Three Key Principles to Accelerating Web Applications

Robert Haynes, June 2014

Most organizations are concerned with the speed of their applications. Whether it involves an internal business application or a customer-facing shopping site, web application performance can have a huge impact on business performance. A faster-performing web application results in greater sales, better user engagement, and improved productivity.

For the purposes of this article, we'll use the term "web application" to refer to any application rendered primarily using HTML and delivered from the origin server using HTTP/HTTPS. This could include a public website or an internal business application. The aim of accelerating these web applications is to improve the end-user experience with faster page-load times and more responsive interactivity.

There are three essential principles driving web application acceleration technologies today:

- Send data as efficiently as possible
- Send data as infrequently as possible
- Send as little data as possible

Send data as efficiently as possible

Regardless of any optimization and compression techniques that are used, data must still be sent to and from the client and the server over the network. Sending this data as efficiently as possible can make significant improvements in the end-user experience. Data transfer can be optimized both at the transport layer through TCP optimization and at the application layer by optimizing HTTP requests using the SPDY protocol.

Optimizing the TCP connection for the client network conditions can produce impressive results. In tests using mobile 3G and 4G networks, we have found that page load times decrease an average of 7 percent to 29 percent between various different locations simply by optimizing the TCP connection separately for clients and servers.

The SPDY protocol, which is the starting point of HTTP 2.0, manipulates HTTP traffic with the goal of reducing page-load times and increasing security (SSL/TLS encryption is required to use the protocol). Using SPDY has been shown to improve page-load times due to the more efficient transfer mechanism of the SPDY protocol, which multiplexes multiple HTTP requests on a single TCP connection. While the measured page-load improvements vary, nearly all studies show a worthwhile improvement in page-load times using SPDY.

Send data as infrequently as possible

The fewer network connections and requests a client has to make to assemble a page, the faster it will load. It is possible to minimize the number of requests and connections using techniques such as content inlining (for example, embedding the data to create an image in the HTML page rather than as an external link) and browser cache manipulation.

Intelligent Browser Referencing (IBR) is an example of browser cache manipulation. IBR implements object versioning for cachable objects, such as images and scripts. Versioned objects provide the benefits of caching at the browser while still ensuring that content is not stale. Caching objects for an extended time removes the need for the browser to revalidate content. HTTP requests that simply check if content is still valid can add significantly to page-load times, even if no new data is transferred from the server.

Send as little data as possible

Reducing the amount of data that the server needs to send to the client will result in a reduced network transfer time and faster page load. Compressing redundant data before it's sent or eliminating redundant data from the server responses themselves are methods that can achieve this.

Web application server software or a third-party device such as an Application Delivery Controller (ADC) can compress HTTP objects. Moving HTTP object compression to the ADC or other optimization device allows additional content manipulation and potentially more efficient compression. One of the advantages of using the SPDY protocol is the compression of the HTTP headers, which provides an incremental reduction in the amount of data transferred.

Transforming images into a more efficient file format and stripping image metadata are ways to remove unnecessary data from server responses. Another method to is to remove whitespace or comments from text files, such as JavaScript or CSS. While the gains in removing whitespace from individual files might be only small, the incremental impact is measurable, given the number of objects commonly required to load a page.

Choosing the right acceleration method

How can organizations implement these acceleration techniques? How should they decide which method will be the best for them? Like any other project, when assessing acceleration, organizations need to evaluate the return on investment (ROI) of any solution. A three-day project that results in an 8 percent performance increase might produce significantly more ROI than an eight-week application recoding project that results in a 20 percent performance increase.

Different acceleration solutions use different combinations of these techniques, incurring a wide range of costs and producing varied results. For organizations that are concerned with the potential cost and complexity of deploying acceleration technologies, the next step is to explore specific acceleration implementations to determine the best fit to meet the organization's goals at an acceptable cost.

F5 Networks, Inc. 401 Elliott Avenue West, Seattle, WA 98119 888-882-4447 www.f5.com

F5 Networks, Inc.
Corporate Headquarters
info@f5.com

F5 Networks
Asia-Pacific
apacinfo@f5.com

F5 Networks Ltd.
Europe/Middle-East/Africa
emeainfo@f5.com

F5 Networks
Japan K.K.
f5j-info@f5.com

Solutions for an application world.