

Slide 1: How did I get here?

In 1991, a lot was going on, technologically speaking. The Internet, long a closed system, was opening up. Systems like Gopher, Archie and Veronica democratized information to an extent not seen since Gutenberg's printing press, but an emerging system known as the World Wide Web was about to make those systems historical footnotes. In a very short time, the WWW and the Internet would become virtually synonymous.

Slide 2: WWW

I was in college at this time, and one of my assignments this year was to write a term paper about something that would affect education in the future. Subjects included ebonics, word recognition by the shape of the word, new math and so on. I wrote about this emerging World Wide Web. We had a friendly vote in the Education department over whose idea would be the most influential. I came in dead last.

Slide 3: No one would ever use this!

"No one," one of the comments read, "would ever use this!" 20 years later, I think we're beyond the point where "I told you so" is necessary, but perhaps I'll post a recap after next year's reunion.

Slide 4: Hypertext

The fundamental technology of the WWW is called hypertext.

The term "hypertext" was coined in 1963, and hypertext was implemented in various forms through the years. It would be nearly three decades before hypertext's most famous implementation would be unleashed—the World Wide Web.

Slide 5: HTML and HTTP

The need to share hypertext drove the invention of two technologies. One, a way to format the text, and the other, a way to transmit the text. We know these as HTML and HTTP today.

In 1989, Tim Berners-Lee wrote a paper suggesting an Internet-based hypertext system for sharing documents. In 1990, working on a NeXT computer, Berners-Lee released the "next big thing", named the WorldWideWeb project. The WordWideWeb included a transport protocol, markup language and a hypertext browser.

The HTML spec was formally introduced in 1991, unleashing the World Wide Web to savvy ubergeeks in research institutions everywhere.

Slide 6: Lynx

In 1992, the gopher client Lynx was updated to include support for HTTP and HTML, and the era of the World Wide Web began. Early HTML and browsers did not have a lot of options for text decoration or layout, allowing scientists, engineers and programmers to fully utilize their design skills.

Slide 7: Mosaic and HTML 2.0

In 1993, Mosaic brought the WWW into a graphical era, and the HTML 2.0 spec was released. For the first time, graphics and text were displayed side by side on an HTML page, and the

beginnings of “web page design” was upon us. The HTML 2.0 spec brought better layout and markup, and it still a valid spec today.

Slide 8: HTML 3, Netscape

In 1994, the incredible pace of WWW development continued, with the release of the HTML 3.0 spec and a revolutionary client side programming language named JavaScript. Also released in 1994 is “the browser that changed everything”—Netscape.

Slide 9: IE

Duh-duh-duuhhhhh!

Microsoft gets into the game with Internet Explorer, bringing with it the absolute joy of cross-browser coding.

A major limitation of an HTML page is that it is a static text file. The need for ways to produce HTML dynamically manifested itself, and technologies like PHP and Cold Fusion began to appear. The introduction of mixed server code and static HTML brought about another infamous WWW technology—spaghetti code!

Also emerging in 1995 is the concept of automated indexing and searching of websites. Prior to this time, there were few enough sites that they could be categorized manually. But automated crawling and indexing systems made the task more accurate and more scalable. Because of its speed, AltaVista quickly became the most popular search engine of the early World Wide Web. Since web pages at this time are still mainly documents with simple navigation, it was fairly easy to figure out what needed to be indexed.

Slide 10: 1996

HTML 4.0, HTTP 1.1 and CSS 1 were all introduced in 1996.

If you’re keeping count, these are the last approved specifications. In five short years, HTML was invented and had three subsequent versions released. In the 15 years since, there has been no subsequent agreed-upon HTML standard.

Slide 11: Paper masthead

To recap, the first few years after the release of the WorldWideWeb project saw an incredible rate of advancement in standards and browsers. Over time, a technology designed for the sharing of scientific documents now allowed artists and animators to get into the design game.

With the entry of designers came the advent of UI design. UI design brought us a more sophisticated page layout, featuring thoughtfully designed graphics meant to enhance the user’s interaction with the website.

Slide 12: Hamster Dance

UI design also brought us a website committed entirely to dancing hamsters.

Also during this time, people began to figure out you could sell stuff on the WWW. The ecommerce era, and the first wave of next big things was upon us. UPS and FedEx began to deliver all manner of books, dog food and Pez dispensers right to our doors.

Slide 13: 1998, CSS2 spec lifecycle

In 1998, the CSS 2.0 spec was issued. As of 2011, the CSS 2 spec was still a draft.

Slide 14: 1999-2008

When we look at the timeline of when some very pivotal technologies were released, it's hard to believe there isn't better support and acceptance. Personally I attribute this to the ongoing browser war, where good ideas weren't implemented across the board simply because they were someone else's idea.

Slide 15: Menu and Really Important Information.

As time marched on, the amount of content on the World Wide Web grew tremendously, and the need to filter this information became more severe. Search engines grew in number and popularity. But the only information available to them was markup--code intended for the visual display of the text. There were no clues as to what the information meant.

Today, search engines are treated as the gateway to the Internet, which is a fundamental change in how the World Wide Web is accessed. And still, SEs have no good way to tell the difference between navigation and really important stuff. Plus, search engines can be gamed easily.

Slide 16: Refreshable Braille Display

Hide and seek content is not the only reason to introduce this level of change. As the WWW began to play a more fundamental role in our lives, the vision impaired became less and less able to participate. Assisted readers were developed, but they suffered from the same misunderstanding of information on an HTML page as SEs.

The major limitation of the current technology is that the current set of HTML tags doesn't adequately express what the conceptual meaning of their content is. On a blog site, looking at the code, we can't really separate the menu from the post body from the footer text. The function or meaning of the words --the semantics of the content--are not easily described with just the use of P, DIV and SPAN tags.

Slide 17: Bar chart of # of developers (n) and ways of doing things (1.32n)

SEs and government bodies and independent organizations published numerous guidance suggestions for improving how we discern content. But, all the right ways mean there is no one clear way to provide the necessary markup information.

Which brings us to today. HTML 4, CSS 2/3 and JavaScript have undergone incremental changes, but for the most part, have not changed significantly in 15 years. JS libraries like jQuery, YUI and Moo have been developed. We use these libraries to manipulate HTML, but they don't give any clues as to what the information means. Companies have spent millions and millions of dollars and man hours trying to make sense of the information on a web page. What we really needed all along was a better set of tags, better guidance in using these tags, and not trying to compensate for bad markup.

Slide 18: Semantic Markup

The new "semantic markup" is intended to be this better set of tags. In our HTML, we can now clearly delineate between menu text and a blog post with the <nav> and <article> tags. These new tags are more than just CSS shortcuts--these are actual guides to search engines and assistive reading devices.

Slide 19: Input types

Better guides for content are not the only advancement in HTML 5. Visually, the new input types and validations are intended to make websites more user friendly, as well as reducing input errors.

Slide 20: Browser support

But there are still issues with browser support. None of today's browser's fully implements the HTML 5 features. Also, since there are more browsers today than in years past, more versions of these browsers, and more people using browsers, it will be some time before we can assume a client browser is HTML 5 compliant.

Slide 21: Quirks mode

This means we're probably in for a long painful existence in "quirks mode", where few of the benefits of the new tags and technologies are realized. This shouldn't dissuade us from beginning to implement HTML 5 right away, we just remember to test our sites in a number of browsers, and make sure QA and support personnel have a number of browsers available to them as well.

Slide 22: IE 6 countdown

To really speed up the adoption of more modern browsers, we'll need to see more efforts like the IE6 Countdown, where major sites had to deny access to people using the browser and its own manufacturer had to initiate a program to get rid of it.

Slide 23: Database design

The new tags will do more than change the change how we design and users interact with our site. The new input types will challenge our database capabilities and schema design. For instance, if we're accepting date only input, we should store date only values.

Slide 24: Application shell

With the increased capabilities of browsers, and increasing functionality in JavaScript libraries, can we consider that the browser is evolving from a display app to an application shell?

Slide 25: An actual standard?

One of the big questions surrounding HTML 5 is whether or not there will be an actual specification. The W3C is targeting 2014 for a “Recommendation”, which is one of the preliminary steps before becoming a final specification. The length of time and focus on certain details has led to some criticism of the W3C. An advantage of the W3C’s approach is that there is a defined set of criteria for being considered “compliant”.

A second working group is also developing an HTML 5 spec, but in truth, an HTML 5 specification does not exist anymore for this group. Instead, the WHATWG is working on a living standard simply known as HTML. A living standard allows for incremental advances, but is also a moving target for a full implementation.

With multiple working groups with differing ideas regarding a specification will probably lead to some confusion in the browser market. Where the groups match up, the path will be very clear, but where there is divergence, their efforts will not have done much to improve the situation we have today.

Slide 26: Technology choice

When I’m out in the community, I’m often asked if I were building an application today, would I choose HTML 5 over another technology, such as Silverlight or Flash.

Slide 27: My precious!

When it comes to the choice of technologies, the decision may be more a case of compatibility over capability. The first question I ask is whether or not the application may be run on an iPad, and very often the answer is an emphatic affirmative. Our technology choice may be determined by what smartphone or tablet the executive team holds dear.

Above all other features, compatibility may be the “killer feature” of HTML 5. The browser is becoming the ultimate runtime, providing cross-platform support not seen in other technologies.

Being a huge fan of my iPad, I do appreciate the irony of creating my presentation about HTML 5 in a Flash application.

Slide 28: Onward!

It’s time to put what we’ve learned into action. Make a concerted effort to architect in new HTML features as the opportunity arises. Amazing things can be done in the browser right now, and awesomeness is on its way. Start building your web apps for the future.

Slide 29: That’s all, folks!

On behalf of Code Project and all the speakers today, we thank you for attending. It means a great deal to all of us that you’ve given your time to hear what we have to say.

Be cool, learn, build good things, share your knowledge

We'll see you in the community!

Slide 30: URL for script and references

For the script of this keynote, including a list of references and credits, visit c1.ms/c. As you add HTML 5 features to your web apps, have a look at the amazing Wijmo jQuery UI widgets, providing interactive charts and grid not found anywhere else. If you're on Twitter, communicate with me there at [rj_dudley](https://twitter.com/rj_dudley).

Thanks again for your participation!

References and Credits

<http://evolutionofweb.appspot.com/>

<http://www.devproconnections.com/article/html5/html5-aspnet-developer-139543>

http://www.livinginternet.com/w/wi_browse.htm

<http://en.wikipedia.org/wiki/Hypertext>

NeXT Logo from <http://www.logodesignlove.com/next-logo-paul-rand>

<http://en.wikipedia.org/wiki/Html>

[http://en.wikipedia.org/wiki/Lynx_\(web_browser\)](http://en.wikipedia.org/wiki/Lynx_(web_browser))

Netscape logo from <http://www.goodlogo.com/extended.info/3026>

http://en.wikipedia.org/wiki/Mosaic_browser

[http://en.wikipedia.org/wiki/Netscape_\(web_browser\)](http://en.wikipedia.org/wiki/Netscape_(web_browser))

http://en.wikipedia.org/wiki/Internet_explorer

<http://www.hampsterdance.com/classics/originaldance.htm>

http://en.wikipedia.org/wiki/Alta_Vista

Refreshable Braille display

<http://www.afb.org/Section.asp?DocumentID=3652&SectionID=7&SubTopicID=97&TopicID=330>

<http://lists.whatwg.org/pipermail/whatwg-whatwg.org/2009-December/024477.html>

<http://makingcreativematter.com/2011/04/my-ipad-completes-me/>