

Introducing Web Services

I imagine you picked up this book for a reason.

Perhaps you've built a fantastic website for your organization, but now that it's done, you want more: more ways to view the information, more ways to alter the information, more ways to share the information. Web services can help.

Maybe you run a small business through eBay and want better ways to manage the items you are selling. Web services can help.

You could be a student, a teacher, or an institution wanting to leverage the vast wealth of information compiled by Google. Web services can help.

Web services can help you use the information on the Web in new ways, from simply allowing someone to subscribe to pockets of information so that they can be immediately notified when new information is available, to creating complex business applications that can manage complex real-time supply chains.

Before digging in to how to use web services in the following chapters, this chapter presents you with the following:

- ☐ What web services are
- ☐ Why web services are used
- ☐ How web services work
- ☐ Who uses web services
- ☐ And finally, why you should use web services

Defining Web Services

Web services are a collection of protocols that are used to exchange data between disparate applications or systems. The essence of web services is the open standards on which they are built, by leveraging public and common protocols like HTTP, along with the XML document model. Web services are easy to implement with existing technologies. Not only are you (as a PHP developer) already familiar with many aspects of web services and the tools required, but you already have the facilities required to deploy them.

Why They Are Used

Simply put, web services allow information to easily pass from those who have it to those who desire it. Both feeds and APIs allow the requestor to obtain information from the service, regardless of the information type. Web services allow customers to access the information they desire, without the extraneous information generally presented on web pages. For example, running a search for a book on Amazon.com will yield several relevant results, generally (under the current layout) in the center column of the page. The right and left columns will contain other information, not directly related to your search (recommended titles, recently viewed items, and so on). Performing the same search via the API will yield the same results, but without that extraneous information.

This points to a key factor in web services — they are used to obtain specific information, or complete specific tasks. Unlike people, the automated processes that utilize web services don't make impulse decisions (like choosing a recommended title from Amazon), so there is no point in presenting them with those types of options.

How They Are Used

Web services generally present information already available via another method (that is, a website). The advantage in the services is the consistent presentation of the information (in stark contrast to methods like screen scraping) in an easy-to-parse format. Here are some examples:

- ❑ Federal Express — Allows customers to do rate lookups and schedule shipments after logging into its online system. These same tasks can be accomplished via its API; as such, customers with larger shipping needs can integrate the API with their own systems to quickly automate shipping, and allow customers to determine how much it will cost to ship a particular product.
- ❑ Amazon — Allows customers to search for products via its website, and similar tasks can be accomplished via its API (often with more granularity than the traditional web interface provides). Using the API allows small booksellers to transparently integrate with Amazon to offer additional books and apply their own pricing.
- ❑ My Personal Blog — Allows people to read my thoughts on various professional matters. The feeds provided by my blog allow those people to read the posts on their own terms, within their own client, without ads or cute pictures of my cat.

It's important to recognize both types of interaction presented here. In some cases the web service is accessed directly by the end user (as is often the case with blogs and feeds). In other cases the service is consumed by an intermediate service, then presented (through various means) to its users.

Who Uses Them

If you accept my broad definition of web services, including both feeds and APIs, the answer is almost everyone. Most news sites offer at least some of their information via an XML feed. Blog sites almost without exception include some form of feed, and APIs are becoming more prevalent offerings from businesses of all sizes.

Also keep in mind that the scope of web services varies widely depending on who offers them. Many are merely informational in nature (such as the National Weather Service API), providing read-only access to information. Others, however, allow you to present information to the server; this could be purely digital in nature (such as adding a bookmark to your del.icio.us account), or it could set into motion a series of physical events (like scheduling a package pickup through FedEx).

Why You Should Use Web Services

There are two main reasons to use web services, both remarkably straightforward: someone else has information you require, or you have information you wish to provide to others. When planning a new web service, remember that servers can either provide or accept information. Remember that web services can provide any level of security, from a completely open service open to all who request it, to a private service with transfers happening over SSL and client-side certificates to validate identities.

Convincing the Boss

Here are a few more tidbits that should help convince the boss (or you) that web services are a route to look into:

- ❑ Web services make use of technology you already have deployed — you already have a web server, you already have a database, and so on. Why not provide another way to access the data?
- ❑ Web services fit into the security scheme you are already employing. Careful deployment will keep security concerns in line with regular web access.
- ❑ Technically savvy users will find a way to access the information they desire. Without web services they will need to resort to methods like screen scraping, which is more difficult to control and can be unreliable (and providing unreliable access to your information is worse than providing none at all).
- ❑ Web services can help manage costs; easy integration with companies (like FedEx) that offer services to you can result in lower variable costs for your transactions.
- ❑ Web services will allow your business to scale; allowing clients to integrate their systems with yours will help encourage continued relationships, and avoid devoting resources to creating systems for specific clients.

Summary

This chapter answered some basic questions about web services that should allow you to do the following:

- ❑ Explain what web services are and how they work
- ❑ See the value in utilizing web services in your own projects
- ❑ Convince others that it's definitely worth the effort to build and use web services

In the next chapter, you start working with web feeds, a simple but useful form of web services. As you progress through the book, you'll start getting into more complex uses of web services and eventually create your own web service APIs.