



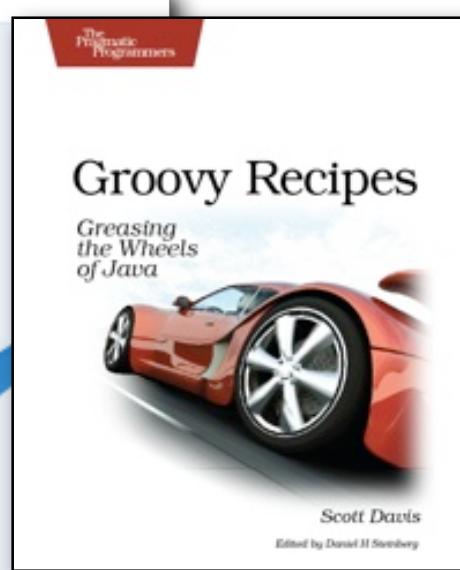
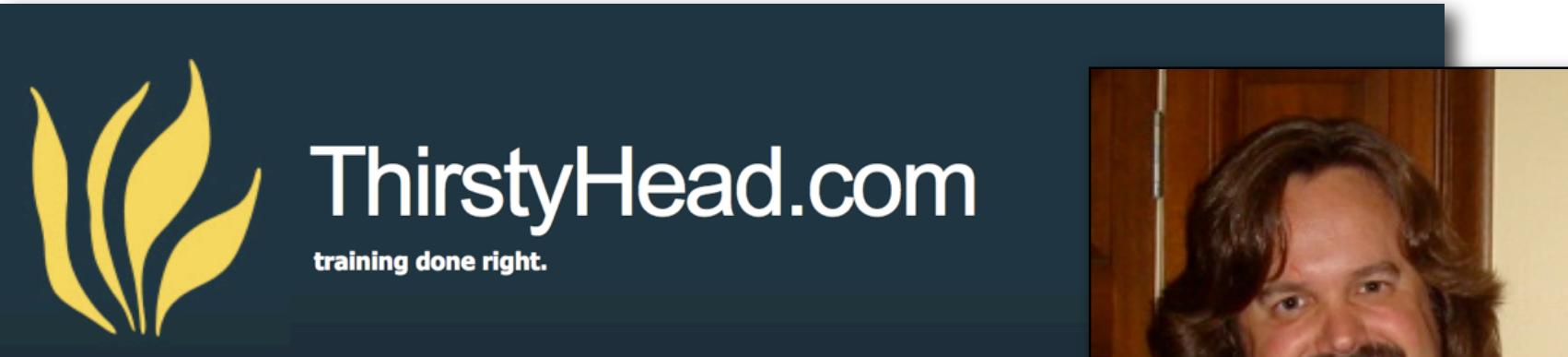
Tomorrow's Tech Today: HTML5

Scott Davis, ThirstyHead.com



ThirstyHead.com

training done right.



Scott Davis



developerWorks > Java technology

Mastering Grails Practically Groovy

Do you know HTML?

*Have you had any
formal training in
HTML?*



5





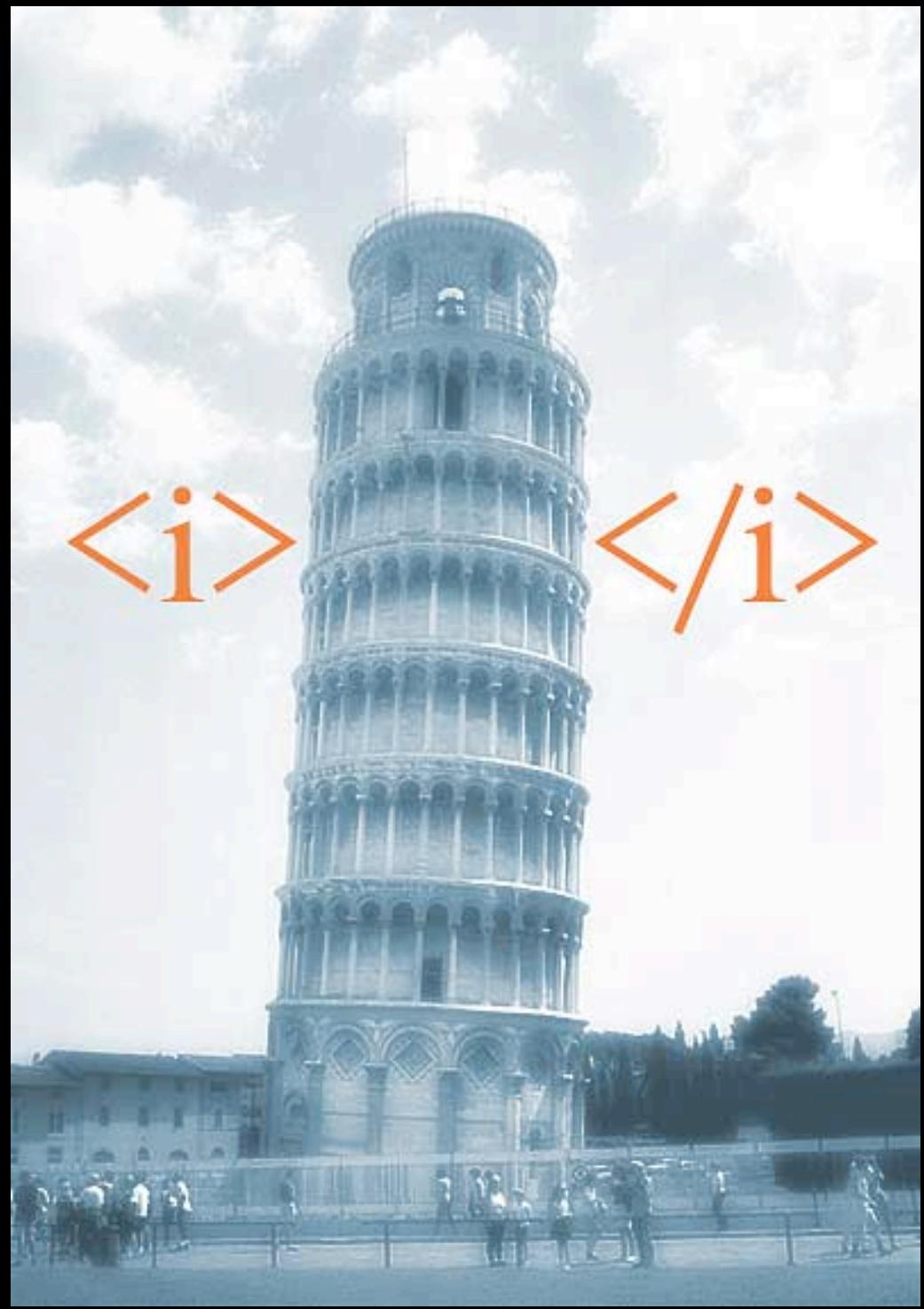
New elements

Form enhancements

Mobile Support

Video

*Semantics
over
Markup*



HTML 4

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN"  
"http://www.w3.org/TR/html4/strict.dtd">  
<html>  
  <body>  
    <div id="header">aboutGroovy.com</div>  
  
    <div id="nav">  
      <a href="...">News</a>  
      <a href="...">Podcasts</a>  
      <a href="...">Books</a>  
    </div>  
  
    <div id="section">  
      <div class="article">...</div>  
      <div class="article">...</div>  
      <div class="article">...</div>  
    </div>  
  
    <div id="footer">&copy; 2010, ThirstyHead.com</div>  
  </body>  
</html>
```

HTML 5

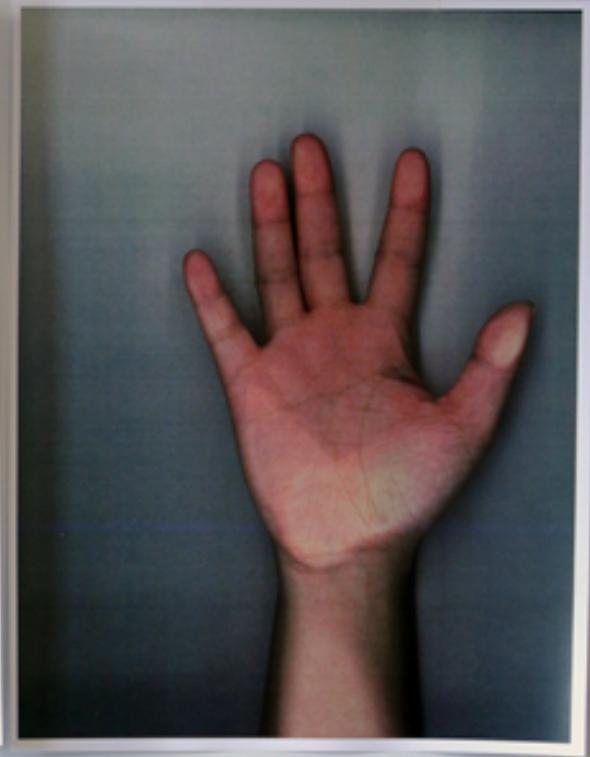
```
<!doctype html>
<html>
  <body>
    <header>aboutGroovy.com</header>

    <nav>
      <a href="...">News</a>
      <a href="...">Podcasts</a>
      <a href="...">Books</a>
    </nav>

    <section>
      <article>...</article>
      <article>...</article>
      <article>...</article>
    </section>

    <footer>&copy; 2010, ThirstyHead.com</footer>
  </body>
</html>
```

Who is using HTML5?



Wed

May 27
2009



Google Bets Big on HTML 5: News from Google I/O

by [Tim O'Reilly](#) | [@timoreilly](#) | comments: 69

listen 

print 

"Never underestimate the web," says Google VP of Engineering [Vic Gundotra](#) in his keynote at [Google I/O](#) this morning. He goes on to tell the story of a meeting he remembers when he was VP of Platform Evangelism at Microsoft five years ago. "We believed that web apps would never rival desktop apps. There was this small company called [Keyhole](#), which made this most fantastic geo-visualization software for Windows. This was the kind of software we always used to prove to ourselves that there were things that could *never* be done on the web." A few months later, Google acquired Keyhole, and shortly thereafter released Google Maps with satellite view.

"We knew then that the web had won," he said. "What was once thought impossible is now commonplace."

Google doesn't want to repeat that mistake, and as a result, he said, "we're betting big on HTML 5."



<!doctype html>

Google Chrome

Apple shows off Safari's HTML 5 support

Posted on Mar 9, 2009 2:54 pm by Jim Dalrymple, Macworld.com

Category | Mac » Software » E-mail and Internet

With Safari 4's [improved history search](#), Top Sites and [controversial tabbed windows](#) features garnering much of the attention, users have largely overlooked support for HTML 5 introduced in the [public beta of the next version of Apple's Web browser](#). [HTML 5](#) is the latest Hypertext Markup Language—the core language for the Internet. Web developers use this programming language to build the Web sites that we visit every day.



<!doctype html>

Web Features

The **demo** uses the following features from HTML5 included in Firefox 3.6:

- **HTML5 Drag and Drop:** You can drag and drop items inside of the web page and drag images from your desktop directly into the browser.
- **HTML5 localStorage:** to store the image data across browser restarts
- **HTML5 Application Cache:** This allows you to create applications that can be used when the browser isn't connected to the Internet. It stores the entire app in the browser and also gives you access to online and offline events so you know when you need to re-sync data with a server.
- **HTML5 Canvas:** The HTML5 Canvas element is used through this demo to edit and render images.
- **Cross-Origin Resource Sharing** to host an application at one site and publish data to another.

<!doctype html>





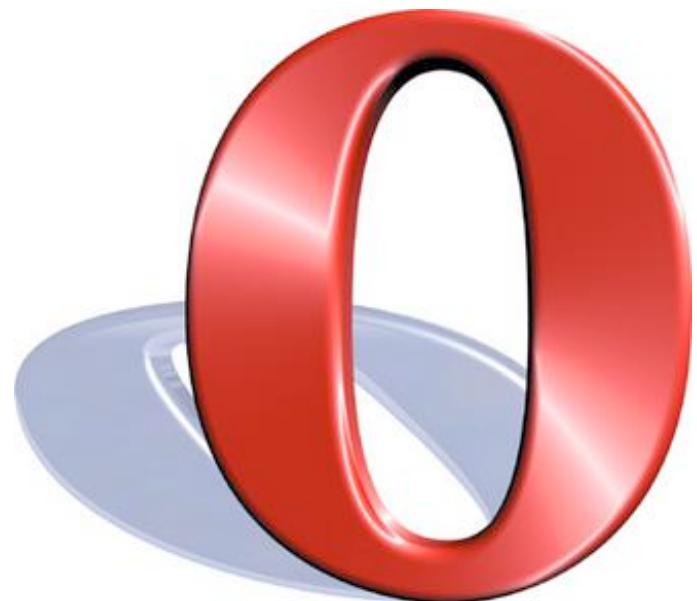
Opera 10.50 Web browser

The fastest browser on Earth

Surf the Web with lightning speed. Check out the elegant, new design and many other improvements in Opera 10.50.

Rewriting the Web with HTML5 and CSS3

Opera 10.50 includes improved standards support for HTML5 and CSS3, giving Web designers the tools they need to make eye-catching Web applications that work on completely open technologies. [Read more](#) for complete information on standards support in Opera 10.50.



MARCH 16, 2010

Microsoft embraces HTML5 specification in IE9

The move could be at odds with company's promotion of its own Silverlight RIA technology

By [Paul Krill](#) | [InfoWorld](#)

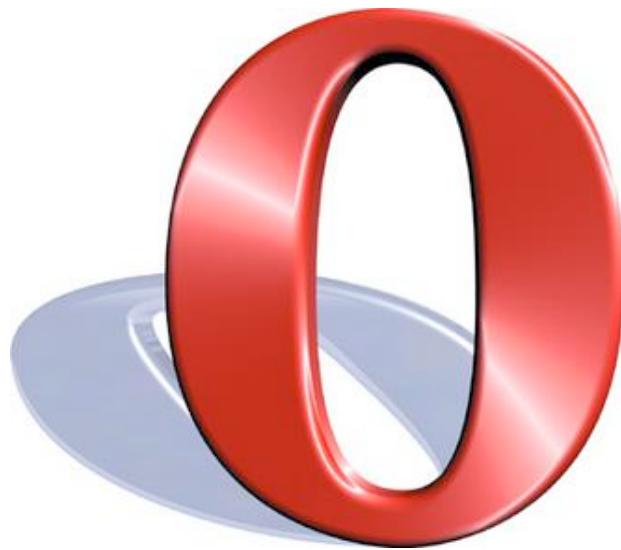
[HTML5](#) has the potential to bring standards-based multimedia capabilities to applications that rival what is possible now with proprietary plug-in software like Adobe Systems Flash and, yes, Silverlight. But Microsoft nonetheless appears gung-ho on HTML5, which has been in development for several years.

"We love HTML5 so much we want it to actually work," said Dean Hachamovitch, general manager of Internet Explorer at Microsoft, during a keynote presentation. "In IE9, it will."

Company officials even demonstrated support for HTML5 [SVG \(Scalable Vector Graphics\)](#) in IE9. Previously, Microsoft has not backed the graphics standard. Additionally, an update planned for the IE9 platform preview will add support for HTML5 video capabilities.

"When we started looking deeply at HTML5, we saw that it enabled a whole new class of applications," Hachamovitch said.









HTML5 from a Mobile Perspective

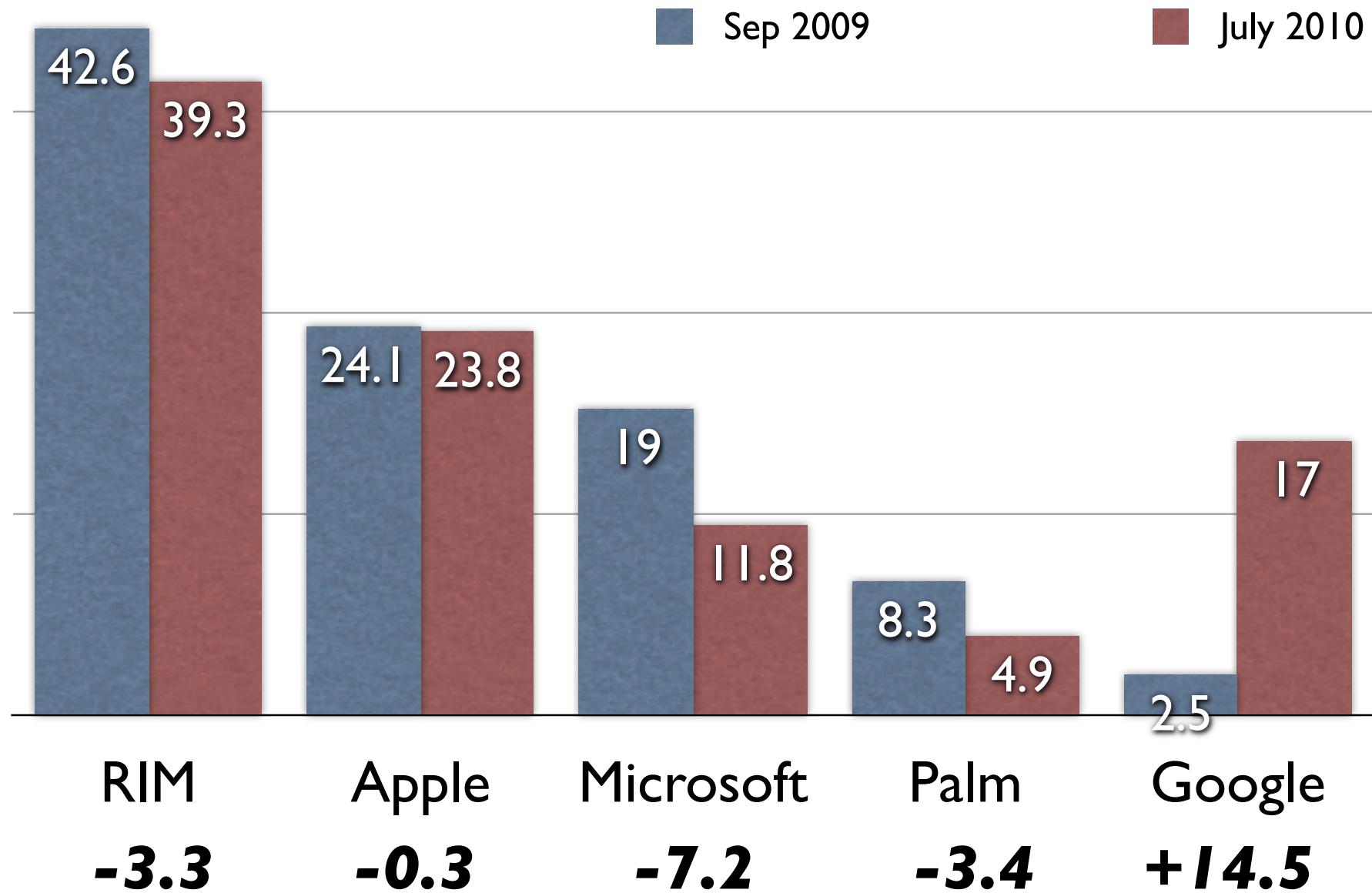
July 22nd, 2009 by Jason Grigsby

HTML5 for Mobile

HTML5 is a critical step for mobile web application development. Some of the key elements that it provides are:

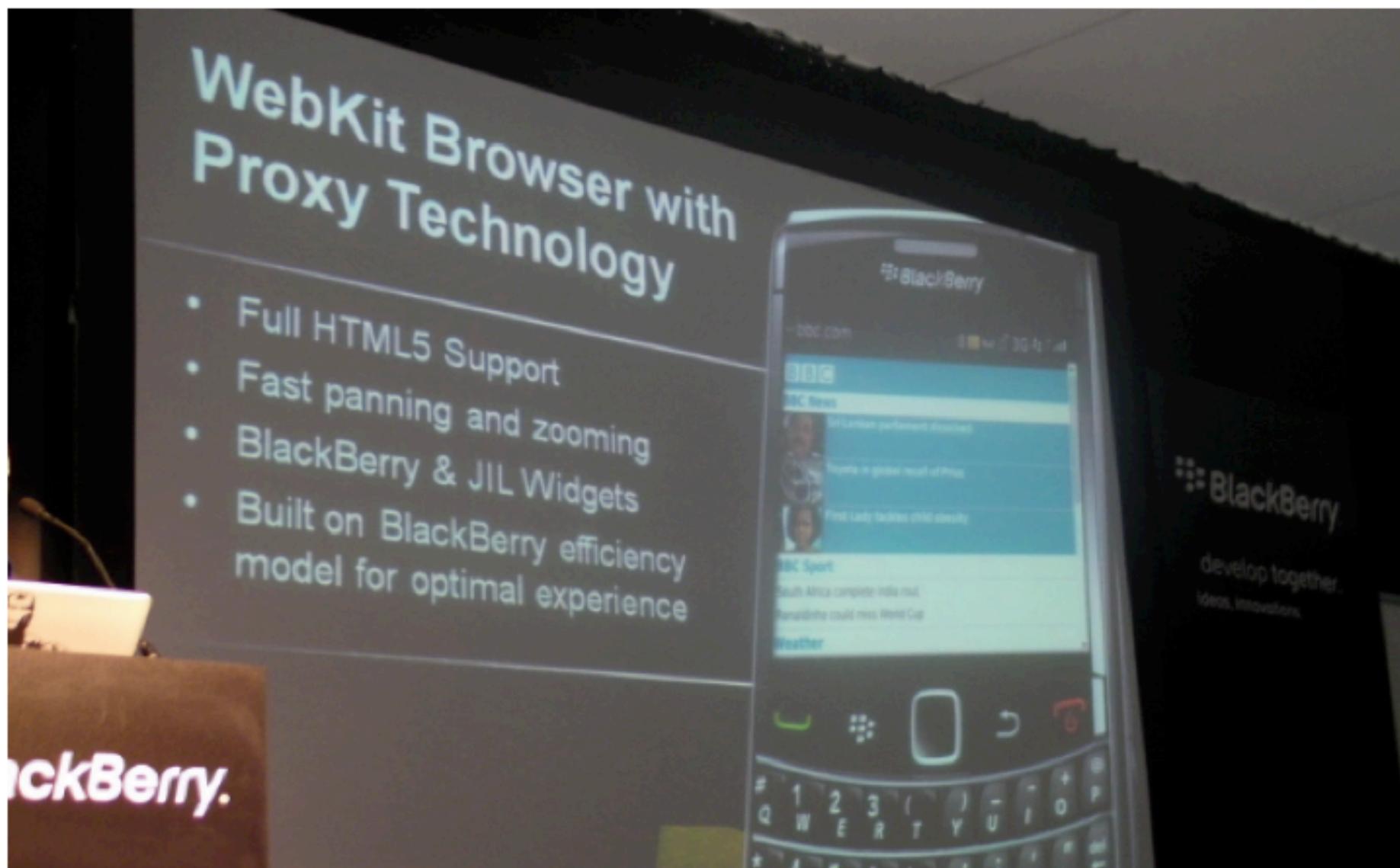
- **Offline Support** — The AppCache and Database make it possible for mobile developers to store things locally on the device and now that interruptions in connectivity will not affect the ability for someone to get their work done.
- **Canvas and Video** — These two features are designed to make it easy to add graphics and video to a page without worrying about plugins. When supported by the phone's hardware, as is the case with the iPhone, they provide a powerful way to get media into a page.
- **GeoLocation API** — This is actually not part of HTML5, but is a [separate specification](#). That said, it is often bundled together because the mobile phones that are including HTML5 are generally supporting the GeoLocation API.
- **Advanced Forms** — Even simple things like the improvements in HTML5 for forms could make life easier for mobile applications. Fields that can be validated by the browser are improvements for mobile devices. The more that can be handled by the browser means less time downloading javascript code and less round trips to the server if validation can be found before the form is posted.

comScore Reports July 2010 U.S. Mobile Subscriber Market Share



RIM announces new BlackBerry web browser, 'SuperApps', BES Express server

February 16, 2010 | Devindra Hardawar | 1 Comment  |  Share 9 | 68 





Good news- HTML5 coming to Windows Phone IE Mobile after all

October 8th, 2010 | Author: Surur

[Share](#) A few months ago we reported Microsoft Technology Evangelist Frank Prengel telling the world [Microsoft had no concrete plans for HTML5 in Windows Phone 7](#).

We can now thankfully report that things have changed at the software giant.

Microsoft has posted [a job description](#) asking for a Senior software Development Engineer for the Windows Phone Browser, Search & Maps team responsible for the IE browser in Windows Phone 7.

The division is looking for “*a strong senior engineer that is excited about the product and technical opportunities with the mobile web experience*” to work on “*building the next generation IE platform for the next Windows Phone release*” which would include “*a major overhaul of standard support and new approaches to make significant advances in performance, power consumption and bandwidth utilization*”, with the candidate needing “*a strong interest in JavaScript, CSS3, HTML5 and Windows programming*”.

Now the only question is if this software will come as an update to Windows Phone 7, or will wait till Windows Phone 8.



42

[Tweet](#)

Apple And Android Now Make Up 75 Percent Of U.S. Smartphone Web Traffic

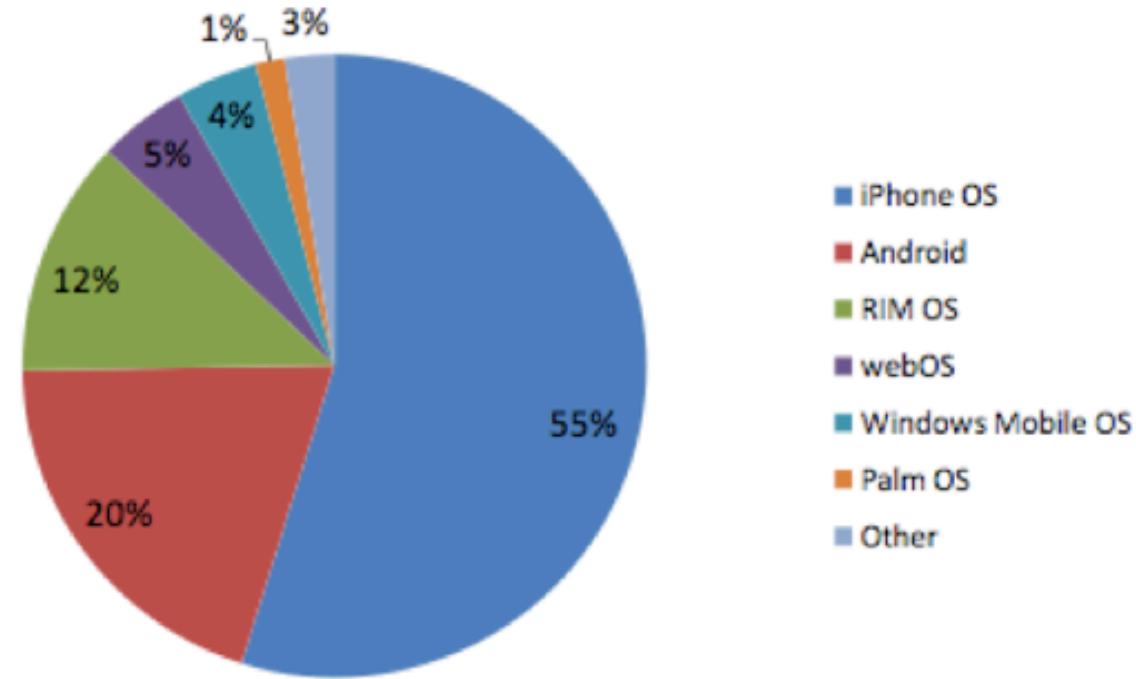
by Erick Schonfeld on Nov 23, 2009 116 Comments 940 retweet

f Share 154

Buzz it

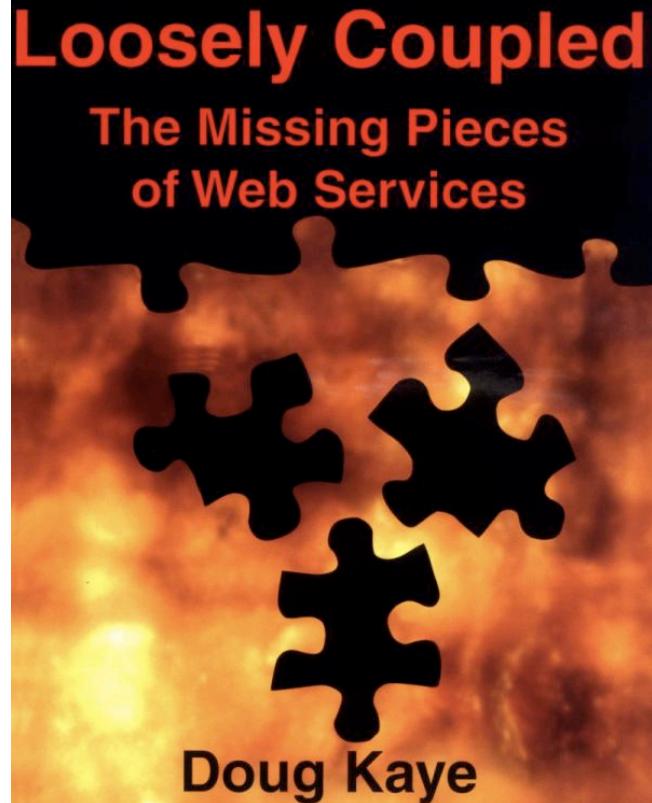
When it comes to the mobile Web, increasingly there are only two mobile platforms which matter: Apple and Android. According to AdMob's [October, 2009 mobile metrics report](#), the iPhone/iPod Touch and Android phones accounted for 75 percent of mobile Web traffic in the U.S., as measured by all the mobile ad requests it tracks. That number is up from a combined 65 percent in September, 2009.

Smartphone Requests by OS: US

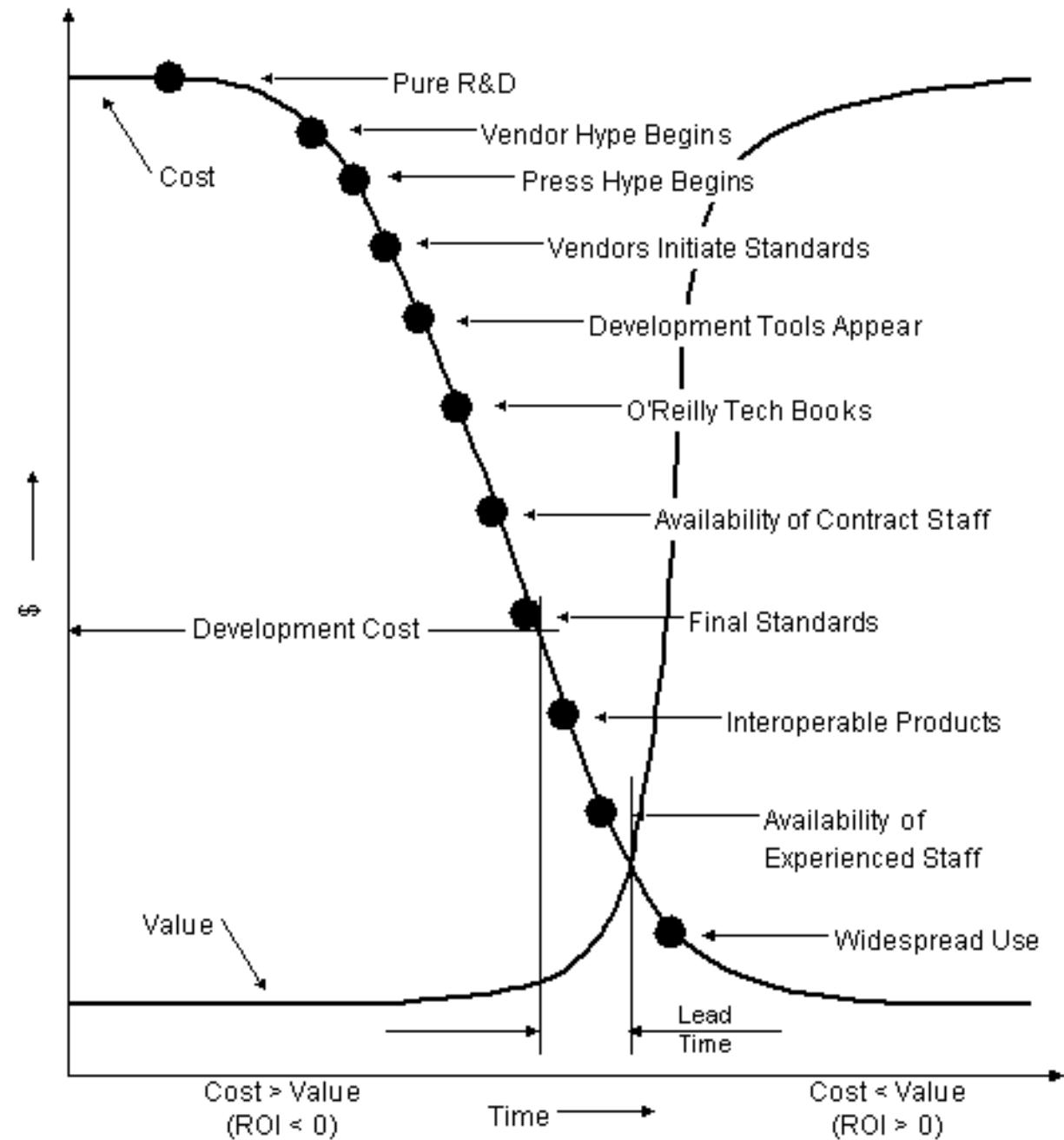


Note: Above share % refers only to requests from Smartphones

Copyrighted Material



The O'Reilly Curve



HTML5: Up and Running

By [Mark Pilgrim](#)

Publisher: O'Reilly Media

Released: June 2010 (est.)

Pages: 250 (est.)



HTML5

Up and Running

O'REILLY®

Mark Pilgrim

<http://diveintohtml5.org/>

DIVE INTO HTML5

BY

MARK PILGRIM

WITH ILLUSTRATIONS FROM THE PUBLIC DOMAIN

<new elements />

THE DOCTYPE

Now then. Where were we? Ah yes, the doctype:

```
<!DOCTYPE html  
    PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"  
    "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
```

That happens to be one of the 15 doctypes that trigger “standards mode” in all modern browsers. There is nothing wrong with it. If you like it, you can keep it. Or you can change it to the HTML5 doctype, which is shorter and sweeter and also triggers “standards mode” in all modern browsers.

This is the HTML5 doctype:

```
<!DOCTYPE html>
```

NEW SEMANTIC ELEMENTS IN HTML5

HTML5 is not just about making existing markup shorter (although it does a fair amount of that). It also defines a number of new semantic elements.

<header>

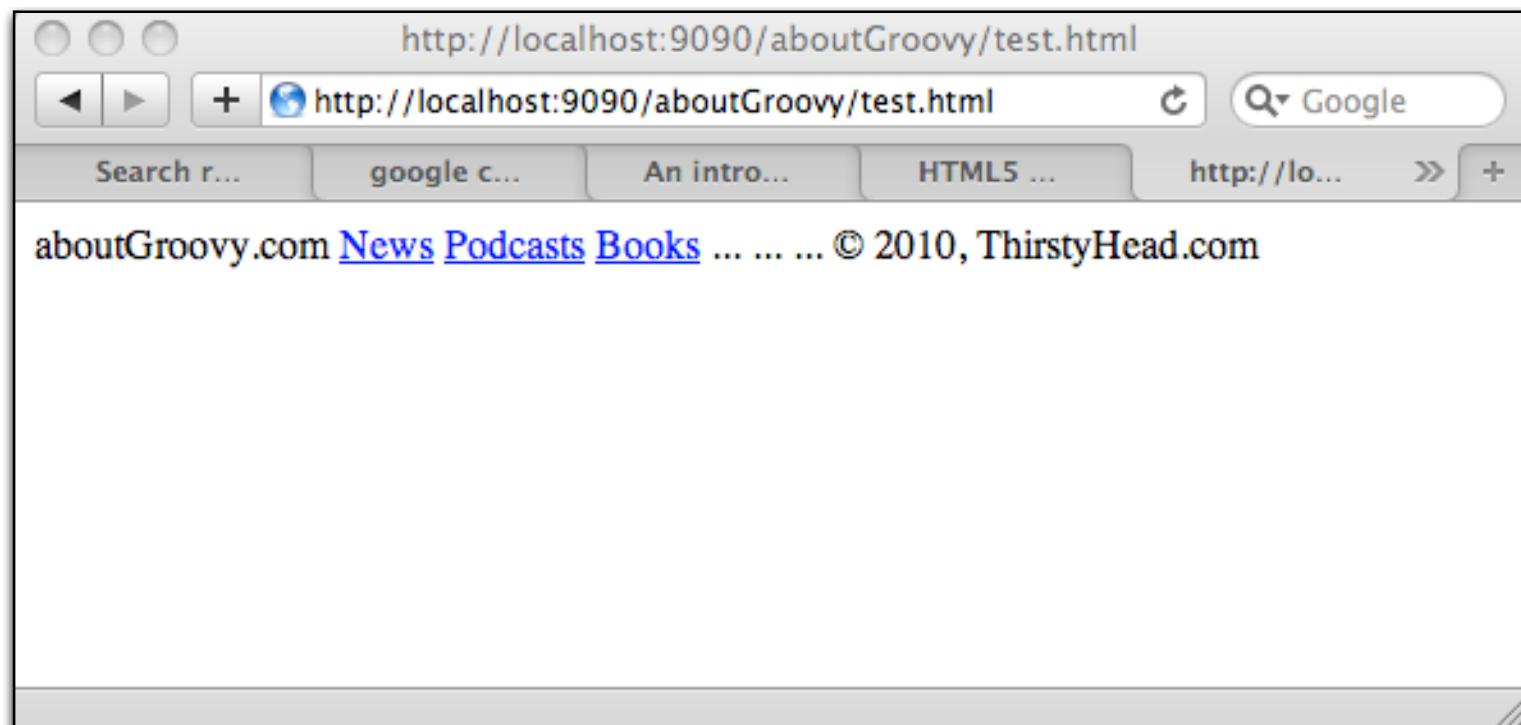
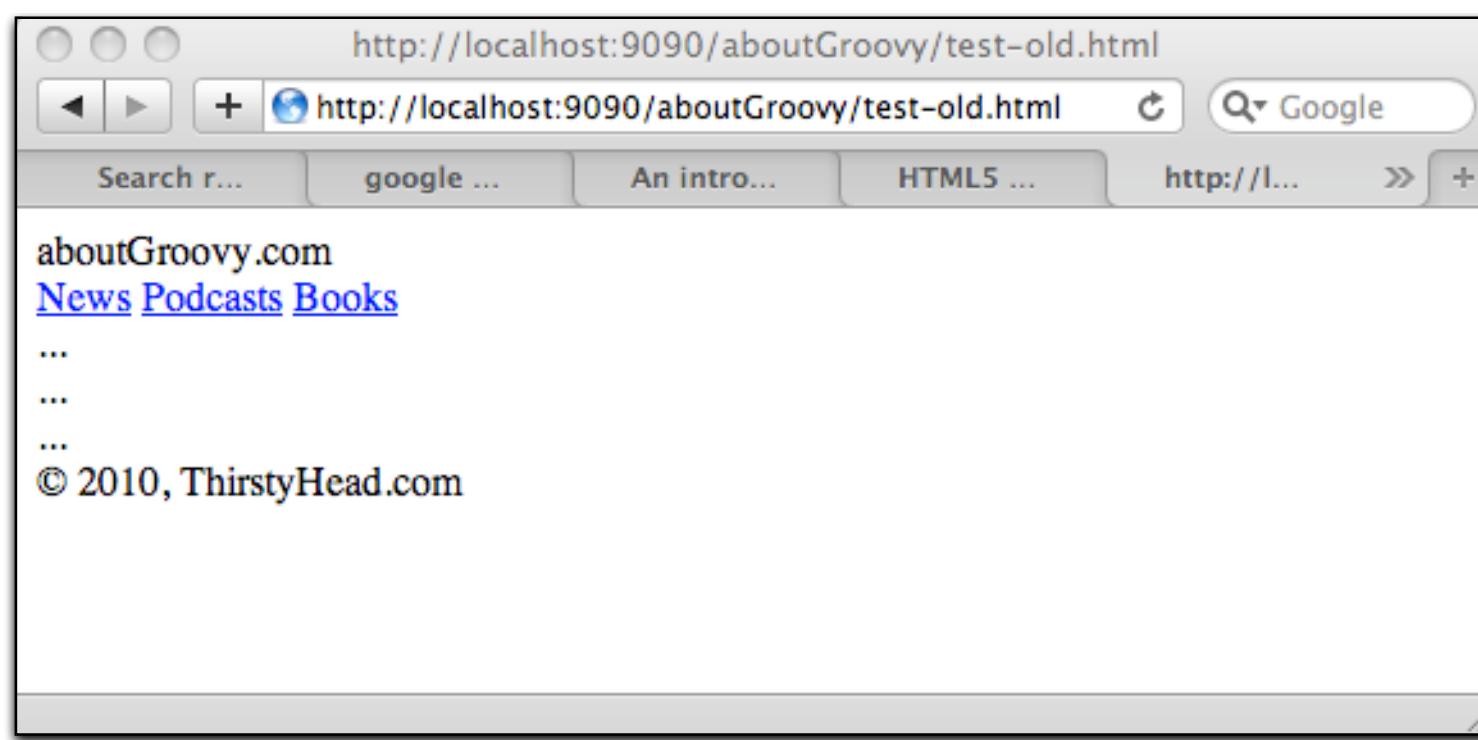
<footer>

<nav>

<section>

<article>

<aside>



<html>5 doctor

[Home](#) [About](#) [Article Archive](#) [Ask the doctor](#) [Contact](#) [Glossary](#) [RSS](#) 

HTML5 Reset Stylesheet

We've had a number of people ask about templates, boilerplates, and styling for HTML 5. Last week, [Remy](#) introduced some [basic boilerplates for HTML 5](#), so to keep the momentum going, I've modified [Eric Meyer's CSS reset](#) for you to use in your HTML 5 projects.

```
article,aside,canvas,details,figcaption,figure,  
footer,header,hgroup,menu,nav,section,summary {  
    display:block;  
}
```

IE doesn't believe in HTML 5 elements

Quite simply, IE doesn't even *see* HTML 5 elements, much less style them.

This is actually the same issue that we had before HTML 5, where the `<abbr>` element couldn't be styled in IE 6, resulting in [all manner of workarounds](#). (Let me add that we'll also fix the `<abbr>` element while we convince IE to recognise HTML 5 elements).

The fix

There is hope! The trick, discovered by [Sjoerd Visscher](#), is simply to create the new element using JavaScript, and *voilà*, IE is able to style it:

```
document.createElement('header');
```

John Resig has also written about this [HTML 5 shiv](#).

« [CSS Tricks' Link Nudge](#)

[Element 'in view' Event Plugin](#) »

HTML5 enabling script

Since HTML5 is getting [more attention](#) by way of marking up our new pages, and the only way to get IE to acknowledge the new elements, such as `<article>`, is to use the [HTML5 shiv](#), I've quickly put together a mini script that enables *all* the new elements.

Usage & Download

The [html5.js](#) and must be inserted in the `head` element (this is because IE needs to know about the element before it comes to render them - so it can't sit in the footer of the page, i.e. below the elements in question).

I've now moved HTML5 shiv to be hosted on a Google code project with the correct mime type being served, so if you're happy with the extra HTTP request, you can hot link the script: <http://html5shiv.googlecode.com/svn/trunk/html5.js>

It's conditional within the code, so Firefox et al won't run the code - but it doesn't hurt to wrap it in an IE conditional call to reduce the http pulls for other browsers:

```
<!--[if IE]>
<script src="http://html5shiv.googlecode.com/svn/trunk/html5.js"></script>
<![endif]-->
```

I've also minified the file, so it's a matter of bytes to download.

```
// For discussion and comments, see: http://remysharp.com/2009/01/07/html5-enabling-script/
(credit to @jdalton for minif)
/*@cc_on'abbr article aside audio canvas details
figcaption figure footer header hgroup mark menu
meter nav output progress section summary time
video'.replace(/\w+/g,function(n)
{document.createElement(n)})@*/
```

<form enhancements />

N_o 10.

A FORM OF MADNESS

For over a decade, web forms comprised just a few kinds of fields. The most common kinds were

Field Type	HTML Code	Notes
checkbox	<input type="checkbox">	can be toggled on or off
radio button	<input type="radio">	can be grouped with other inputs
password field	<input type="password">	echos dots instead of characters as you type
drop-down lists	<select><option>...	
file picker	<input type="file">	pops up an “open file” dialog
submit button	<input type="submit">	
plain text	<input type="text">	the type attribute can be omitted

All of these input types still work in HTML5. If you’re “upgrading to HTML5” (perhaps by [changing your DOCTYPE](#)), you don’t need to make a single change to your web forms. Hooray for backward compatibility!

<http://diveintohtml5.org/>

The `input` element's `type` attribute now has the following new values:

- `tel`
- `search`
- `url`
- `email`
- `datetime`
- `date`
- `month`
- `week`
- `time`
- `datetime-local`
- `number`
- `range`
- `color`

The idea of these new types is that the user agent can provide the user interface, such as a calendar date picker or integration with the user's address book, and submit a defined format to the server. It gives the user a better experience as his input is checked before sending it to the server meaning there is less time to wait for feedback.

The first of these new input types is for email addresses. It looks like this:

```
<form>
  <input type="email">
  <input type="submit" value="Go">
</form>
```

I was about to write a sentence that started with “in browsers that don’t support type=”email”...” but I stopped myself. Why? Because I’m not sure what it would mean to say that a browser doesn’t support type=”email”. All browsers “support” type=”email”. They may not do anything special with it (you’ll see a few examples of special treatment in a moment), but browsers that don’t recognize type=”email” will treat it as type=”text” and render it as a plain text field.

And then there's the iPhone.

The iPhone does not have a physical keyboard. All "typing" is done by tapping on an on-screen keyboard that pops up at appropriate times, like when you focus a form field in a web page. Apple did something very clever in the iPhone's web browser. It recognizes several of the new HTML5 input types, and **DYNAMICALLY CHANGES THE ON-SCREEN KEYBOARD** to optimize for that kind of input.



<http://diveintohtml5.org/>

aboutGroovy.com

Plain old text box:

Email:

Spinner: 

Slider: 

Date:

DateTime:   UTC

Search:

Search with Placeholder:

Autofocus field:

```
<form>
    Plain old text box: <input type="text" /><br />

    Email: <input type="email" /><br />

    Spinner: <input type="number"
                  min="0"
                  max="10"
                  step="2"
                  value="6" /><br />

    Slider: <input type="range"
                  min="0"
                  max="10"
                  step="2"
                  value="6" /><br />

    Date: <input type="date" size="10"/><br />

    DateTime: <input type="datetime" /><br />

    Search: <input type="search" /><br />

    Search with Placeholder:
        <input type="search"
                  placeholder="Type your search here"/><br />

    Autofocus field: <input type="text" autofocus />
</form>
```



Latest release:

Version 1.1 fills in some of the gaps by testing for more exciting HTML5 features like localStorage, Web Workers, applicationCache and more...

[Read the full release notes](#)

1.1

What is Modernizr?

Have you ever wanted to do if-statements in your CSS for the availability of cool features like `border-radius`? Well, with **Modernizr** you can accomplish just that! The syntax is very intuitive, too:

```
.multiplebgs div p {  
  /* properties for browsers that
```

Home

News

Docs

Sites

Modernizr detects support for:

@font-face ✓	CSS Columns ✓
Canvas ✓	CSS Gradients ✓
Canvas Text ✓	CSS Reflections ✓
HTML5 Audio ✓	CSS 2D Transforms ✓
HTML5 Video ✓	CSS 3D Transforms ✓
rgba() ✓	CSS Transitions ✓
hsia() ✓	Geolocation API
border-image: ✓	localStorage ✓
border-radius: ✓	sessionStorage ✓
box-shadow: ✓	Web Workers ✓
Multiple backgrounds ✓	applicationCache ✓
opacity: ✓	Input Types†
CSS Animations ✓	Input Attributes‡

A green checkmark behind a feature indicates your current browser supports this.

CSS CLASSES: (Classes for input types are not applied)

JAVASCRIPT PROPERTY: **Modernizr.inputtypes[type]**

HTML5 introduces [thirteen new values](#) for the <input>'s type attribute. They are as follows: `search`, `tel`, `url`, `email`, `datetime`, `date`, `month`, `week`, `time`, `datetime-local`, `number`, `range`, `color`.

These types can enable native datepickers, colorpickers, URL validation, and so on. If a browser doesn't support a given type, it will be rendered as a text field.

Sample Usage:

```
<!-- In your HTML: -->
<input type="date" name="birthday" id="birthday">

// In your JavaScript:
if (!Modernizr.inputtypes.date){
    // if no native support, use a datepicker script
    createDatepicker(document.getElementById('birthday'));
}
```

NEW!
Developer Tools
 Send a url and get detailed reports

FindMeByIP.com

Web Design Checklist

Get a FREE Account

Sign In

your IP address

FindmebyIP

.com

User support revealed

{75.243.39.120}

Browser

Modernizer Support

HTML5 Forms 2.0

CSS3 Selector Tests

IP Address & GeoLocation

Browser Information

A "Browser" is a piece of software used to access & display the Internet on your computer. Examples include Internet Explorer, Mozilla Firefox, Opera, Safari and Google Chrome.

Information about your browser and the operating system it runs on can be found in the table to the right.

Browser



Apple Safari

Browser Version

4.0

Operating System



Apple OS X

User Agent String

Mozilla/5.0 (Macintosh; U; Intel Mac OS X 10_5_8; en-us)
 AppleWebKit/531.21.8 (KHTML, like Gecko) Version/4.0.4
 Safari/531.21.10

NEW!
Developer Tools

Send a url and get detailed reports
about browser support for Address & GeoLocation

HTML5 Web Forms 2.0

The first Web Forms 2.0 Draft appeared as far back as February 2004. Now superseded by the [Forms chapter](#) of the HTML5 specification it introduces new elements which offer a wide range of functionality previously only possible with the help of third party scripts.

Your browser's support for basic Web Forms 2.0 is show in the list to the right.

CSS3 Tests

The CSS3 spec defines a series of extremely powerful selectors, many of which can be used today in modern browsers.

Modernizer Support

HTML5 Forms 2.0

CSS3 Selector Tests

✓ <input type="search">

✓ <input type="tel">

✓ <input type="url">

✓ <input type="email">

✗ <input type="datetime">

✗ <input type="date">

✗ <input type="month">

✗ <input type="week">

✗ <input type="time">

✗ <input type="datetime-local">

✓ <input type="number">

✓ <input type="range">

✗ <input type="color">

✓ E[att^="val"]

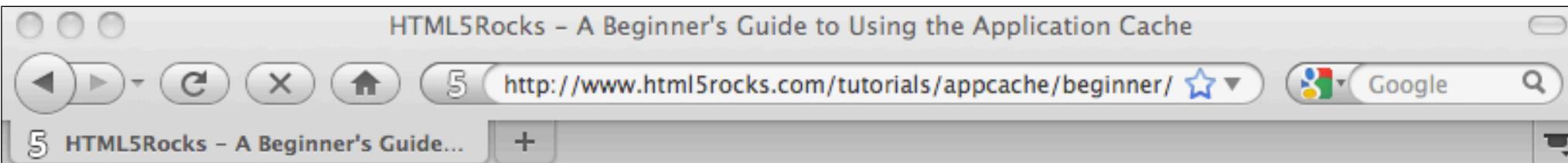
✓ E[att\$="val"]

✓ E[att*="val"]

✓ E:root

<mobile support />





Presentation Playground Studio **Tutorials** Resources

[Home](#) > [Tutorials](#) > A Beginner's Guide to Using the Application Cache

A Beginner's Guide to Using the Application Cache

Eric Bidelman, June 18, 2010

Supported browsers:



70



Your browser appears to support all of the functionality used in this sample.

- [Introduction](#)
- [The cache manifest file](#)
 - [Referencing a manifest file](#)
 - [Structure of a manifest file](#)
- [Updating the cache](#)
- [References](#)

Transferring data from www.google.com...



Introduction

It's becoming increasingly important for web-based applications to be accessible offline. Yes, all browsers have caching mechanisms, but they're unreliable and don't always work as you might expect. HTML5 addresses some of the annoyances of being offline with the [ApplicationCache](#) interface.

Using the cache interface gives your application three advantages:

1. Offline browsing - users can navigate your full site when they're offline
2. Speed - cached resources are local, and therefore load faster.
3. Reduced server load - the browser will only download resources from the server that have changed.

The Application Cache (or AppCache) allows a developer to specify which files the browser should cache and make available to offline users. Your app will load and work correctly, even if the user presses the refresh button while they're offline.

The cache manifest file

The cache manifest file is a simple text file that lists the resources the browser should cache for offline access.

Referencing a manifest file

To enable the application cache for an app, include the manifest attribute on the document's `html` tag:

```
<html manifest="example.manifest">  
  ...  
</html>
```



5 HTML5Rocks – A Beginner's Guide...



Note: Sites are limited to 5MB worth of cached data.

Note: If the manifest file or a resource specified in it fails to download, the entire cache update process fails. The browser will keep using the old application cache in the event of failure.

Lets take a look at a more complex example:

```
CACHE MANIFEST
# 2010-06-18:v2

# Explicitly cached entries
CACHE:
index.html
stylesheet.css
images/logo.png
scripts/main.js

# Resources that require the user to be online.
NETWORK:
login.php
/myapi
http://api.twitter.com

# static.html will be served if main.py is inaccessible
# offline.jpg will be served in place of all images in images/large/
FALLBACK:
/main.py /static.html
images/large/ images/offline.jpg
```



5 HTML5Rocks – A Beginner's Guide...

+

Note: These sections can be listed in any order and each section can appear more than one in a single manifest.

The following .manifest defines a "catch-all" page (`offline.html`) that will be displayed when the user tries to access the root of the site while offline. It also declares that all other resources (e.g. those on remote a site) require an internet connection.

```
CACHE MANIFEST
# 2010-06-18:v3

# Explicitly cached entries
index.html
css/style.css

# offline.html will be displayed if the user is offline
FALLBACK:
/ /offline.html

# All other resources (e.g. sites) require the user to be online.
NETWORK:
*

# Additional resources to cache
CACHE:
images/logo1.png
images/logo2.png
images/logo3.png
```

A screenshot of a web browser window. The title bar says "HTML5Rocks – A Beginner's Guide to Using the Application Cache". The address bar shows the URL "http://www.html5rocks.com/tutorials/appcache/beginner/". Below the address bar, there are five tabs open, with the first tab labeled "HTML5Rocks – A Beginner's Guide...".

5 HTM... +

```
var appCache = window.applicationCache;

switch (appCache.status) {
  case appCacheUNCACHED: // UNCACHED == 0
    return 'UNCACHED';
    break;
  case appCache.IDLE: // IDLE == 1
    return 'IDLE';
    break;
  case appCache.CHECKING: // CHECKING == 2
    return 'CHECKING';
    break;
  case appCache.DOWNLOADING: // DOWNLOADING == 3
    return 'DOWNLOADING';
    break;
  case appCache.UPDATEREADY: // UPDATEREADY == 5
    return 'UPDATEREADY';
    break;
  case appCache.OBSOLETE: // OBSOLETE == 5
    return 'OBSOLETE';
    break;
  default:
    return 'UNKNOWN CACHE STATUS';
}
};
```

5 HTML5Rocks – A Beginner's Guide...

+

To programmatically update the cache, first call `applicationCache.update()`. This will attempt to update the user's cache (which requires the manifest file to have changed). Finally, when the `applicationCache.status` is in its `UPDATEREADY` state, calling `applicationCache.swapCache()` will swap the old cache for the new one.

```
var appCache = window.applicationCache;  
  
appCache.update(); // Attempt to update the user's cache.  
  
...  
  
if (appCache.status == window.applicationCache.UPDATEREADY) {  
    appCache.swapCache(); // The fetch was successful, swap in the new cache.  
}
```

As you may expect, many events are exposed to monitor these states. The cache interface fires events for things like download progress, updating the app cache, and error conditions. The following snippet sets up event listeners for each type of cache event:

```
function handleCacheEvent(e) {  
    //...  
}  
  
function handleCacheError(e) {  
    alert('Error: Cache failed to update!');  
};
```



Nº 7.

THE PAST, PRESENT & FUTURE OF LOCAL STORAGE FOR WEB APPLICATIONS

Which browsers? Well, the latest version of pretty much every browser supports HTML5 Storage... even Internet Explorer!

HTML5 STORAGE SUPPORT

IE	FIREFOX	SAFARI	CHROME	OPERA	IPHONE	ANDROID
8.0+	3.5+	4.0+	4.0+	10.5+	2.0+	2.0+

USING HTML5 STORAGE

HTML5 Storage is based on named key/value pairs. You store data based on a named key, then you can retrieve that data with the same key. The named key is a string. The data can be any type supported by JavaScript, including strings, Booleans, integers, or floats. However, the data is actually stored as a string. If you are storing and retrieving anything other than strings, you will need to use functions like `parseInt()` or `parseFloat()` to coerce your retrieved data into the expected JavaScript datatype.

```
interface Storage {  
    getter any getItem(in DOMString key);  
    setter creator void.setItem(in DOMString key, in any data);  
};
```

Calling `setItem()` with a named key that already exists will silently overwrite the previous value. Calling `getItem()` with a non-existent key will return `null` rather than throw an exception.

5 Local Storage – Dive Into HTML5



How does it work? Every time a change occurs within the game, we call this function:

```
function saveGameState() {  
    if (!supportsLocalStorage()) { return false; }  
    localStorage["halma.game.in.progress"] = gGameInProgress;  
    for (var i = 0; i < kNumPieces; i++) {  
        localStorage["halma.piece." + i + ".row"] = gPieces[i].row;  
        localStorage["halma.piece." + i + ".column"] = gPieces[i].column;  
    }  
    localStorage["halma.selectedpiece"] = gSelectedPieceIndex;  
    localStorage["halma.selectedpiecehasmoved"] = gSelectedPieceHasMoved;  
    localStorage["halma.movecount"] = gMoveCount;  
    return true;  
}
```

As you can see, it uses the `localStorage` object to save whether there is a game in progress (`gGameInProgress`, a Boolean). If so, it iterates through the pieces (`gPieces`, a JavaScript Array) and saves the row and column number

5 Local Storage – Dive Into HTML5

+

The Web SQL Database specification has been implemented by four browsers and platforms.

WEB SQL DATABASE SUPPORT

IE	FIREFOX	SAFARI	CHROME	OPERA	IPHONE	ANDROID
.	.	4.0+	4.0+	10.5+	3.0+	2.0+

Of course, if you've used more than one database product in your life, you are aware that "SQL" is more of a marketing term than a hard-and-fast standard. (Some would say the same of "HTML5," but never mind that.) Sure, there is an actual SQL specification (it's called [SQL-92](#)), but there is no database server in the world that conforms to that and only that specification. There's Oracle's SQL, Microsoft's SQL, MySQL's SQL, PostgreSQL's SQL, and SQLite's SQL. Indeed, each of these products adds new SQL features over time, so even saying "SQLite's SQL" is not sufficient to pin down exactly what you're talking about. You need to say "the version of SQL that shipped with SQLite version X.Y.Z."

All of which brings us to the following disclaimer, currently residing at the top of the Web SQL Database specification:

<video />

Nº 5. VIDEO ON THE WEB



nyone who has visited YouTube.com in the past four years knows that you can embed video in a web page. But prior to HTML5, there was no standards-based way to do this. Virtually all the video you've ever watched "on the web" has been funneled through a third-party plugin — maybe QuickTime, maybe RealPlayer, maybe Flash. (YouTube uses Flash.) These plugins integrate with your browser well enough that you may not even be aware that you're using them. That is, until you try to watch a video on a platform that doesn't support that plugin.

HTML5 defines a standard way to embed video in a web page, using a `<video>` element. Support for the `<video>` element is still evolving, which is a polite way of saying it doesn't work yet. At least, it doesn't work everywhere. But don't despair! There are alternatives and fallbacks and options galore.

As of this writing, this is the landscape of HTML5 video:

- Mozilla Firefox (3.5 and later) supports Theora video and Vorbis audio in an Ogg container.
- Opera (10.5 and later) supports Theora video and Vorbis audio in an Ogg container.
- Google Chrome (3.0 and later) supports Theora video and Vorbis audio in an Ogg container. It also supports H.264 video (all profiles) and AAC audio (all profiles) in an MP4 container.
- Safari on Macs and Windows PCs (3.0 and later) will support anything that QuickTime supports. In theory, you could require your users to install third-party QuickTime plugins. In practice, very few users are going to do that. So you're left with the formats that QuickTime supports "out of the box." This is a long list, but it does not include Theora video, Vorbis audio, or the Ogg container. However, QuickTime DOES support H.264 video (main profile) and AAC audio in an MP4 container.
- Mobile phones like Apple's iPhone and Google Android phones support H.264 video (baseline profile) and AAC audio ("low complexity" profile) in an MP4 container.

```
<video src="NewOrleans2006.ogv"  
       width="320"  
       height="240"></video>
```

```
<video width="320" height="240" controls autoplay>  
  <source src="NewOrleans2006.ogv" type='video/ogg; codecs="theora, vorbis"'>  
  <source src="NewOrleans2006.mp4" type='video/mp4; codecs="avc1.42E01E, mp4a.40.2"'>  
</video>
```

5 Video on the Web – Dive Into HTM...



The final markup uses a `<video>` element for HTML5 video, a nested `<object>` element for Flash fallback, and a small bit of script for the benefit of Android devices:

```
<video id="movie" width="320" height="240" preload controls>
  <source src="pr6.mp4" />
  <source src="pr6.webm" type='video/webm; codecs="vp8, vorbis"' />
  <source src="pr6.ogv" type='video/ogg; codecs="theora, vorbis"' />
  <object width="320" height="240" type="application/x-shockwave-flash"
    data="flowplayer-3.2.1.swf">
    <param name="movie" value="flowplayer-3.2.1.swf" />
    <param name="allowfullscreen" value="true" />
    <param name="flashvars" value='config={"clip": {"url": "http://wearehugh.com/dih5/good
/bbb_480p.mp4", "autoPlay":false, "autoBuffering":true}}' />
    <p>Download video as <a href="pr6.mp4">MP4</a>, <a href="pr6.webm">WebM</a>, or <a
      href="pr6.ogv">Ogg</a>.</p>
  </object>
</video>
```



Search

Browse

Upload

YouTube HTML5 Video Player

This is an opt-in experiment for HTML5 support on YouTube. If you are using a supported browser, you can choose to use the HTML5 player for videos. Your comments will help us improve and perfect the mixtures we're working on. So jump in, play around, and send your feedback down below!

Supported Browsers

Right now we support browsers that support both the video tag in HTML5 and the h.264 video codec. These include:

- Google Chrome
- Apple Safari (version 4+)
- Microsoft Internet Explorer with Google Chrome Frame installed ([Get Google Chrome Frame](#))

Updates!

- 1/27/2010: Fullscreen support enabled (if supported by browser).

Additional Restrictions (we are working on these!)

- Videos with ads are not supported (they will play in the Flash player)
- Fullscreen is not supported
- If you've opted in to other testtube experiments, you may not get the HTML5 player (Feather is supported, though)

You are not currently in the HTML5 beta.

[Join the HTML5 Beta](#)

[Help](#) [About](#) [Safety](#) [Privacy](#) [Terms](#) [Copyright](#) [Partners](#) [Developers](#) [Advertising](#)

Language: [English](#) Location: [Worldwide](#) Safety mode: [Off](#)



Apple proposes HTTP streaming f...

+

Apple proposes HTTP streaming feature as IETF standard

By Chris Foresman | Last updated July 9, 2009 11:00 PM

When Apple discussed the new features of the forthcoming iPhone OS 3.0, SVP of iPhone Software Engineering Scott Forstall said that the iPhone would be capable of **streaming video and audio directly over HTTP**. Apple also advertised HTTP streaming as a **feature of QuickTime X**, the update of its media architecture coming in Snow Leopard. What it failed to explain, at least publicly, is how this streaming would be accomplished. Fortunately, Apple **submitted its proposed protocol** last month to the Internet Engineering Task Force (IETF) in the hopes that it will become a ubiquitous standard.

Apple identified what it considers a few issues with standard streaming, which generally uses the **Real Time Streaming Protocol** originally developed by Netscape and Real in the late '90s. The biggest issue with RTSP is that the protocol or its necessary ports may be

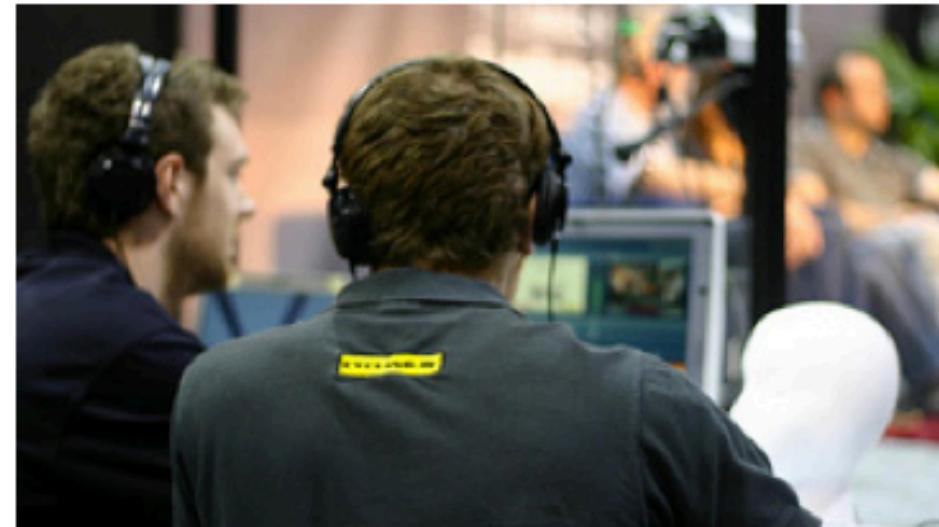


Photo CC Richard Masoner

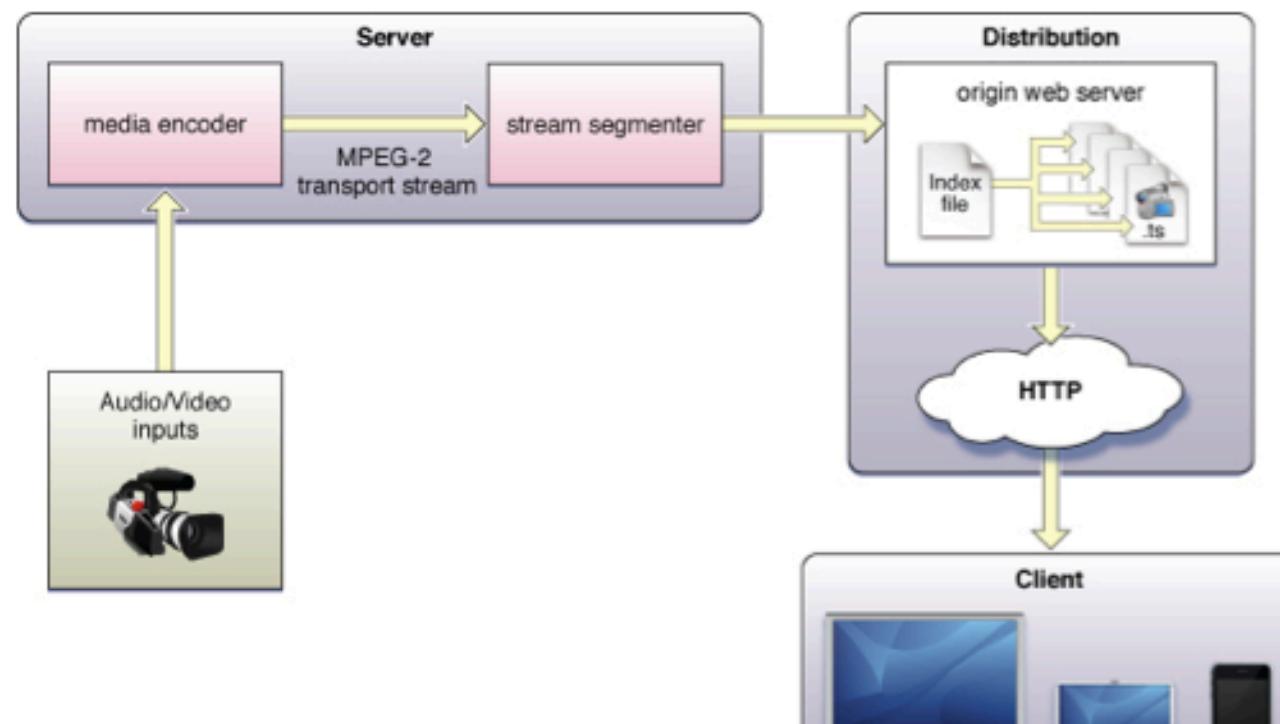
HTTP Live Streaming Overview

Table of Contents

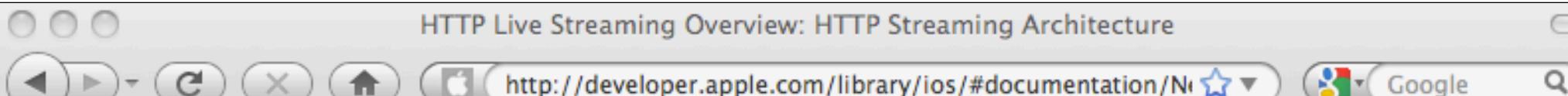
- Introduction
- HTTP Streaming Architecture
- Using HTTP Live Streaming
- Frequently Asked Questions
- Revision History

An example of a simple HTTP streaming configuration is shown in "A basic configuration."

Figure 1–1 A basic configuration



HTTP Live Streaming Overview: HTTP Streaming Architecture



iOS Reference Library

Developer



HTTP Live Streaming Overview

Table of Contents

- Introduction
- HTTP Streaming Architecture
- Using HTTP Live Streaming
- Frequently Asked Questions
- Revision History

Media segments are saved as .ts files (MPEG-2 streams) and index files are saved as .M3U8 files, an extension of the .m3u format used for MP3 playlists.

Note: Because the index file format is an extension of the .m3u file format, and because the system also supports .mp3 audio media files, the client software may also be compatible with typical MP3 playlists used for streaming Internet radio.

Here is a very simple example of an .M3U8 file a segmenter might produce if the entire stream were contained in three unencrypted 10-second media files:

```
#EXTM3U
#EXT-X-MEDIA-SEQUENCE:0
#EXT-X-TARGETDURATION:10
#EXTINF:10,
http://media.example.com/segment1.ts
#EXTINF:10,
http://media.example.com/segment2.ts
#EXTINF:10,
http://media.example.com/segment3.ts
```

Apple to Provide Live Video Strea...

Media Alert

Apple to Provide Live Video Streaming of September 1 Event

What:

Live video stream of Apple's September 1 event

When:

Wednesday, September 1, 2010, 10:00 a.m. PDT

Where:

www.apple.com

Live Video Streaming

Apple® will broadcast its September 1 event online using Apple's industry-leading HTTP Live Streaming, which is based on open standards. Viewing requires either a Mac® running Safari® on Mac OS® X version 10.6 Snow Leopard®, an iPhone® or iPod touch® running iOS 3.0 or higher, or an iPad™. The live broadcast will begin at 10:00 a.m. PDT on September 1, 2010 at www.apple.com.

HTML5 Demos and Examples

http://html5demos.com/ Google

Search ... google ... An intr... differe... Video o... http://l... Demos ... > +

Fork me on GitHub

HTML 5 Demos and Examples

HTML 5 experimentation and demos I've hacked together. Click on the browser support icon or the technology tag to filter the demos (the filter is an OR filter).

Demo	Support	Technology
Two videos playing in sync		video
Interactive canvas gradients		canvas
Canvas & Video		video canvas
Video		video
Canvas		canvas
Content Editable		contenteditable storage
Geolocation Works on Safari Mobile too		geolocation

Contacts

Events

Locations

Tagspaces

Bookmarks

Resources

Options

git Html5 cross browser polyfills – M...

+



scottdavis99 1

Dashboard

Inbox 0

A

Explore GitHub

Gist

Blog

Help



Modemizr / Modernizr

Watch

[Source](#) [Commits](#) [Network \(49\)](#) [Issues \(33\)](#) [Wiki \(2\)](#) [Graphs](#)[Home](#) [Pages](#) [Wiki History](#) [Git Access](#)

HTML5 Cross browser Polyfills

[Edit](#)

So here we're collecting all the shims, fallbacks, and polyfills in order to implant html5 functionality in browsers that don't natively support it.

The general idea is that: we, as developers, should be able to develop with the HTML5 apis, and scripts can create the methods that should exist. Developing in this future-proof way means as users upgrade, your code doesn't have to change but users will make the experience cleanly.

Looking for a way to conditionally load these scripts client-side based on feature detects? See [yepnope.js](#)



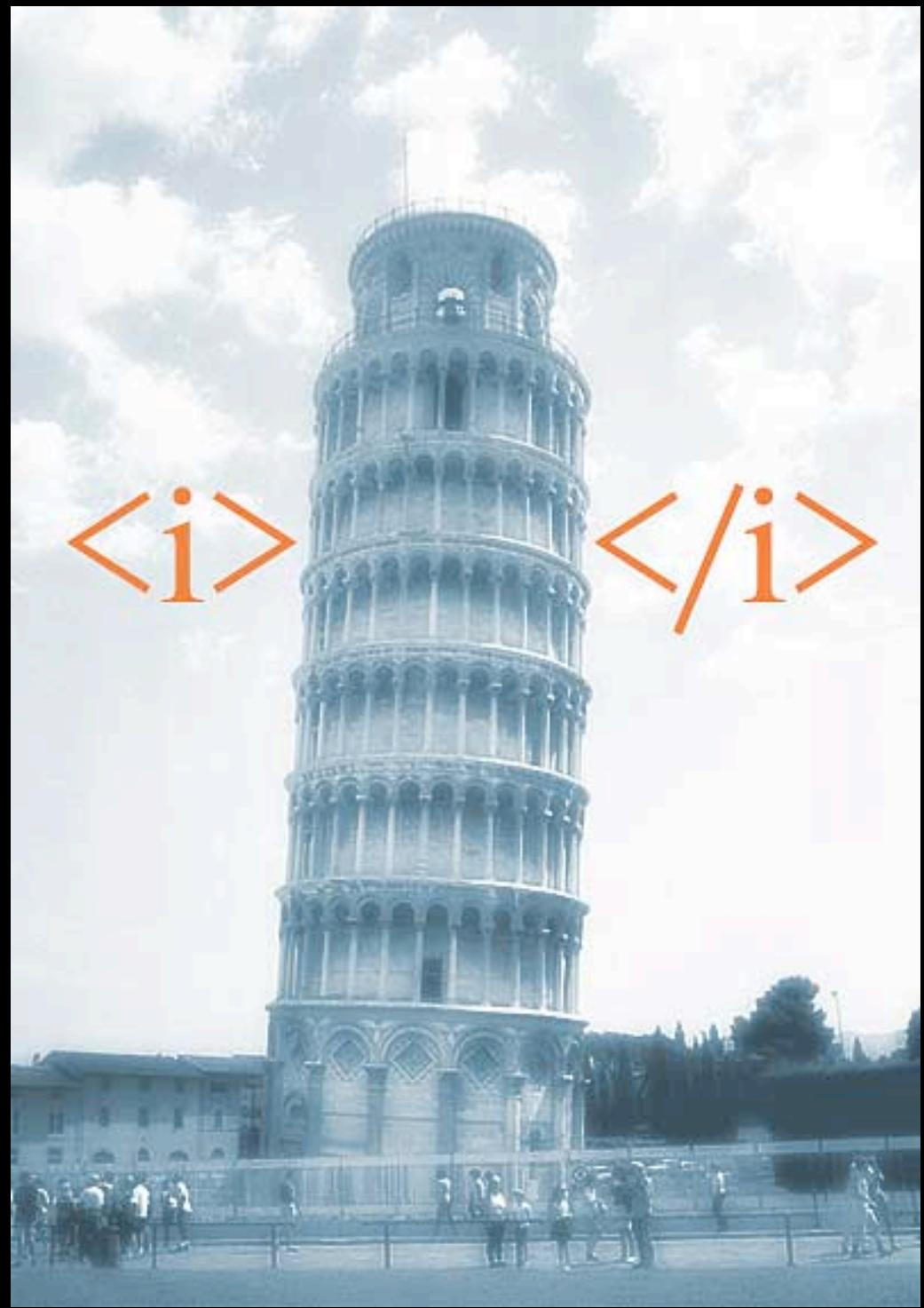
New elements

Form enhancements

Mobile Support

Video

*Semantics
over
Markup*





ThirstyHead.com
training done right.



Scott Davis
scott@thirstyhead.com

Questions?
Thanks for your time.



Tomorrow's Tech Today: HTML5

Scott Davis, ThirstyHead.com

Image Credits

<http://www.flickr.com/photos/rutty/503238148/>

<http://www.flickr.com/photos/daniello/422213306/>

<http://www.flickr.com/photos/f-l-e-x/3096005116/>

<http://www.flickr.com/photos/photohannah/512202109/>

<http://www.flickr.com/photos/jesper/346483297/>

<http://www.flickr.com/photos/stinajonsson/3932774410/>