This book is about HTML5 Programming. Before you can understand HTML5 programming, however, you need to take a step back and understand what HTML5 is, a bit of the history behind it, and the differences between HTML4 and HTML5.

In this chapter, we get right to the practical questions to which everyone wants answers. Why HTML5, and why all the excitement just now? What are the new design principles that make HTML5 truly revolutionary—but also highly accommodating? What are the implications of a plugin-free paradigm; what’s in and what’s out? What’s new in HTML, and how does this kick off a whole new era for web developers? Let’s get to it.

**The Story So Far—The History of HTML5**

HTML goes back a long way. It was first published as an Internet draft in 1993. The ’90s saw an enormous amount of activity around HTML, with version 2.0, versions 3.2, and 4.0 (in the same year!), and finally, in 1999, version 4.01. In the course of its development, the World Wide Web Consortium (W3C) assumed control of the specification.

After the rapid delivery of these four versions though, HTML was widely considered a dead-end; the focus of web standards shifted to XML and XHTML, and HTML was put on the back burner. In the meantime, HTML refused to die, and the majority of content on the web continued to be served as HTML. To enable new web applications and address HTML’s shortcomings, new features and specifications were needed for HTML.

Wanting to take the web platform to a new level, a small group of people started the Web Hypertext Application Working Group (WHATWG) in 2004. They created the HTML5 specification. They also began working on new features specifically geared to web applications—the area they felt was most lacking. It was around this time that the term Web 2.0 was coined. And it really was like a second new web, as static web sites gave way to more dynamic and social sites that required more features—a lot more features.

The W3C became involved with HTML again in 2006 and published the first working draft for HTML5 in 2008, and the XHTML 2 working group stopped in 2009. Another year passed, and that is where we stand today. Because HTML5 solves very practical problems (as you will see later), browser vendors are feverishly implementing its new features, even though the specification has not been completely locked down. Experimentation by the browsers feeds back into and improves the specification. HTML5 is rapidly evolving to address real and practical improvements to the web platform.
Brian says: “Hi, I’m Brian, and I’m an HTML curmudgeon. I authored my first home page back in 1995. At the time, a ‘home page’ was something you created to talk about yourself. It usually consisted of badly scanned pictures, `<blink>` tags, information about where you lived and what you were reading, and which computer-related project you were currently working on. Myself and most of my fellow ‘World Wide Web developers’ were attending or employed by universities.

At the time, HTML was primitive and tools were unavailable. Web applications hardly existed, other than a few primitive text-processing scripts. Pages were coded by hand using your favorite text editor. They were updated every few weeks or months, if ever.

We’ve come a long way in fifteen years.

Today, it isn’t uncommon for users to update their online profiles many times a day. This type of interaction wouldn’t have been possible if not for the steady, lurching advances in online tools that built on each previous generation.

Keep this in mind as you read this book. The examples we show here may seem simplistic at times, but the potential is limitless. Those of us who first used `<img>` tags in the mid-1990s probably had no idea that within ten years, many people would be storing and editing their photos online, but we should have predicted it.

We hope the examples we present in this book will inspire you beyond the basics and to create the new foundation of the Web for the next decade.”

The Myth of 2022 and Why It Doesn’t Matter

The HTML5 specification that we see today has been published as a working draft—it is not yet final. So when does it get cast in stone? Here are the key dates that you need to know. The first is 2012, which is the target date for the `candidate recommendation`. The second date is 2022, which is the `proposed recommendation`. Wait! Not so fast! Don’t close this book to set it aside for ten years before you consider what these two dates actually mean.

The first and nearest date is arguably the most important one, because once we reach that stage, HTML5 will be complete. That’s just two years away. The significance of the proposed recommendation (which we can all agree is a bit distant) is that there will then be two interoperable implementations. This means two browsers equipped with completely interoperable implementations of the entire specifications—a lofty goal that actually makes the 2022 deadline seem ambitious. After all, we haven’t even achieved that in HTML4.

What is important, right now, is that browser vendors are actively adding support for many very cool new features. Depending on your audience, you can start using many of these features today. Sure, any number of minor changes will need to be made down the road, but that’s a small price to pay for enjoying the benefits of living on the cutting edge. Of course, if your audience uses Internet Explorer 6.0, many of the new features won’t work and will require emulation—but that’s still not a good reason to dismiss HTML5. After all, those users, too, will eventually be jumping to a later version. Many of them will probably move to Internet Explorer 9.0 right away, and Microsoft promises to design that browser