

Accelerating Wordpress for Pagerank and Profit

Practical tips and tricks to increase the speed of your site, improve conversions and climb the search rankings

By: Allan Jude

November 2011

Vice President

ScaleEngine.com

Scalable Infrastructure, Hosting and CDN

Speed Is Everything

- Amazon research found that every 100ms delay costs $> 1\%$ of sales [Make Data Useful – Amazon 2006]
- Google found that even delays of less than half a second had a detrimental effect on the number of searches performed per day [The Impact of Server Delays – Google & Amazon 2009]
- Google has since incorporated speed as one of the vectors in the pagerank algorithm, giving sites that load faster the advantage over sites of similar relevance [Using site speed in pagerank – Google 2010]

Actually Making Your Site Faster

- Use a CDN for Images, CSS, Javascript
- Use an Opcode cache for PHP
- Use a database and object cache if possible
- Serve external files and media from a cookieless domain to increase cacheability
- Leverage users' browser cache to speed subsequent page views and repeat visits
- Use GoogleAPI hosting for things like jQuery
- Reduce the number of external files by combining CSS and JS files and by using Image Sprites

Don't Use Shared Hosting

Dedicated Server

- Full Administrative rights
- All resources are dedicated to your site(s)
- Increased security
- Dedicated caches with custom expiration settings
- Allows the use of nonstandard web server configurations
- Requires Unix skills or an Administrator

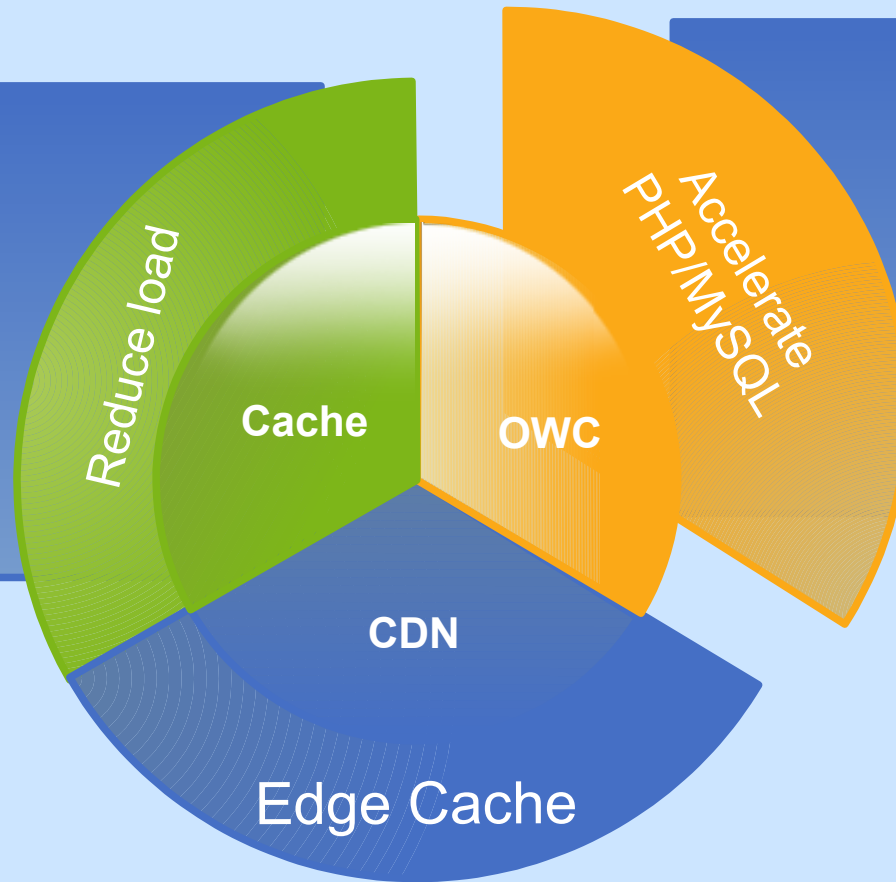
Shared Server

- No Administrative rights
- Limited resources must be shared by all sites
- Shared environment increases chance of a problem with another site effecting yours
- No control over caches, other sites will push your content out of the cache
- Instant provisioning

A shared environment, by definition, means that someone else's busy site will adversely affect your site. Overloaded virtual servers with hundreds of sites cannot sustain bursts of valuable traffic. Network and disk throughput are often poor to terrible with shared hosting, failing to reach even the basic 5Mbps download speed for the average US internet user. Many power users now have download averages in excess of 30Mbps. Your hosting and CDN needs to be on 1Gbps lines for your target market of well provisioned end users

How ScaleEngine Works

Use the database less by caching results, store compiled PHP in an Opcode Cache, and render cacheable content



Generate the page more quickly by distributing the load of both PHP and MySQL across a cluster. ScaleEngine Origin Web Cluster can run most off the shelf apps without any customization

CDN and Edge Cache

- A CDN or Content Distribution Network is a series of servers designed to deliver your content to visitors as quickly as possible
- Edge Cache takes this concept one step further: rather than serving the content from disk as usual, it keeps the most commonly accessed content in memory, allowing it to be served even faster
- The CDN also applies the expire headers and other logic to reduce the request load
- CDN servers are usually geographically distributed, and use some system to route users to the newest, fastest or least loaded node

The Evils of Wordpress

- Avoid plugins written by amateurs at all costs
 - We had a customer whose wordpress plugin did a non-atomic update to a 'hit counter' every time someone visited a page. This nearly crippled the database server when the site got busy
- Reduce the number of plugins. With less code to execute, the site will load more quickly
- Database searching is slow, if you have search volume, implement full text indexing using Sphinx or Solr
- If you receive a lot of comments, it may be advantageous to convert your comments table to a different MySQL storage engine

W3 Total Cache

- The basic configuration will usually upgrade your ySlow & PageSpeed ratings to an A
- Will apply Expires headers to files stored in the user's browser cache, eliminating useless and costly 304 requests
 - Without these headers, each time a user loads your page their browser will re-query each object to see if it has changed! Expires headers tell the browser to not check again until after the expiration date
- Progressive Rendering: your page will start being displayed once the base html and css has loaded; this gives the appearance of a faster site

W3TC to the Rescue

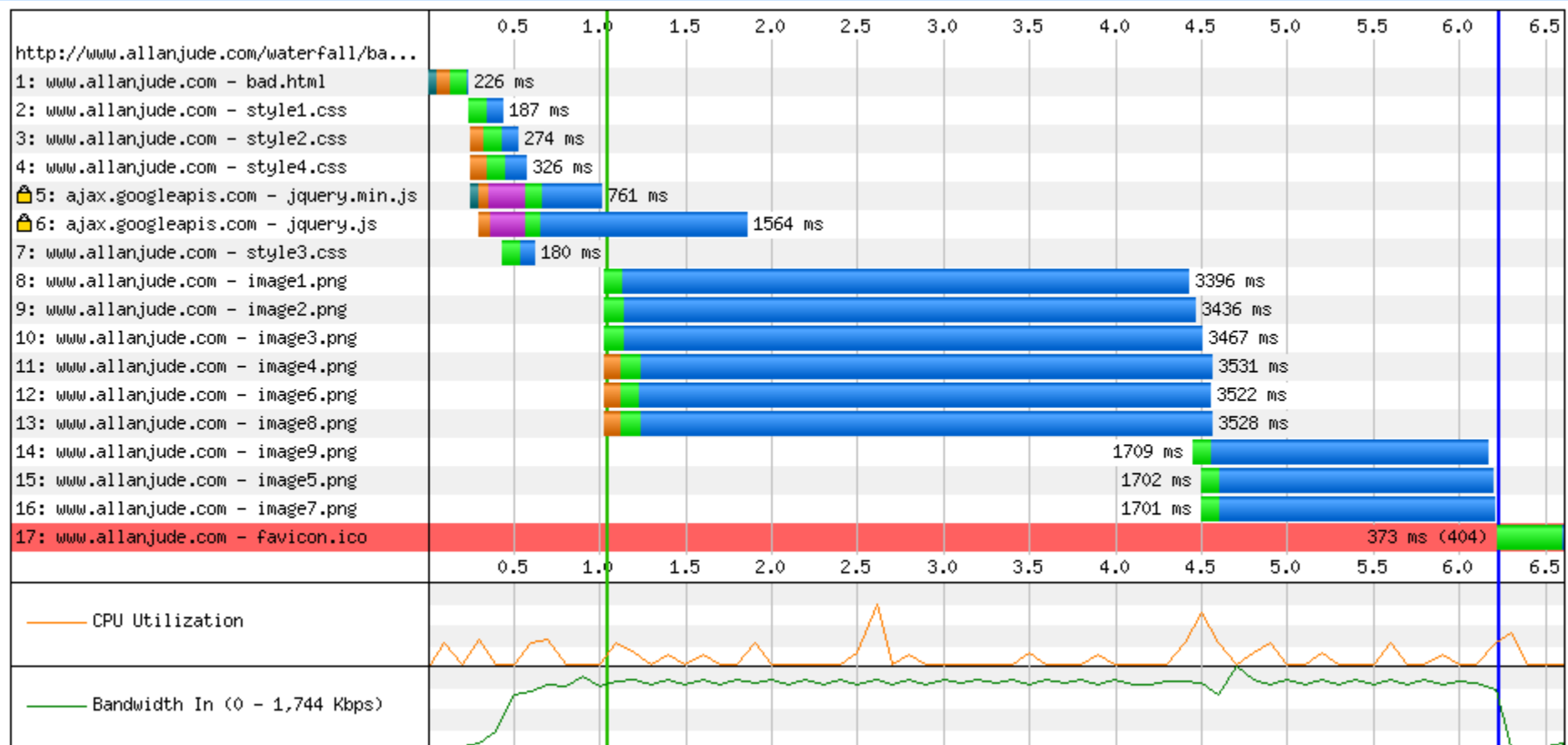
- Will minify and combine your CSS and JS files, reducing the number of separate objects. Also allows you to group JS files into the “Head”, as well as the Top and Bottom of the page to avoid load-order issues
- Transparently rewrites media (images, audio) and externally CSS/JS URLs to use the CDN of your choice
- Easy configuration of Opcode, Database and Object Cache, including automatic purging on edit
- Can generate static html pages to allow no-render responses to popular content (overridden by logged in users, etc.)

Stop Getting Dragged Down

- Advertising networks and Social Media plugins are often the slowest loading elements on your site
- Widgets can be loaded 'asynchronously' so they do not block other components, however this still delays the 'on load' event which blocks many other scripts
- Try to load non-critical elements after the page has been rendered. No one is going to 'like' your page in the first 100ms, but they will leave if the page takes too long to load
- Even Google Analytics should wait to fire until the page is completely loaded, to prevent it delaying the execution of your own javascript code

The Waterfall

A waterfall diagram shows how the objects on your site are loaded. Browsers follow a number of standard practices that are predictable, and if your site is not designed to avoid them, they can adversely affect how long your page takes to load. The most important of these rules is the limit of connections per domain. As you can see in the diagram below, the loading of some of the images is delayed until the first batch of images are loaded.



CDN to the Rescue

In this waterfall diagram you can see that by offloading the images to a CDN and using multiple hostnames, all of the content loads concurrently. This results in a much quicker page load, better bandwidth utilization for the user and a more responsive feel to the website. This technique also takes better advantage of HTTP pipelining and keepalive.



Not Just Wordpress

- These same techniques can be applied to almost any site on any platform
- While it is easier to adjust sites that have a modular architecture, it is still possible to accelerate a legacy site or web application with experienced developer attention
- Some platforms like Wordpress and Drupal have prebuilt plugins to simplify some of this process, while others do not. Use expertise, whether acquired or hired.
- Even closed proprietary applications can be accelerated using custom middleware solutions to adjust the output before it reaches the user

Hosting Technique Overview

- You need to distribute and optimize your site's "objects", from the base html, to the scripts and css, to the images and video
- Reduce waste by combining and minifying CSS/JS, gzipping text and compressing images
- Insist on redundant Tier1 high quality network transit
- Find a hosting company with properly tuned memory, disks, storage, application, database and web servers
- The only way to tell if your site loads quickly and properly, is to test and measure constantly
- If you don't know, ask!

You Don't Get What You Pay For, You Get What You Get!

- Clustering and pooling of server resources can dramatically expand capacity and throughput
- Is the PHP engine distributed on multiple servers for maximum horsepower?
- Are PHP processes isolated from web server processes?
- Are web server processes optimized for concurrency and maximum throughput?
- Can your host intelligently proxy requests to maximize efficient delivery of content and leverage existing site logic?