

# Website Architecture

*Memorandum*

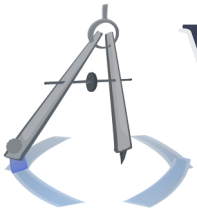
## Recommended Software

A list of software recommended for architecting websites.

Michael Serritella

Summer 2010





# Website Architecture

Memorandum

Recommended Software

## Introduction

This is a list of recommended software for architecting websites, the core of which will be used for the projects in this course. Those which are used for projects are **provided through a virtual-machine image of a server**, so the student does not have to acquire them. Details on the virtual-machine image are on the course website.

Most of the optional software would be of interest to someone who is beginning in Web programming (e.g. development environments, debugging tools).

I offer recommendations for a domain-name company and a Web-hosting company, along with basic configuration advice.

Additionally, I recommend using software tools that I have developed, which I detail at the end of this document. The tools are provided on my website via a permissive license.

## Server-side software

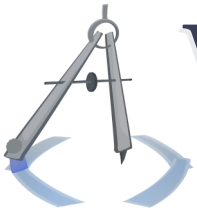
The following software runs on the server in order to enable software-driven websites. All of the relevant software here will be included in the virtual-machine image, so this is more for the sake of due diligence for those who want to install everything themselves.

### Note: AMP stacks

There are software bundles which install many of these components at once. This is particularly useful for Windows users, since the individual applications were seemingly designed for Linux and can be painful to install on Windows. The original package was **LAMP**: Linux, Apache, MySQL, and PHP. There are many flavors now, most notably those offered by **BitNami**. They generally have similar acronyms, like LAMPP and WAMP. The one with the best set of tools for this course is **WAPP**, found at **BitNami**.

A Linux virtual-machine image is provided with this course, anyway, so you likely don't even want to use WAPP. But it is there.





# Website Architecture

Memorandum

Recommended Software

## Apache

Apache interprets HTTP requests and thus communicates with the user to serve web pages and accept user input. It is the most common HTTP server in the world.

Apache doesn't frequently make major revisions, so any quasi-recent version will work perfectly well for this course.

**Official site:** [Apache HTTP Server](#)

## PHP

PHP is a scripting language which has lots of built-in string-handling functions and support for Web technologies, so it is suitable to make websites. It is by far the most common server-side scripting language, except for Java, which is technically the most popular language used on the Web, because it is used on so many other, non-Web platforms.

Any version after 5.0 is fine for this course. Version 5 was a partial rewrite which incorporated a lot of object-oriented features. Newer versions, like the most recent builds and the upcoming version 6, are probably not extremely useful, especially for this course.

PHP is typically integrated into Apache as an 'Apache module'. So you should probably install Apache first and then explore your options for installing the PHP module. PHP may also be run from the command line as `php [script_filename]`. Sometimes, you will see this invocation method called `php-cli` (command-line interface), though your executable name will likely just be `php`.

**Official site:** [PHP](#)

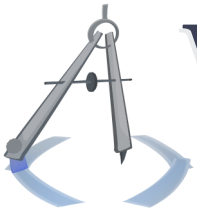
## PostgreSQL

PostgreSQL is a fully-featured database management system which is a competitor of MySQL. It is far better architected than MySQL, and its feature set is vastly more mature. We will not use MySQL in this course, and you should probably not use MySQL in your life.

Any quasi-recent version of PostgreSQL is seemingly fine for this course. The software is pronounced "post-gress-Q.L."

**Official site:** [PostgreSQL](#)





# Website Architecture

Memorandum

Recommended Software

## SQLite

SQLite is a database management system which can execute SQL commands on a simple file. It is lightweight and pretty well-featured. Both Linux and Windows versions are available. SQLite is the premier embedded DBMS on the Web today.

Note that version 2 is significantly different from version 3; the database files generated by one version are incompatible with the other (not even backwards-compatible). When you invoke the version 3 executable on Linux, you invoke it as `sqlite3`. For a summary of the version differences, see the official site's [SQLite Version 3 Overview](#).

It's quite possible that either version is OK for a site which is not too dependent on the DBMS; just be careful to use the same version on your home machine as the one on the Web server. We will be using version 3 in this course.

**Official site:** [SQLite](#)

## Source-code editors

The following programs should help you code in Web languages by at least syntax highlighting.

### Linux

Source-code editors for Linux:

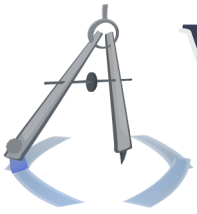
#### gedit

gedit is a text editor with syntax highlighting for every computer language you care about. It will change the formatting of text according to the language's grammar, like an IDE does, but it does not track information about your program, like function names and variable names, so it cannot offer any code-completion or any other such assistance.

This is simply called "Text Editor" in some places within the Gnome desktop environment. It is my default source-code editor in Linux. I have never tried any Linux IDEs for Web languages.

gedit is likely pre-installed with your Linux distribution. Else it could be downloaded via a package manager.





# Website Architecture

Memorandum

Recommended Software

## Windows

Source-code editors for Windows:

### Notepad++

This is a good syntax highlighter, similar to `gedit`. And like `gedit`, it may occasionally detect the wrong language upon first opening a document. Correct it, and you're good to go. This is my default source-code editor on Windows.

**Cost:** Free.

**Official site:** [Notepad++](#)

**Portable version:** [Notepad++ at PortableApps](#)

### phpDesigner

I don't really like IDEs for Web, because the application is usually distributed across so many languages and components that no IDE can have comprehensive scope, and then making up for its shortcomings is more trouble than it's worth. Plus, I know the languages well enough that I don't need a lot of the code-completion and knowledge-base features. However, if you are not (yet) like me, you may want to try `phpDesigner`. A couple years ago, it was free for personal use. I tried it, and I thought it was pretty alright. If you're using Windows and need some training wheels for PHP, perhaps you can consider it. I haven't used any other IDEs, so I wouldn't know how it compares.

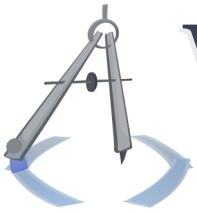
**Cost:** Not free.

**Official site:** [phpDesigner](#)

## Client-side debugging

Client-side languages can be difficult to debug; visual aids and verbose error messages may be very helpful. Here are a list of `Firefox` extensions which will make your life easier. Most of it is particularly useful to a beginner who wants to inspect what HTML, CSS, and JavaScript are used to create features on existing sites.





# Website Architecture

Memorandum

Recommended Software

## Firebug

This is nearly a comprehensive client-side-developer toolkit. Among many other things, it allows you to click on an element in the rendered page and see its HTML and CSS. It also allows you to edit code on the fly. And its network monitor will show you the HTTP requests made to the server for the content items of each page (images, scripts, etc.), along with their HTTP responses. More than that, it shows you a waterfall view of the time taken for each of the requests. This can be invaluable for optimizing large sites.

This does require Firefox 3.5, so you may have to mess with some installation or upgrading.

**Firefox Add-on:** [Firebug](#)

## View Source Chart

After you have changed your HTML (or, more accurately, the DOM) using JavaScript, this allows you to view the DOM's HTML code. This is a great tool to diagnose what JavaScript is really writing to your DOM.

**Firefox Add-on:** [View Source Chart](#)

## Total Validator

This error-checks your client-side languages, and most importantly, its advanced options allow you to request a screenshot of a site from many different browsers, including ones that are almost outdated.

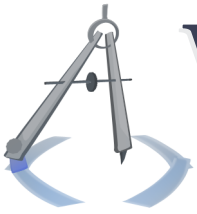
**Firefox Add-on:** [Total Validator](#)

## JSONView

If you use JSON to communicate data objects with Ajax, you can view the objects in a pretty and syntax-highlighted fashion.

**Firefox Add-on:** [JSONView](#)





# Website Architecture

Memorandum

Recommended Software

## File-transfer client

You will have to connect to your site in order to transfer files and update its content. To do this, you need an FTP program. **FileZilla** is a pretty-good-enough client, which is free and is available on multiple platforms. There is even a portable version, which runs in Windows without needing to be installed.

**Official site:** [FileZilla](#)

**Portable version (Windows):** [FileZilla at PortableApps](#)

## Domain names and hosting

When putting a site on the Internet, you generally need to deal with two types of companies: a domain-name registrar and a host.

### Introduction

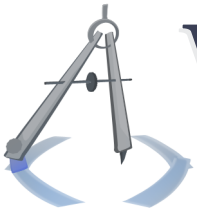
The hosting company holds your site data and distributes it to the world. It runs all the server software mentioned above. However, it does not provide an easily-accessible address to the rest of the world - just an IP address (e.g. *http://74.125.65.105*). The domain-name registrar is the one who links a domain name (e.g. *www.google.com*) with the IP address.

In actuality, the host provides a set of **nameservers** which map all of the domain names it hosts to the appropriate server in its server farm and then the appropriate members' folders (i.e. yours). The domain-name registrar points the domain name to the nameservers.

### Costs

Domain names generally go for around \$10/year, and "shared" hosting goes for about \$5/month if you buy a long-enough contract (perhaps \$6-\$7/month otherwise). A shared hosting account is adequate for the vast majority of sites. In a shared account, multiple websites are run on the same machine. This also implies something about the business model, in that shared machines cannot be customized by the users, and the hosting company can offer extremely low prices to the user, likely due to using the banker's algorithm and buying in bulk.





# Website Architecture

Memorandum

Recommended Software

## Signing up

You should generally buy any domain names first, since they are scarce, and you should buy them all at once, since you will save on bulk pricing. Then you can sign up for hosting and then link them via the nameservers; that will pose the greatest delay, which is probably within 24-48 hours.

When you register your domain, you should opt for "private registration"; otherwise, you will be publicly listed as the owner of the domain, along with your contact information.

Once you get hosting, the company will email you with details on how to connect to their webspace (via an FTP program) and how to change the nameservers from the perspective of your domain-name registrant. The hosting company will at least provide you with the nameservers to use, and the domain-name registrant will have options in their control panel to change the nameservers.

## Companies

These are companies I have used many times, all with good experiences. There are links to them on my website: <http://www.MikeSerritella.com>.

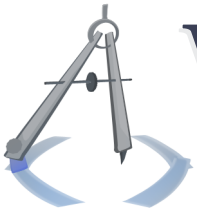
### Domains: Go Daddy

There isn't much to say that hasn't already been said; they are the *de facto* standard. However, I can offer some personal experiences and some advice.

I have had very good experiences (perhaps mostly lucky experiences) with their domain backordering and auction services. I have acquired a few domains on backorder. On one, in particular, the seller put it up for auction on Go Daddy, and, since no one bid, Go Daddy put down a starting bid on my behalf (for free) and gave me the domain. I was impressed. I have also had good experiences with their customer service, wherein I had to cancel an old domain that auto-renewed after I had forgotten about it.

If you buy in bulk (currently 6 domains or more), you get bulk pricing, which is some percentage off, and you get free private registration. Otherwise, private registration costs at least as much as a *.com* domain. An additional trick is that currently, *.info* domains are very cheap - less than \$1/year. So you really might as well buy in bulk and fill any space with *.infos*. I think the price is only for one-year contracts, so watch out when you finalize your cart.





# Website Architecture

Memorandum

Recommended Software

## Hosting: Lunarpages

Lunarpages is the best hosting company I have used. They are just shy of perfect. In their basic, shared hosting plan, they offer you unlimited quantities of anything that they can. Storage is unlimited, bandwidth is unlimited, email addresses are unlimited, and so are services that they call Parked Domains, Additional Domains, and Subdomains. With Parked Domains, if you register *www.ABC.com*, you can point *www.DEF.com* to the same location, transparent to the user. With Additional Domains, you can make a folder in your webspace (e.g. *www.ABC.com/some\_l33t\_hidden\_name*) and then direct *www.XYZ.com* and *www.ETC.com* to that folder. The user will not know the difference. And Subdomains are the same as in common parlance - you can create *something.ABC.com*, which is directed to *www.ABC.com/something*, which is also transparent to the user.

Their customer service is extremely quick and helpful, and their hosting service is quite satisfactory and reliable.

They do have their faults. For one, they don't currently run SQLite v3 on their shared accounts. That is a bit of a drag but will be fixed with time. And if there is some feature you need which they offer on a different shared server, they will move you. More important, however, is their password-management system. After you sign up, they email you your password as part of your welcome email, which is godawful. It means that they might store your password unencrypted, as well, although I can't be sure. I've never had a problem in the sense of getting hacked, and realistically, I don't expect to have that problem, but the fact that they email it to you has its own problems. Anyone who reads (or administrates) your email can read your password, and so can anyone on the Internet who helped deliver that packet to your email server. However, at least when you change your password, they don't email it to you, so you can sign up with a bogus one at first.

## Architectural tools & frameworks

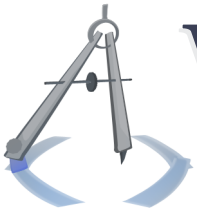
I have made some tools which vastly simplify the site-building process and allow you to focus on software engineering. They also perform very well and produce excellent-quality sites.

Both are available from my website: <http://www.MikeSerritella.com>.

## The HTTPPS

A suite of pretty printers which compose the (X)HTML of a page by managing the opening & closing of tags and indentation. When used properly, this can ensure valid markup, which aesthetically





# Website Architecture

Memorandum

Recommended Software

looks as good as anything out there. The HTTPPS also chooses the proper language (HTML vs. XHTML) based on the user agent and writes the proper MIME type in the HTTP response. Its chief benefit to an architect is its facilitation of modular design, similar to the `ToString()` method in modern object-oriented languages.

## MiniArc

A basic site framework for PHP which provides the HTTPPS, DBMS interface, data validation, and basic debugging tools. Each of the modules is independent, and the use of the framework, in whole or in part, should not complicate the architecture of the site software. The downloadable package comes with a small example site which uses the framework.

