

Title:

High Performance Web Applications - Part 1

Word Count:

496

Summary:

Today most developers look at the performance considerations after launching the application in Production which is primary reason failures of lot of applications. Performance requirements should be considered from the initial design all the way to production launch. This series is going to describe how high performance can be achieved on the client side (Browsers), Server Side (App Server) and on the DB.

This first part will give some basic tips on improving client side t...

Keywords:

articles,high performance,web tuning,CSS,JS tuning,Expires Header,response time,fast response

Article Body:

Today most developers look at the performance considerations after launching the application in Production which is primary reason failures of lot of applications. Performance requirements should be considered from the initial design all the way to production launch. This series is going to describe how high performance can be achieved on the client side (Browsers), Server Side (App Server) and on the DB.

This first part will give some basic tips on improving client side tuning:

1. Reduce the round trips to the server by combining multiple JS files into one JS file.
2. Reduce the round trips to the server by combining the multiple CSS files to single CSS file.
3. Set Expires header attribute on everything you can into future based on your needs. This tells the browser it is okay to not revalidate on every request, which can add latency of at least one round-trip per object per page load for no reason. Particularly ensure directory where images are stored has Expires Header attributes set to enable the Browser caching. Just by reducing the 304 status

calls to your web server would minimize the load on the web server and improves the page response time.

4. Externalize the JS code to separate JS file and include it and ensure JS files directory on the web server has caching turned on. You need to change the name of file if you make any changes else browser won't download the latest JS files.

5. Make use of CSS attributes instead of using STYLE attribute in the individual HTML elements. Browsers renders pages faster with CSS class attributes than STYLE attributes.

6. Minimize the HTTP response size by enabling GZIP or any compression on the web server. Most modern supports GZIP compression and this reduces the page load time dramatically.

7. You can also improve the response time by adding parallelism to your content. Most browsers limits two concurrent connections to the single host by creating CNAMEs to the same domain you can make use of todays high bandwidth and reduce the response time. Its not advised to have more than 4 aliases to the same domain name. Having more than four aliases/CNAMEs create undesired effects.

8. Minimize the number of domains referenced in the web page. Each host name adds the overhead of an extra DNS entry lookup and an extra TCP three-way handshake. So too many domains referenced in the same page may cause an unexpected delay in the response time.

9. Minimize HTTP request size. Often cookies are set domain-wide, which means they are also unnecessarily sent by the browser with every image request from within that domain.

By implementing the above tips mentioned which doesn't involve any code changes and just by making these environmental related changes you can improve the response time of your pages dramatically and enhance the user's perception about your site.

Today's AJAX based web applications has become quite common and in the next part we will discuss about tips to improve the performance of your JavaScript code.