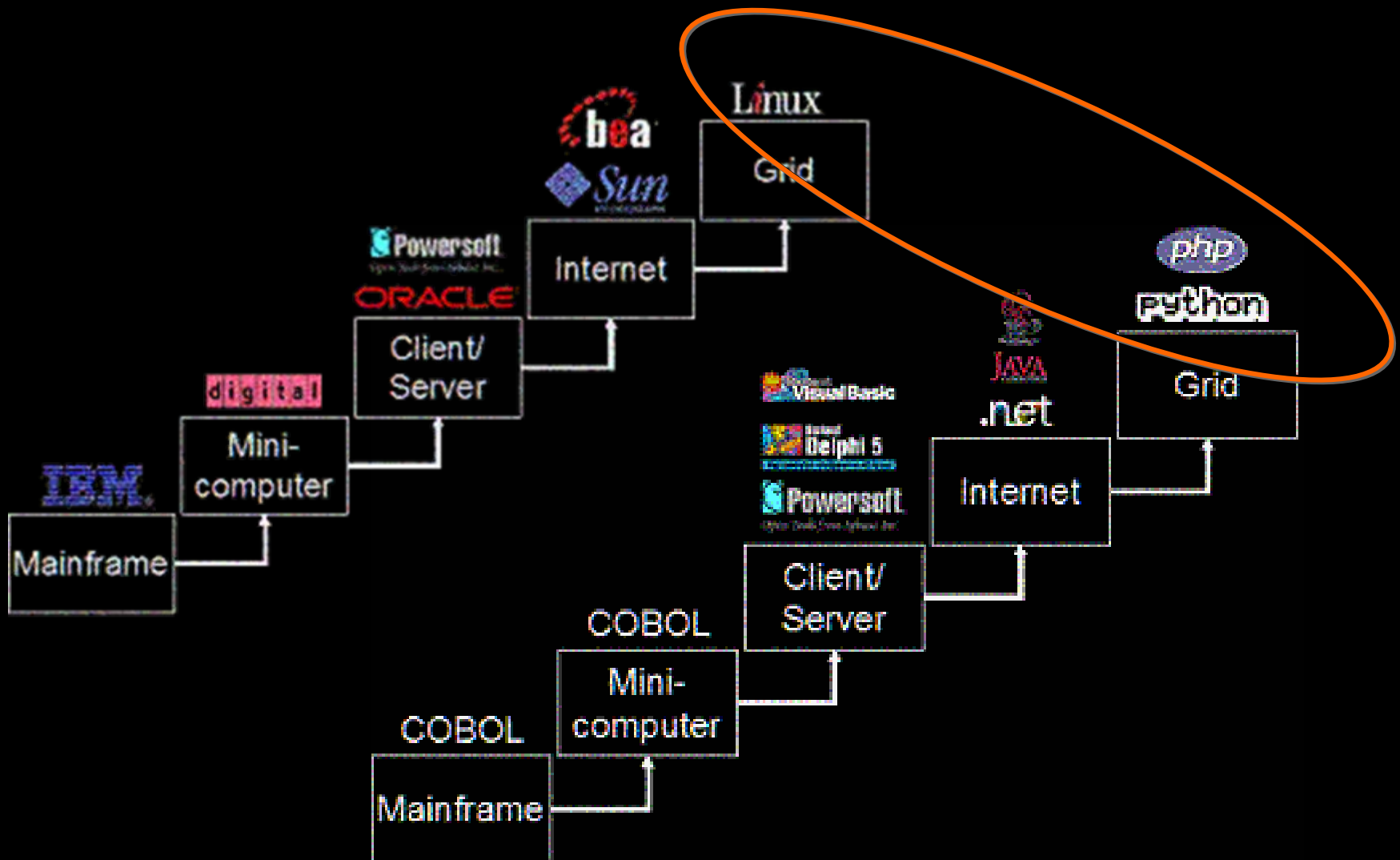
■ MVC 모델

- Model : 문서나 데이터 구조
- View: 보여주는 방법을 제공
- Controller: Model과 View를 연결

■ 웹 프레임워크 대거 등장

- Ruby on Rail, Django, cakePHP
- Java (Spring Framework)

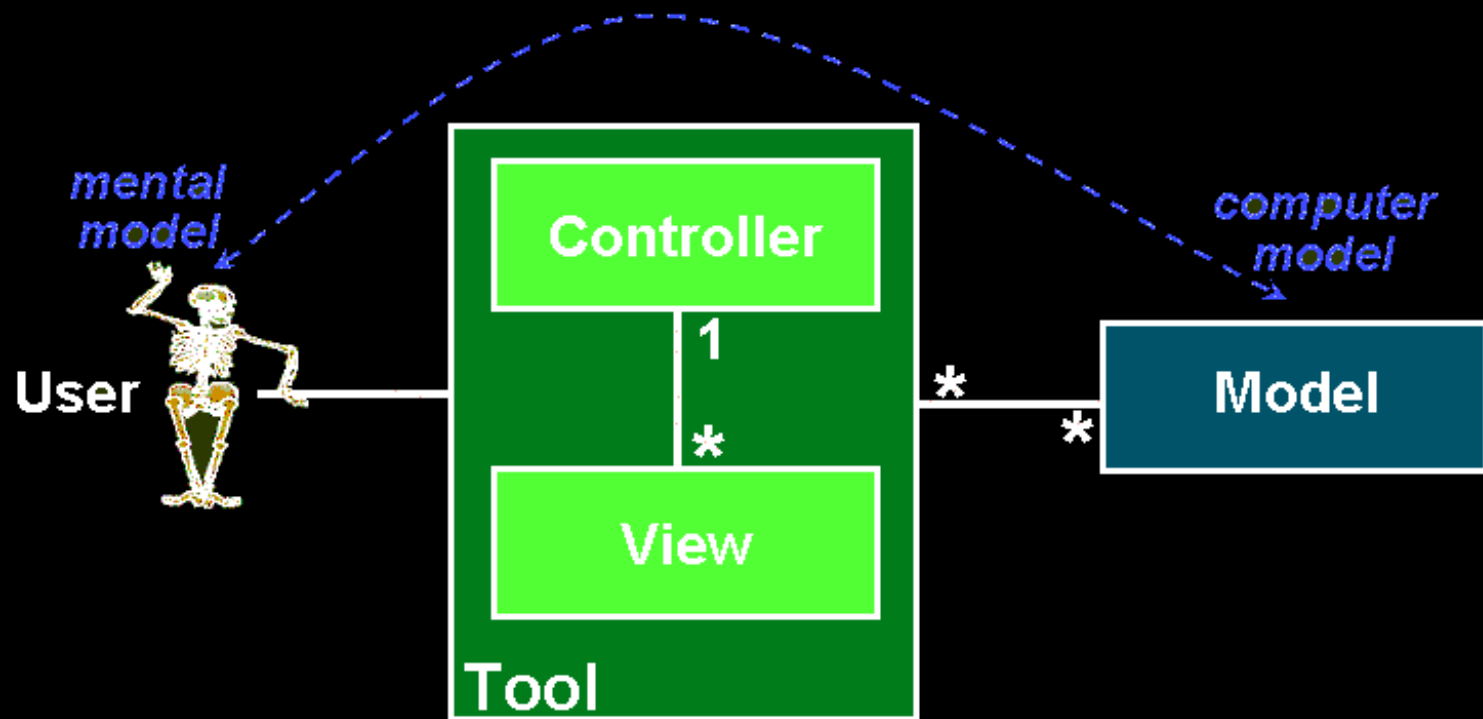
컴퓨팅 환경의 진화



웹 개발 프레임워크의 변화

- 90년대 중반: Client-Server (2-tier)
 - C, Perl CGI
- 2000년대 초반: Client-Server-미들웨어 (3-tier)
 - Java, .Net framework (Unix/Windows Server)
- 2000년대 중반: Client-Grid (2-tier)
 - PHP, Python (Linux)
- 현재: 빠른 개발 프레임워크
 - Ruby on Rails, Django, CakePHP

MVC 개발 방법론



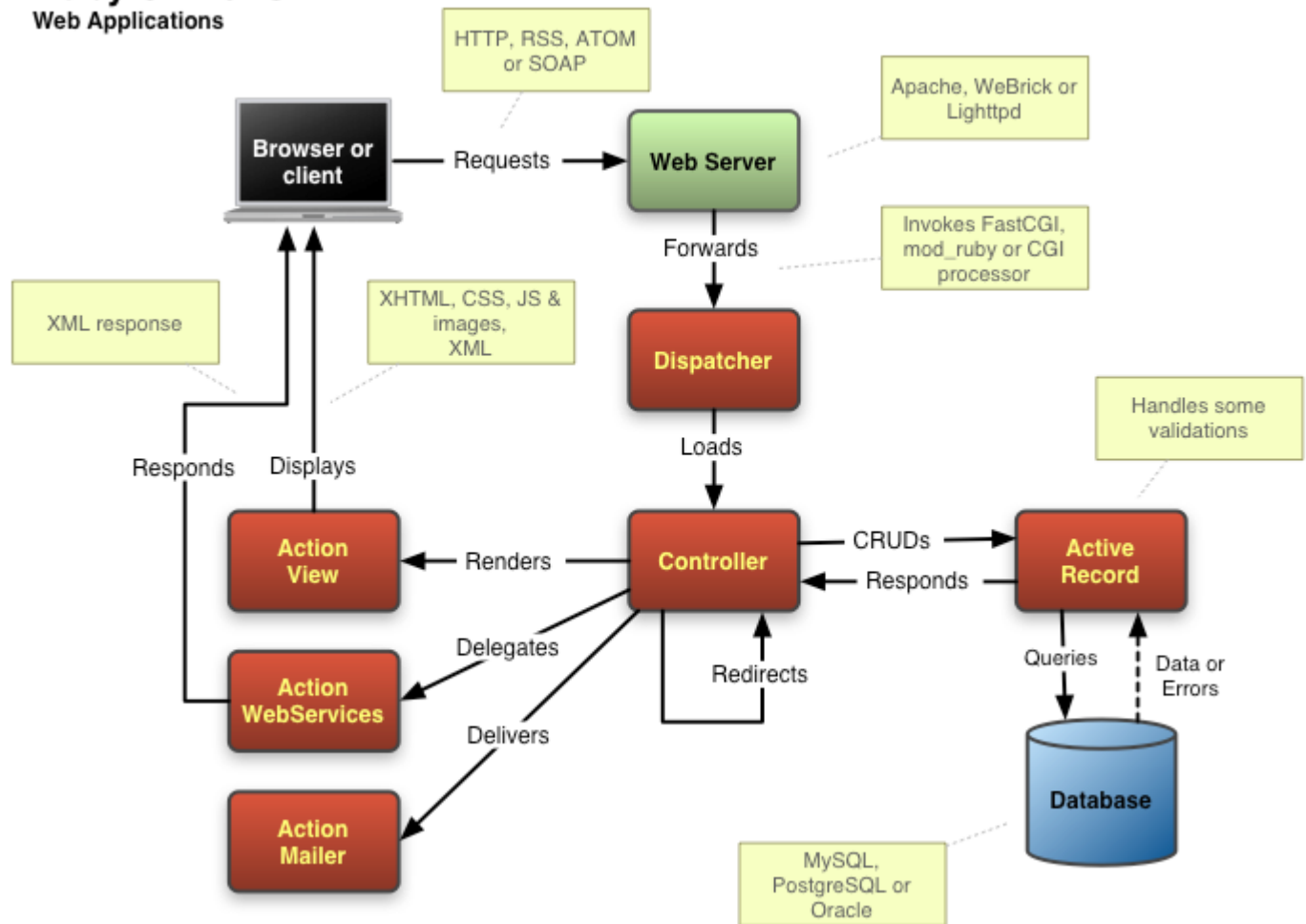
왜? RAD 프레임워크이 뜨나.

- MVC 모델이 적용 되어 있다.
- 코딩 규칙, 명명법, DB 구조 등이 이미 약속!
- 한번의 세팅, 바로 어플리케이션 개발 가능!
- Ajax, 자바 스크립트 등 다양한 UI 도구 내장.
- 코드 유효성 도구가 내장.
- 다양한 개발 라이브러리를 추가 가능.
- 오픈 소스 개발로 무료로 많은 참여자가 확보 가능.



Ruby on Rails

Web Applications





CakePHP Naming Conventions

Models	Controllers	Views
Class Name: singular, camel cased (LineItem, Person)	Class Name: plural, camel cased, ends in "Controller" (LineItemsController, PeopleController)	Path: controller name, underscored (app/views/line_items/<file>, app/ views/people/<file>)
File Name: singular, underscored (line_item.php, person.php)	File Name: plural, underscored (line_items_controller.php, people_controller.php)	File Name: action name, lowercase (index.html, view.html)
Table Name: plural, underscored (line_items, people)		

Global Functions

config ('file_name')	up ('string')
uses ('file_name')	r ('search', 'replace', 'text')
vendor ('file_name')	pr (array)
debug ('message')	am (array, [array, array])
a (element, [element, element])	env ('HTTP_HEADER')
aa (key, value, [key, value])	cache (path, data, expires, [target])
e ('message')	clearCache ([params, type, ext])
low ('STRING')	countdim (array)

Model

Properties		Association Properties
\$cacheQueries	\$recursive	\$belongsTo
\$data	\$useDbConfig	\$hasAndBelongsToMany
\$displayField	\$useTable	\$hasMany
\$id	\$validate	\$hasOne
\$name	\$validationErrors	
\$primaryKey	\$_tableInfo	
Methods		
bindModel (params)	getLastInsertID ()	
create ()	getNumRows ()	
delete (id, [cascade])	hasAny ([conditions])	
escapeField (field)	hasField (name)	
execute (data)	invalidate (field)	
exists ()	invalidFields ([data])	
field (name, conditions, order)	isForeignKey (field)	
find ([conditions, fields, order, recursive])	loadInfo ()	
findAll ([conditions, fields, order, limit, name, recur])	query ([sql])	

Controller

Properties	
\$name	\$action
\$autoLayout	\$autoRender
\$base	\$beforeFilter
\$cacheAction	\$components
\$data	\$helpers
\$here	\$layout
\$output	\$pageTitle
\$params	\$persistModel
\$plugin	\$uses
\$view	\$viewPath
\$webroot	
Methods	
cleanupFields ()	
constructClasses ()	
flash (message, url, [pause])	
flashOut (message, url, [pause])	

View

Properties	
\$action	\$autoLayout
\$autoRender	\$base
\$controller	\$ext
\$hasRendered	\$helpers
\$here	\$layout
\$loaded	\$models
\$name	\$pageTitle
\$params	\$parent
\$plugin	\$subDir
\$themeWeb	\$uses
\$viewPath	
Methods	
element (name)	
error (code, name, message)	
pluginView (action, layout)	
render ([action, layout, file])	

ScreenCast Demo

15분만에 블로그 만들기