HTML5
What is it and why should I care?

By Peter Lubbers,
Kaazing
Just Who Is Peter Lubbers?

• Director of Documentation and Training, Kaazing (an HTML5 WebSocket company, first to provide global HTML5 training)  
  www.kaazing.com (yes, we’re hiring)

• Co-Founder San Francisco HTML5 User Group  
  http://www.sfhtml5.org/

• Co-author Apress book  
  Pro HTML5 Programming  
  (Alpha Release now available)

• Ultra Distance Runner

• Twitter: @peterlubbers
Web2Open Special!

10% Off Any HTML5 Training
With Coupon Code WEB2OPEN
http://tech.kaazing.com/training/index.html
Agenda

• What is HTML5 (and why should I care)?
  o Brief History
  o HTML5 Standards Bodies and Specifications
  o The HTML5 Vision
  o HTML5 Features

• Q&A

• Resources
So, Why Should I Care?
What do Google, Apple, and Microsoft All Agree on?
Verbatim

“The world is moving to HTML5.”
—Steve Jobs, Apple

“The Web has not seen this level of transformation, this level of acceleration, in the past ten years... we're betting big on HTML5.”
—Vic Gundotra, VP of Engineering, Google

“In a nutshell, we love HTML5, we love it so much we want it to actually work.
—Dean Hachamovitch, General Manager for Internet Explorer, Microsoft

“I had no idea there was so much HTML5 already in play.”
—Tim O’Reilly
Another Reason?

• The iPad
  o One million units sold in 28 Days
  o No Flash
What is HTML5?
HTML Timeline
HTML Timeline < 1980
# HTML Timeline 1980-1990

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Version</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Event</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Tim Berners-Lee proposes an Internet-based hypertext system</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Tim Berners-Lee writes a memo proposing an Internet-based hypertext system</td>
<td></td>
</tr>
<tr>
<td><strong>Other Events</strong></td>
<td>IBM PC, MS DOS</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
# HTML Timeline 1991-2000

<table>
<thead>
<tr>
<th>Year</th>
<th>Event</th>
</tr>
</thead>
<tbody>
<tr>
<td>1991</td>
<td>HTML Tags first mentioned in public</td>
</tr>
<tr>
<td>1992</td>
<td>Hypertext Markup Language published by IETF as an Internet-Draft</td>
</tr>
<tr>
<td>1993</td>
<td>HTML 2.0 released by IETF, HTML3 proposed and abandoned</td>
</tr>
<tr>
<td>1994</td>
<td>HTML 2.0 released by IETF, HTML3 proposed and abandoned</td>
</tr>
<tr>
<td>1995</td>
<td>HTML 2.0 released by IETF, HTML3 proposed and abandoned</td>
</tr>
<tr>
<td>1996</td>
<td>HTML 3.2 released by W3C (January), HTML 4.0 released (December)</td>
</tr>
<tr>
<td>1997</td>
<td>HTML 3.2 released by W3C (January), HTML 4.0 released (December)</td>
</tr>
<tr>
<td>1998</td>
<td>HTML 4.01 published as a W3C Recommendation</td>
</tr>
<tr>
<td>1999</td>
<td>HTML 4.01 published as a W3C Recommendation</td>
</tr>
<tr>
<td>2000</td>
<td>XHTML 1.0 released</td>
</tr>
</tbody>
</table>
# HTML Timeline 2001-2011

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Last errata of HTML 4.01 published, XHTML 1.1 published</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>WHATWG starts work on HTML5 under the name Web Applications 1.0</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Web 2.0</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>W3C—Wait!</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>W3C announces that the XHTML 2 group will stop work by end of 2009</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>iPod v1</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>iPad Nano v5</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Today!</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Copyright © 2010 - Kaazing Corporation. All rights reserved.
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>HTML5 Candidate Recommendation</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>HTML5 Proposed Recommendation</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Standards Bodies and Specifications
HTML5 Standards Bodies

- Web Hypertext Application Technology Working Group (WHATWG)
  - Founded in 2004 (by individuals working for browser vendors Apple, Mozilla, Google, and Opera)
  - Develops HTML and APIs for web application development
  - Unofficial and open collaboration of browser vendors and interested parties

- World Wide Web Consortium (W3C)
  - Created draft of HTML5 in 2008

- Internet Engineering Task Force (IETF)
  - Protocols (HTTP, WebSocket)
Specifications

• WHATWG
  o http://www.whatwg.org/specs/web-apps/current-work/

• W3C
  o http://dev.w3.org/html5/spec/Overview.html

• IETF (Protocol specifications)
Is This HTML5?

• Many pieces of the HTML5 effort (for example, Local Storage, WebSocket, and Geolocation) were originally part of the HTML5 specification

• Moved to a separate standards document to keep the specification focused

• Industry still refers to the original set of features, including Web Sockets, as "HTML5"

• See: http://www.whatwg.org/specs/web-apps/current-work/multipage/introduction.html#is-this-html5?
HTML5 Vision and Design Principals
Design Principal #1 Compatibility

- Pave the cow paths
- Degrade gracefully
- Research common behavior; solve real problems
- Support existing content
- Evolution not revolution
- Don’t reinvent the wheel (or at least make a better one!)
Design Principal #2
Utility

• Priority of Constituencies:
  o When in doubt… value users over authors, over implementers (browsers), over specifiers (W3C/WHATWG), over theoretical purity.

• Secure by design
• DOM consistency (HTML5 and XHTML5)
• Separation of presentation and content
Presentation vs. Content

• HTML5 strives to separate content from presentation where possible (use CSS)
• Most of the presentational features from earlier versions of HTML are no longer supported
• This was already in the works (HTML4 Transitional, XHTML1.1)
• Problems with presentational markup:
  o Poor accessibility
  o Unnecessary complexity
  o Larger document size
Design Principal #3
Interoperability

• Specify well-defined behavior
  o Specific instead of vague

• Handle errors well
  o Improved and ambitious error handling plans
  o Prefer graceful error recovery to hard failure

• Avoid needless complexity
  o Simple is better
Simple is Better

• Simplify wherever possible
• Examples:
  o Native browser ability instead of complex JavaScript code
  o New doctype
  o Character set
  o HTML5 APIs
Simplified Doctype

HTML4/XHTML
- HTML 4.01 Strict
- HTML 4.01 Transitional
- HTML 4.01 Frameset
- XHTML 1.0 Strict
- XHTML 1.0 Transitional
- XHTML 1.0 Frameset
- XHTML 1.1

<!DOCTYPE HTML PUBLIC "--//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">

<!DOCTYPE html>

HTML5
<!DOCTYPE html>
Simplified Character Set

HTML4

<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">

HTML5

<meta charset="utf-8">
More Detail (Where Needed)

• HTML5 is more detailed than previous specs to avoid misinterpretation
• Aim to define things thoroughly, especially web applications
• ~900 Pages spread over multiple specifications
Design Principal #4
Universal Access

• Support for all world languages
  o For example, `<ruby>`
    (Ruby annotations, used in East Asian typography)

• Accessibility
  o Support users with disabilities
  o Web Accessibility Initiative (WAI) Accessible Rich Internet Applications (ARIA)
  o WAI-ARIA roles can be added today
  o Supported by screen readers
Plugin-Free Paradigm

- HTML5 provides *native* support for many features that were only possible with plugins or complex hacks
  - Plugins may not be installed
  - Plugins can be disabled or blocked (iPad does not ship with Flash plugin)
  - Plugins are a separate attack vector
  - Plugins are difficult to integrate with the rest of an HTML document (plugin boundaries, clipping, and transparency issues)
HTML5 Features
HTML5 Features

- HTML5 Markup
  - New and removed HTML elements
  - New content types
- HTML5 Forms
- HTML5 APIs
- Other HTML5 Features
HTML5 Markup
New HTML5 Tags

<article> <aside> <audio> <canvas>
<command> <datalist> <details> <dialog>
<embed> <figure> <footer> <header> <hgroup>
<keygen> <mark> <meter> <nav>
<output> <ruby> <rt> <rp> <section>
<source> <video>
Removed Tags in HTML5

- <acronym>
- <applet>
- <basefont>
- <big>
- <center>
- <frame>
- <frameset>
- <noframes>
- <dir>
- <font>
- <s> <strike>
- <u>
- <tt>
**HTML5 Cheat Sheet**

By Antonio Lupetti

Anatomy of an HTML5 Page
Anatomy of an HTML5 Page

HTML code (head section)

<!DOCTYPE html>
<html>
    <head>
        <meta charset="utf-8" />
        <title>HTML 5</title>
        <link rel="stylesheet" href="html5.css">
    </head>
</html>
Anatomy of an HTML5 Page

HTML code body section (part 1)

```html
<body>
  <header>
    <h1>Header</h1>
  </header>

  <div id="container">
    <nav>
      <h3>Nav</h3>
      <h3>Nav</h3>
    </nav>
  </div>
</body>
```
Anatomy of an HTML5 Page

HTML code

```
<section>
  <article>
    <header>
      <h1>Article Header</h1>
    </header>
    <footer>
      <h2>Article Footer</h2>
    </footer>
  </article>
</section>
```
Anatomy of an HTML5 Page

HTML code body section (part 3)

```html
<aside>
    <h3>Aside</h3>
</aside>
<footer>
    <h2>Footer</h2>
</footer>
</div>
</body>
</html>
```
Anatomy of an HTML5 Page

Header
Subtitle
HTML5 Rocks!

Nav

Article Header
Lorem ipsum dolor HTML5 scelerisque elit, consectetur adipiscing elit. Vivamus at est enim. Ut fringilla nunc.
Pellentesque odio diam, chamuscus nunc congue in. Class aptent taciti sociosqu ad litora torquent per conubia nostra.

Perceptus hammenus. Quisque feugiat, justo at vehicula pellentesque, turpis lorem dictum nunc.

Article Footer

Article Header
Lorem ipsum dolor sit amet, HTML5 scelerisque elit, consectetur adipiscing elit. Vivamus at est enim. Ut fringilla nunc.
Pellentesque odio diam, chamuscus nunc congue in. Class aptent taciti sociosqu ad litora torquent per conubia nostra.

Perceptus hammenus. Quisque feugiat, justo at vehicula pellentesque, turpis lorem dictum nunc.

Article Footer

Aside

HTML5: "Lorem ipsum dolor sit amet, consectetur adipiscing elit. Vivamus at est enim. Ut fringilla nunc. Pellentesque odio diam, chamuscus nunc congue in. Pellentesque odio diam, chamuscus nunc congue in. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per perceptus hammenus. Quisque feugiat, justo at vehicula pellentesque, turpis lorem dictum nunc."

Footer

#5 No CSS
Anatomy of an HTML5 Page

#6 With CSS
Anatomy of an HTML5 Page

#7 Explained
HTML5 and CSS Level 3

• CSS level 2 (W3C)
  o published as a Recommendation 1998
  o Candidate Recommendation 2007

• CSS Level 3 (W3C)
  o Like HTML5, under development
  o Modularized with sub-specifications for easier browser uptake (transformation, animation, transition)

• Some CSS Level 3 examples:
  o http://www.zurb.com/playground/
  o http://www.zachstronaut.com/
HTML5 Forms
HTML5 Forms

- Formerly called Web Forms 2.0
- Native functionality (no scripting for validation) means less coding, for example:

Regular expression for a valid e-mail address:

```regex
(?:[a-z0-9!#$%&'\*+/=?^`{|}~\-]+(?:\.|\[\x01-\x08\x0b\x0c\x0e-\x1f\x21\x23-\x5b\x5d-\x7f]\]|\[(?:[\x01-\x09\x0b\x0c\x0e-\x1f\x21\x23-\x5b\x5d-\x7f]\]|\[(?:\[\x01-\x08\x0b\x0c\x0e-\x1f\x21\x23-\x5b\x5d-\x7f]\])\])*\@(?:(?:[a-z0-9]+(?:[a-z0-9-]*[a-z0-9])?\.)+[a-z0-9]+(?:[a-z0-9-]*[a-z0-9])?\.)\{3\}(?:[\x01-\x08\x0b\x0c\x0e-\x1f\x21-\x5a\x53-\x7f]\]|\[(?:[\x01-\x09\x0b\x0c\x0e-\x1f\x21-\x5b\x5d-\x7f]\)])
```
HTML5 Forms

• New form functionality:
  o Date and color pickers
  o Search, e-mail, web address input types
  o Validation
  o Spin boxes and Sliders
  o More

• Backward compatible

• Features degrade gracefully (unknown input types are treated as text-type) input
Example HTML5 Forms

HTML 5 Forms Demo

Please note, these HTML 5 extensions are currently only supported in Opera 9.5+.

A word on validation. There is a bug in the HTML 5 validator that protests at my use of attr.
I'm telling it to update everytime the slider input is changed, via onforminput which

Fields with red borders are required.

<table>
<thead>
<tr>
<th>Name (required)</th>
<th>Peter</th>
</tr>
</thead>
<tbody>
<tr>
<td>Title</td>
<td>Mr</td>
</tr>
<tr>
<td>Shoesize</td>
<td>18</td>
</tr>
<tr>
<td>Email (required)</td>
<td><a href="mailto:peter.lubbers@kaazing.com">peter.lubbers@kaazing.com</a></td>
</tr>
<tr>
<td>Webpage</td>
<td><a href="http://tech.kaazing.com">http://tech.kaazing.com</a></td>
</tr>
<tr>
<td>Date of Birth</td>
<td>July 1970</td>
</tr>
<tr>
<td>Attractiveness</td>
<td></td>
</tr>
</tbody>
</table>
Browser Support for Forms

- Opera 9.6+
- Chrome (Partial)
- Safari (Partial)
HTML5 APIs
HTML5 APIs

• Audio and Video
• Canvas and SVG
• WebSocket
• Server-Sent Events
• Communication
  o Cross Document Messaging
  o XMLHttpRequest Level 2
• Web Workers
• Geolocation
• Web Storage
HTML5 Audio and Video
Audio and Video

• New HTML5 media elements
  o `<audio>`
  o `<video>`

• Control using Audio/Video API

• Native support in the browser (embedded codecs)

• Originally, the HTML5 specification was going to require codec support…
AV Containers and Codecs

- Audio and Video containers:
  - H264 and Ogg

- Audio and video codecs (algorithms used to encode and decode an audio or video stream):
  - Audio: AAC, MP3, Vorbis
  - Video: H264, MP4, Theora

- Stalemate over patented/free codec inclusion (Ogg vs. H264)

- You can add multiple formats per (audio/video) element to fallback
Example:
YouTube HTML5 Channel

http://www.youtube.com/user/rocpoc
Browser Support for Audio and Video

- Chrome 3.0+ (Ogg + H264)
- Firefox 3.5+ (Ogg)
- Safari 3.2 (H264)
- Opera 10.5 (Ogg)
HTML5 Canvas & SVG
About Canvas and SVG

• Provide native drawing functionality on the Web—Basically a scriptable *bitmap* canvas (images that are drawn are final and cannot be resized)

• Previously possible with Flash, VML, Silverlight and very complex to do in JavaScript

• Completely integrated into HTML5 documents (part of DOM)

• Can be styled with CSS

• Can be controlled with JavaScript
Canvas vs. SVG

- Not a matter of good or bad
- Both can be used and they can also be combined

<table>
<thead>
<tr>
<th>Canvas</th>
<th>SVG</th>
</tr>
</thead>
<tbody>
<tr>
<td>Low level</td>
<td>High level</td>
</tr>
<tr>
<td>Immediate mode</td>
<td>Retained mode</td>
</tr>
<tr>
<td>Fixed size</td>
<td>Scalable</td>
</tr>
<tr>
<td>Best for keyboard-based apps</td>
<td>Best for mouse-based apps</td>
</tr>
<tr>
<td>Animation (no object storage)</td>
<td>Medium animation</td>
</tr>
<tr>
<td>Pixels</td>
<td>XML object model</td>
</tr>
<tr>
<td>No interaction</td>
<td>User interaction (hit detection, events on the tree)</td>
</tr>
</tbody>
</table>
Canvas

• HTML5 element and plugin-free 2D drawing API that enables you to dynamically generate and render
  o Graphics
  o Charts
  o Images
  o Animation

• Canvas was originally introduced by Apple to be used in Mac OS X WebKit to create dashboard widgets
Canvas

• No object tree
• Can be manipulated with JavaScript and styled with CSS
• 2D Context
• 3D Context (Web GL)
Example: Canvas Animation

http://www.canvasdemos.com
Browser Support for 2D Canvas

- Chrome 2.0+
- Firefox 3.0+ (3.5+)*
- Opera 9.6+ (10.5+)*
- Safari 3.1+ (4.0+)*

* Additional support for Canvas Text API
Browser Support for 3D Canvas (Web GL)

- Chrome 5.0+
- Nightly builds of Safari
- Planned for Firefox 3.7

* Additional support for Canvas Text API
SVG

• Modularized, XML-based language for describing 2D vector and mixed vector/raster graphics
• SVG has been around on the Web since 2002 (using plugins)
• SVG provides high fidelity graphics at any zoom level
• Until now mostly used for static content (maps, org charts, floor plans, etc.)
• HTML5 now allows native SVG content in the browser (control with JavaScript, style with CSS)
SVG

• “Scalable” Vector Graphics—not bitmaps
Example SVG Game

SVG–oids  This re-creation of a classic 1979 arcade game shows what can be done with a little bit of SVG, JavaScript, and programming skill.

Use the up and down keys to move, the left and right keys to rotate, and the spacebar to fire. Press the "n" key to play in a nebula.

http://ie.microsoft.com/testdrive/Graphics/35SVG--oids/
Browser Support for SVG

The percentages shown are based on combined support for all SVG modules (for example Basic, Inline, filters):

- Chrome 2.0+ (58%)
- Firefox 3.0 (33%) 3.6 (46%)
- Opera 9.6+ (83%)
- Safari 3.1 (13%) 3.2 (46%) 4.0 (58%)

Note: IE 9 Will provide full support
HTML5 WebSocket
What do WebSocket and model trains have in common?
WebSocket

• Today’s Web applications demand reliable, real-time communication with near-zero latency, for example:
  o Financial applications
  o Social networking applications
  o Online games
  o Smart power grid
• HTTP is not designed for this
• “Real-time” often achieved through complex polling hacks (Comet)
• Comet lacks a standard implementation
WebSocket

• W3C API and IETF Protocol
• Full-duplex, single socket
• Enables web pages to communicate cross-origin with a remote host
• Traverses firewalls, proxies, and routers seamlessly (unlike Flash)
• Share port with existing HTTP content (no holes in the firewall needed)
• Low latency and enormous reduction in unnecessary network traffic and latency (up to 1000:1) over Comet
Unnecessary Network Throughput Reduction
Comet vs. WebSocket
Example: WebSocket-Based Quake II Game Port

http://code.google.com/p/quake2-gwt-port
Browser Support for WebSocket

- Chrome 4.0+
- WebKit Nightly builds
- Support planned for Firefox 3.7, Opera
- **Note**: Full HTML4 Emulation available
  - Kaazing WebSocket Gateway (supports native WebSocket and provides emulation for older browsers)
HTML5
Server-Sent Events
Server-Sent Events

- Server-Sent Events (SSE) standardizes and formalizes how a continuous stream of data can be sent from a server to a browser
- Effectively standardizes Comet and Reverse Ajax (Downstream/Server Push)
- EventSource API for broadcasting data from server to client
Example: News Broadcast

New York Times

“Self-Fulfilling Prophecies” broadcasts an undeniable cheer

Oscar Tuazon: ‘My Flesh to Your Bare Bones’
Oscar Tuazon has come up with a fine-tuned and personal response to a piece by Vito Acconci. It combines sound with fragmentary sculpture, both haunting. By KAREN ROSENBERG

Human Orbits
Certain books are so bad that they’re actually rather good. Ian McEwan’s new novel is just the opposite: it’s so ingeniously designed, irreproachably high-minded and skillfully brought off that it’s actually quite bad. By WALTER KIRN

Thai Commandos Raid Protesters’ Hotel in Bangkok
Special forces had surrounded the hotel in downtown Bangkok, where leaders of the anti-government protests that have roiled the capital were staying. By SETH MYDANS

Charter Extension for Stanford’s School Denied and Highlights from Friday’s Bay Area Pages
Excerpts from this week’s Bay Area Report. By THE NEW YORK TIMES

Eve of Destruction

http://kaazing.me/
Browser Support for Server-Sent Events

• Opera 9.0+ partial support
• Development in Firefox Trunk
• **Note**: With WebSocket support, SSE support is not as important
HTML5 Communication
HTML5 Communication

HTML5 defines two improved document communication features

- Cross Document Messaging
- XMLHttpRequest Level 2
Cross Document Messaging

- Enables secure cross-origin communication across iframes, tabs, and windows (using origin security)
- Provides asynchronous message passing between JavaScript contexts
- HTML5 clarifies and refines domain security by introducing the concept of an *origin*
- Defines the `postMessage` API as a standard way to send messages
PostMessage Architecture
Browser Support for Cross Document Messaging

- Chrome 2.0+
- Firefox 3.5+
- IE 8.0+
- Opera 9.6+
- Safari 4.0+
XMLHttpRequest Level 2

• XMLHttpRequest is the API that made Ajax possible
• XMLHttpRequest Level 2 significantly enhances XMLHttpRequest
  o Progress events
  o Cross-origin XMLHttpRequest
• XMLHttpRequest Level 2 allows for cross-origin XMLHttpRequests using Cross Origin Resource Sharing (CORS)
  http://www.w3.org/TR/access-control/
Browser Support for XMLHttpRequest Level 2

- Chrome 2.0+
- Firefox 3.5+
- Safari 4.0+
HTML5 Web Workers
Web Workers

- Provide background processing capabilities to web applications
- Typically run on separate threads so apps can take advantage of multicore CPUs
  - Background number-crunchers
  - Background price updates from server
  - Search queries
Example: Prime Numbers

Click "Go!" to calculate prime numbers w/out hosing your browser.

The highest prime number discovered so far is:

19738109

Source code adapted from W3C working draft.

http://htmlfive.appspot.com/static/primes-good.html
Browser Support for Web Workers

- Chrome 4.0+
- Firefox 3.5+
- Safari 4.0+
HTML5 Geolocation
Geolocation

• Allows users to share their location for location-aware services
  o Show user's position on map
  o Tag content (photos/sound/video)
  o Turn-by-turn navigation
  o Alert users of nearby points of interest
  o Social networking

• Users must be allowed to opt in to location sharing
Location Sources

• A device can use any of the following sources:
  o IP address
  o Coordinate triangulation
  o Global Positioning System (GPS)
  o Wi-Fi with MAC addresses from RFID, Wi-Fi, and Bluetooth
  o GSM or CDMA cell phone IDs
  o User defined

• Each have pros and cons
Privacy Architecture

1. Geolocation Application
2. Browser
3. Device
4. Internet

Server that hosts the Geolocation Application
External Location Service
**Example: Geolocation Demo**

This Demo works with a GeoLocation aware browser such as **Firefox 3.1b3 (or later)**. If you don't have hardware installed to determine your location you can install an **addon** for Firefox where you can set your location.

Back to [my blog](http://maxheapsize.com/static/html5geolocationdemo.html).

http://maxheapsize.com/static/html5geolocationdemo.html
Example: Privacy Policy
Browser Support for Geolocation

- Firefox 3.5+
- Safari 4.0+ on iPhone
- Chrome (Dev channel)
- Support planned in Opera
- Also available in Gears
HTML5 Web Storage
Web Storage

• Web Storage a.k.a “cookies on steroids”
• Three types:
  1. Session Storage Area
  2. Local Storage Area
  3. Web SQL Database
Cookies vs. Web Storage

- Cookies are limited in size
- Cookies are transferred back and forth on every request (bandwidth)
- Cookies are visible on the network (security)
- Data leaks (while switching tabs)
- Web Storage provides a simple API for storing values in easily retrievable JavaScript objects which persist across page loads
Example: Storage Demo

http://html5demos.com/storage
Browser Support for Web Storage

- Chrome 3.0+
- Firefox 3.0+
- Safari 4.0+
- Opera 10.5+
- IE 8.0+
Browser Support for Web SQL Database

- Chrome 3.0+
- Safari 3.2+
- Opera 10.5+

**Note:** Web SQL Database is based on SQLite and listed as stalled in the WHATWG spec. Indexed Database (a.k.a WebSimpleDB) is simpler and with no specific SQL database version, but not implemented yet.
Other HTML5 Features
Offline Web Applications
Offline Web Applications

- Offline Web Applications
  - HTML5 allows detection of online/offline mode
  - Users can continue to interact with web applications and documents when their network connection is unavailable
  - Example: Gmail (Moving away from Gears)
- Use a cache manifest file with details about files to be cached
- Browsers cache data is in the application cache
Browser Support for Offline Web Applications

- Chrome 4.0+
- Firefox 3.5+
- Safari 4.0+
Other HTML5 Features

• Drag and Drop Mechanism,
  o Currently only in Firefox 3.5+

• History
  o Includes nested browsing contexts
  o Not widely supported yet

• ContentEditable
  o Editable elements
  o HTML5 provides a formalization of what is already implemented in HTML4 (all browsers)
Resources

• “When Can I Use?” a website with up-to-date HTML5 Feature support matrix:
  http://a.deveria.com/caniuse

• WHATWG HTML5 Spec
  o http://www.whatwg.org/specs/web-apps/current-work/

• W3C HTML5 Spec
  o http://dev.w3.org/html5/spec/Overview.html
HTML5 Helper Tools

• modernizr
  o Small, simple JavaScript library that checks support for HTML5 and CSS3 features
  o Moves away from user-agent sniffing and uses feature detection instead
  o http://www.modernizr.com/

• html5shiv
  o Creates the new HTML5 DOM elements in Internet Explorer so that styles can be applied to them
  o http://code.google.com/p/html5shiv/
HTML5 Feature Emulation

• JavaScript-based emulations
• Explorercanvas—Canvas emulation for Internet Explorer: http://code.google.com/p/explorercanvas/
• SVG Web Toolkit—SVG in Internet Explorer: http://code.google.com/p/svgweb/
• WebSocket emulation for HTML4 browsers: Kaazing WebSocket Gateway www.kaazing.com
THANKS!

Also check out Sidda’s BOF session about WebSocket @7 in Foothill A

And another WebSocket presentation on Thursday

E-mail: peter.lubbers@kaazing.com

Twitter: @peterlubbers
Copyright © 2010 Kaazing Corporation, All rights reserved.

All materials, including labs and other handouts are property of Kaazing Corporation. Except when expressly permitted by Kaazing Corporation, you may not copy, reproduce, publish, or display any part of this training material, in any form, or by any means.