AJAX and PHP

Building Modern Web Applications – Second Edition

Build user-friendly Web 2.0 Applications with JavaScript and PHP

Bogdan Brinzarea-Iamandi

Cristian Darie

Audra Hendrix

AJAX and PHP
Building Modern Web Applications – Second Edition

Credits

Authors

Bogdan Brinzarea-Iamandi

Cristian Darie

Audra Hendrix

Reviewer

Kalpesh Barot

Acquisition Editor

Douglas Paterson

Development Editor

Dhiraj Chandiramani

Technical Editor

Aanchal Kumar

Indexer

Rekha Nair

Editorial Team Leader

Gagandeep Singh

Project Team Leader

Lata Basantani

Project Coordinators

Srimoyee Ghoshal

Rajashree Hamine

Proofreader

Sandra Hopper

Graphics

Nilesh Mohite

Production Coordinators

Adline Swetha Jesuthas

Dolly Dasilva

Cover Work

Dolly Dasilva

About the Authors

Bogdan Brinzarea-Iamandi is a software engineer having a strong background in Computer Science. He holds a Master and Bachelor Degree from the Automatic Control and Computers Faculty at the Politehnica University of Bucharest, Romania. He also has an Auditor diploma from the Computer Science department at Ecole Polytechnique, Paris, France.

His main interests include software architecture, web technologies, distributed computing, and software methodologies. Currently, he is the Software Development Manager at a Romanian bank, Banca Romaneasca, a member of the National Bank of Greece, where he coordinates the development and implementation of enterprise software for the banking industry.

He is also the author of the books *AJAX and PHP: Building Responsive Web Applications* and *Microsoft AJAX Library Essentials: Client-side ASP.NET AJAX 1.0 Explained.*

Cristian Darie is a software engineer with experience in a wide range of modern technologies, and is the author of numerous books, which are all listed on his homepage at `http://www.cristiandarie.ro`. Cristian is the manager and the former technical architect of `http://www.okazii.ro`, the largest Romanian e-commerce website.

Audra Hendrix was educated at Northwestern University. She works as a consultant in applied technology and marketing to small and medium-sized businesses. While her client list includes Fortune 500 companies, she prefers the flexibility, agility, and challenges of small to medium-sized businesses. She has consulted both in the United States and France for businesses seeking to better utilize their resources and maximize their gains by reinventing and reapplying back office and Internet applications, data management, cost-effective marketing strategies, staffing requirements, and planning and deployment of new or emerging product lines.

A special thanks goes out to my daughter, Zsa Zsa—an unending and joyful source of inspiration and boundless love. You are, by far, my greatest achievement.

About the Reviewer

Kalpesh Barot has about five years of experience in the world of PHP. He has worked extensively on small- and large-scale social networking websites developed in PHP. He has been involved in varied projects, from planning and developing websites to creating custom modules on big social networking websites.

He received a Masters degree in Enterprise Software Engineering from University of Greenwich, UK. There he learned the theory behind his computer experience and became a much more efficient computer programmer.

He has worked actively in the IT sector since his freshman year at the university. He has been a PHP developer since then and has developed his skills in this field.

Through his increasing responsibilities, he has learned to prioritize needs and wants, and applies this ability to his projects. He has acted as a technical reviewer for *OOP with PHP* for Packt Publishing.

> I would like to thank my wife, Bansari Barot, for her continued support in all my projects and Rajashree Hamine for her constant efforts in reminding me to review the chapters on time.

Table of Contents

Preface

AJAX is a complex phenomenon that means different things to different people. Computer users appreciate that their favorite websites are now friendlier and feel more responsive. Web developers learn new skills that empower them to create sleek web applications with little effort. Indeed, everything sounds good about AJAX!

At its roots, AJAX is a mix of technologies that lets you get rid of the evil page reload, which represents the dead time when navigating from one page to another. Eliminating page reloads is just one step away from enabling more complex features into websites, such as real-time data validation, drag-and-drop, and other tasks that weren't traditionally associated with web applications. Although the AJAX ingredients are mature (the `XMLHttpRequest` object, which is the heart of AJAX, was created by Microsoft in 1999), their new role in the new wave of web trends is very young, and we'll witness a number of changes before these technologies will be properly used to the best benefit of the end users.

AJAX isn't, of course, the answer to all the Web's problems, as the current hype around it may suggest. As with any other technology, AJAX can be overused, or used the wrong way. AJAX also comes with problems of its own: you need to fight with browser inconsistencies, AJAX-specific pages don't work on browsers without JavaScript, they can't be easily bookmarked by users, and search engines don't always know how to parse them. Also, not everyone likes AJAX. While some are developing enterprise architectures using JavaScript, others prefer not to use it at all. When the hype is over, most will probably agree that the middle way is the wisest way to go for most scenarios.

In *AJAX and PHP: Building Modern Web Applications – Second Edition*, we take a pragmatic and safe approach by teaching relevant patterns and best practices that we think any web developer will need sooner or later. We teach you how to avoid the common pitfalls, how to write efficient AJAX code, and how to achieve functionality that is easy to integrate into current and future web applications, without requiring you to rebuild the whole solution around AJAX. You'll be able to use the knowledge you learn from this book right away, in your PHP web applications.

What this book covers

Chapter 1: The World of AJAX and PHP is all about a quick introduction to the world of AJAX. In order to proceed with learning how to build AJAX applications, it's important to understand why and where they are useful. It describes the `XMLHttpRequest` object, which is the key element that enables the client-side JavaScript code to call a page on the server asynchronously.

Chapter 2: JavaScript and the AJAX Client walks you through many fields such as working with HTML, JavaScript, CSS, the DOM, XML, and `XMLHttpRequest`. It discusses the theory (and practice) that you will need to know to make these components come together smoothly, and form a solid foundation for your future AJAX applications. It also shows you how to implement simple error-handling techniques, and how to write code efficiently.

Chapter 3: Object Oriented JavaScript covers a large area of what object-oriented programming means in the world of JavaScript starting from basic features and going far into the execution context of functions. It teaches you the basic OOP concepts—encapsulation, polymorphism, and inheritance, how to work with JavaScript objects, functions, classes, and prototypes, how to simulate private, instance, and static class members in JavaScript, what the JavaScript execution context is, how to implement inheritance by using constructor functions and prototyping, and the basics of JSON.

Chapter 4: Using PHP and MySQL on the Server starts putting the server to work, using PHP to generate dynamic output, and MySQL to manipulate and store the backend data. This chapter shows you how to use XML and JSON with PHP (so that you can create server-side code that communicates with your JavaScript client), how to implement error-handling code in your server-side PHP code, and how to work with MySQL databases.

Chapter 5: AJAX Form Validation creates a form validation application that implements traditional techniques with added AJAX flavor, thereby making the form more user-friendly, responsive, and pleasing. The intention of this chapter isn't to build the perfect validation technique but, rather, a working proof of concept that takes care of user input and ensures its validity.

Chapter 6: Debugging and Profiling AJAX Applications teaches how to enable and use Internet Explorer's debugging capabilities. It shows how you can work with Web Development Helper, Developer Toolbar, and other Internet Explorer tools and with Firefox plugins such as Firebug, Venkman JavaScript Debugger, and Web Developer.

Chapter 7: Advanced Patterns and Techniques briefly covers some of the most important patterns and techniques covering usability, security, and techniques. Looking at methods, patterns, and techniques is so important that it has developed into its own science and has created a set of guidelines for typical problems that offer us predictable results.

Chapter 8: AJAX Chat with jQuery teaches how to use AJAX to easily implement an online chat solution. This will also be your opportunity to use one of the most important JavaScript frameworks around—jQuery. More precisely, this chapter will explain the basics of jQuery and show how to create a simple, yet efficient client-server chat mechanism using AJAX.

Chapter 9: AJAX Grid explains the usage of an AJAX-enabled data grid plugin, jqGrid.

Appendix: Preparing Your Working Environment covers the installation instructions that set up your machine for the exercises in this book. It also covers preparing the database that is used in many examples throughout the book.

What you need for this book

To go through the examples in this book you need PHP 5, a web server, and a database server. We have tested the code under several environments, but mostly with the Apache 2 web server, and MySQL 4.1 and MySQL 5 databases.

You can choose, however, to use another web server, or another database product, in which case the procedures presented in the chapters might not be 100% accurate. It is important to have PHP 5 or newer, because we use some features, such as Object Oriented Programming support, which aren't available in older versions.

Please read the appendix for more details about setting up your machine. If your machine already has the required software, you still need to read the final part of appendix, where you are instructed about creating a database that is used for the examples in this book.

Who this book is for

This book is written for PHP developers who want to learn how to use PHP, JavaScript, MySQL, and jQuery to implement Web 2.0 applications, are looking for a step-by-step, example-driven AJAX tutorial, want to learn advanced AJAX coding patterns and techniques, and want to be able to assess the security and SEO implications of their code.

Conventions

In this book, you will find a number of styles of text that distinguish between different kinds of information. Here are some examples of these styles, and an explanation of their meaning.

Code words in text are shown as follows: "We can include other contexts through the use of the `include` directive."

A block of code is set as follows:

```
// create the second <ui> element and add a text node to it
oLiOrange = document.createElement("li");
oOrange = document.createTextNode("Orange");
oLiOrange.appendChild(oOrange);
```

When we wish to draw your attention to a particular part of a code block, the relevant lines or items are set in bold:

```
// create the second <ui> element and add a text node to it
oLiOrange = document.createElement("li");
oOrange = document.createTextNode("Orange");
oLiOrange.appendChild(oOrange);
```

Any command-line input or output is written as follows:

```
tar xvfz xampp-linux-X.Y.Z.tar.gz -C /opt
```

New terms and **important words** are shown in bold. Words that you see on the screen, in menus or dialog boxes for example, appear in the text like this: "Now click on the **Start Debugging** button. If you receive a confirmation window like that in the following screenshot, click on **OK**".

> Warnings or important notes appear in a box like this.

> Tips and tricks appear like this.

Reader feedback

Feedback from our readers is always welcome. Let us know what you think about this book—what you liked or may have disliked. Reader feedback is important for us to develop titles that you really get the most out of.

To send us general feedback, simply send an email to feedback@packtpub.com, and mention the book title via the subject of your message.

If there is a book that you need and would like to see us publish, please send us a note in the **SUGGEST A TITLE** form on www.packtpub.com or email suggest@packtpub.com.

If there is a topic that you have expertise in and you are interested in either writing or contributing to a book on, see our author guide on www.packtpub.com/authors.

Customer support

Now that you are the proud owner of a Packt book, we have a number of things to help you to get the most from your purchase.

> Downloading the example code for the book
> Visit http://www.packtpub.com/files/code/7726_Code.zip to directly download the example code.
> The downloadable files contain instructions on how to use them.

Errata

Although we have taken every care to ensure the accuracy of our content, mistakes do happen. If you find a mistake in one of our books—maybe a mistake in the text or the code—we would be grateful if you would report this to us. By doing so, you can save other readers from frustration, and help us to improve subsequent versions of this book. If you find any errata, please report them by visiting http://www.packtpub.com/support, selecting your book, clicking on the **let us know** link, and entering the details of your errata. Once your errata are verified, your submission will be accepted and the errata added to any list of existing errata. Any existing errata can be viewed by selecting your title from http://www.packtpub.com/support.

Piracy

Piracy of copyright material on the Internet is an ongoing problem across all media. At Packt, we take the protection of our copyright and licenses very seriously. If you come across any illegal copies of our works, in any form, on the Internet, please provide us with the location address or website name immediately so that we can pursue a remedy.

Please contact us at `copyright@packtpub.com` with a link to the suspected pirated material.

We appreciate your help in protecting our authors, and our ability to bring you valuable content.

Questions

You can contact us at `questions@packtpub.com` if you are having a problem with any aspect of the book, and we will do our best to address it.

1

The World of AJAX and PHP

"Computer, draw a robot!" said my young cousin to the first computer he had ever seen. (As I had instructed it not to listen to strangers, the computer wasn't receptive to this command.) If you're like me, your first thought would be *how silly* or *how funny*—but this is a mistake. We're being educated to accommodate computers, to compensate for the lack of ability of computers to understand humans, but in an ideal world, that spoken command should have been enough to have the computer please my cousin.

This book doesn't aim to teach you to create software applications that intelligently interact with children—we're still far from that point. However, we'll help you take a small but important step in that direction. We'll teach you how to best use web development technologies available today—AJAX and PHP in particular—to enhance web users' experience with your website, by creating more usable and friendly web interfaces. As far as this chapter is concerned, we'll discuss the following topics:

- **The big picture**: Here we'll answer a question we're often asked: *Why bother improving our applications' user interfaces and features, when the existing ones perform satisfactorily?*

- **Building websites since 1990**: What are the fundamental principles of the Web, and what are the important technologies that make it work? You probably know most of this, but we hope you'll welcome this quick refresher.

- **The world of AJAX**: As you will learn, AJAX is a powerful tool to improve your web interfaces. However, it's important to understand when you should and shouldn't use it. We'll also discuss the basic principles of AJAX, and refer to online resources and tools that can help you along the way.

- **Setting up your environment**: In this book, you'll find plenty of code—and be anxious to see it in action. We've taken care of that by including step-by-step instructions with every exercise.

- **Hello world!:** After reading so much pure theory, and installing many software packages (and we all know how *boring* software installation can be!), you'll be eager to write some code. So at the end of this chapter, you'll write your first AJAX application.

We hope your journey through this book will be a pleasant and useful one! Let's get started.

The big picture

The story about Cristian's seven-year-old cousin (which happened back in 1990) is still relevant today. The ability of technology to be user-friendly has evolved quite a bit, but there's still a long way to go before we have computers that self-adapt to our needs. For now, people must *learn* how to work with computers—some even end up loving a black screen with a tiny command prompt on it!

We will be very practical and concise in this book, but before getting back to your favorite mission (writing code)—it's worth taking a little step back . It's easy to forget that *the very reason technology exists is to serve people*, and make their lives more entertaining at home and more efficient at work.

The working habits of many are driven by software with intuitive (and enjoyable) user interfaces. Successful companies are typically one step ahead of their competition in offering their users more simple and natural ways to achieve their goals—explaining the popularity of the mouse, features such as **drag-and-drop**, and that simple textbox that searches the entire Web for you in just 0.1 seconds (or so it says).

Understanding the way people's brains work is one key to building the ultimate software application. We know that end users need intuitive user interfaces; they don't really care what operating system they're running as long as the functionality they get is what they want. The art of meeting users' interface expectations, understanding the nature of their work, and building software applications accordingly is referred to as **software usability.**

In the past, when users were specifically technically trained, the behavior of any software that interacted with humans was less important. Business needs today dictate that users aren't necessarily technically trained—administrative staff don't usually hold degrees in Computer Science, but still need to deliver good-looking reports for the sales manager, and easily create data entry forms for the sales force.

AJAX is a modern tool used to create user-friendly web applications. As with any other tool, however, it can be used improperly, complicating the user experience, neglecting users with disabilities, and/or lowering search engine performance. These issues can mean your site, and therefore your business, is losing customers, creating a bit of ill will, and/or damaging your reputation!

This being a programming book, our main focus will regard the technical aspects of writing AJAX PHP code. But as a responsible web developer, you should not lose sight of the complementary aspects that affect the success of a web application. To stay on top of this concern, we strongly recommend you check at least some of the following resources:

- *Don't Make Me Think: A Common Sense Approach to Web Usability, second edition*, by Steve Krug (New Riders Press, 2005)

- *Prioritizing Web Usability*, by Jakob Nielsen and Hoa Loranger (New Riders Press, 2006)

- *Designing Interfaces: Patterns for Effective Interaction Design*, by Jenifer Tidwell (O'Reilly, 2005)

- *Ambient Findability*, by Peter Morville (O'Reilly, 2005)

- *Bulletproof Web Design, second edition*, by Dan Cederholm (New Riders Press, 2007)

- *Professional Search Engine Optimization with PHP: A Developer's Guide to SEO*, by Cristian Darie and Jaimie Sirovich (Wrox Press, 2007)

AJAX and Web 2.0

These days, it's increasingly difficult to discuss AJAX without mentioning Web 2.0 (`http://en.wikipedia.org/wiki/Web_2`). What is Web 2.0? Initially, Web 2.0 was associated with the **Semantic Web** (`http://en.wikipedia.org/wiki/Semantic_web`). The Semantic Web is envisioned to be the next step in the Web's evolution, based on online social-networking applications, using tag-based *folksonomies* (user-generated tags for data categorization). Some say it is simply a marketing buzzword without any special meaning, while others use this term to describe the new, open, interactive Web that facilitates online information sharing and collaboration.

Controversies aside, the version number is an allusion to the recent changes of the World Wide Web. The new generation of web applications offers a richer user experience, much closer to that of desktop applications, while using live data from the Internet. In the world of Web 2.0, AJAX plays an essential role providing the technological support to implement rich and responsive web interfaces.

Building websites since 1990

Before getting into the details, let's take the inevitable history lesson to make sure we've got our definitions straight. We promise we'll keep this *short*. If you're a web development veteran, feel free to skip ahead to *The world of AJAX* section.

Although the history of the Internet is a bit longer, 1991 is the year when **HyperText Transfer Protocol** (**HTTP**), still used to transfer data over the Internet, was invented. In its initial versions, it didn't do much more than opening and closing connections. The later versions of HTTP (Version 1.0 appeared in 1996 and Version 1.1 in 1999) became the protocol that we all know and use.

HTTP and HTML

HTTP is supported by all web browsers, and it does its original job very well—retrieving simple web content. Whenever you request a web page using your favorite web browser, the HTTP protocol is assumed. So, for example, when you type `www.msn.com` in the location bar of your web browser, it will assume by default that you meant `http://www.msn.com`.

The standard document type of the Web is **HyperText Markup Language (HTML)**—a markup language that dictates a document's formatting and layout of *static text and images*. When you need to get to another HTML page via HTTP, you initiate a full page reload, and the HTML page you requested must already exist as a static document at the mentioned location prior to the request—it only enables users to retrieve static content (HTML pages) from the Internet. HTTP and HTML are still a very successful pair and are the foundation of the Web as we know it today. Figure 1-1 shows a simple transaction when a user requests a web page from the Internet using the HTTP protocol:
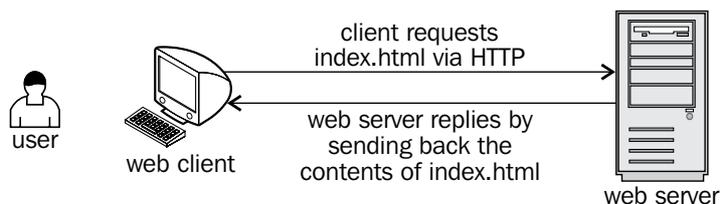

Figure 1-1: A simple HTTP request