



courtesy of
F5 NETWORKS

f.y.i. F5 | guide 4

Improving Web Application Response Time
for Remote and Mobile Users

application
acceleration



Improving Web Application Response Time with Application Acceleration Technologies

Organizations are experiencing two competing trends:

- Business-critical applications with highly dynamic content are becoming web-based
- Users increasingly access these applications from remote and mobile sources

These trends could work in tandem to increase productivity and profitability, but latency has crippled the applications and the business transactions that depend on them. Using the web application or downloading content is often so slow that the applications are useless in remote offices and for mobile workers.

This guide provides an overview on the issues that affect application performance for remote and mobile users, and some possible solutions.

Latency Is Killing Performance

Latency is the largest cause of slow web applications over the WAN or internet, and two culprits are to blame:

Physical distance: It's simple physics—a packet starting at the data center will get to a web browser 20 yards down the hall faster than it will to a web browser on the other side of the world. Even in a hypothetical situation where a packet is traversing an ideal connection over an uninterrupted fiber optic cable to its destination, there will still be a 200 ms (1/4 of a second) delay before it reaches its destination halfway around the earth. Generally, to transfer one object via HTTP, it takes many packets (10 to hundreds).

Network interference: In the real world, most data packets flow over wires (not fiber optic cable) and encounter many switches, routers, and other network devices, each adding a few milliseconds to the journey. If a device is busy, the latency increases. If the connections are low bandwidth and congested, more delay is added to the round trip. As organizations look to send content over mobile and satellite networks, the packet is delayed even more.

Possible Solutions

Web application delays caused by network latency cannot be fixed by increasing bandwidth or increasing server capacity. F5 web acceleration technology solves this problem with its Internal Browser Referencing (IBR) technology, which is a group of capabilities that eliminates the need for the browser to download repetitive or duplicate data, and ensures the best use of bandwidth by controlling browser behavior. By reducing the extra conditional requests and excess data (re)transmitted between the browser and the web application, IBR mitigates the effects of WAN latency, networking errors, and packet loss. IBR also significantly reduces the amount of data downloaded without requiring java applets or making changes to the browser. This is important, because it means IBR is a truly transparent technology. In comparison, other solutions that use applets destroy an organization's ability to troubleshoot a site.

Until recently, web acceleration also usually meant caching static objects in a reverse proxy cache, offloading SSL from busy web servers to a network device, and offloading compression from the web servers to a network device. These types of

A better approach is what's called Intelligent Browser Referencing (IBR), unique to F5

acceleration will improve the time to generate the HTTP response if the web servers are overloaded, but will not improve delivery time over the network.

Intelligent Browser Referencing: Dynamic Content Control

F5's web acceleration technology changes the entire acceleration paradigm by safely caching content at the browser. Subsequent browser requests get the majority of their content from their own cache without the risk of using outdated objects. Use of bandwidth across the wide area network drops dramatically and latency is altogether eliminated, since the browser is not making the request to the server (which avoids the subsequent network roundtrip).

How It Works

The main reason web applications run slowly is the inherent chattiness of the HTTP protocol. The best way to combat this is to cache as many HTTP objects as possible at the browser. This eliminates the HTTP protocol's chattiness and makes the web application much less sensitive to network latency.

Unfortunately, the most common way to cache objects at the browser is to simply set long expiration times in the cache control headers of the objects. This is the approach used by many web development consultants, application server platforms, and other technology companies. A better approach is Intelligent Browser Referencing (IBR), a group of technologies unique to F5.

F5's Approach: Intelligent Browser Referencing

F5's WebAccelerator (WA) also sets long object expiration dates, but does so intelligently. WebAccelerator goes a step further by rewriting the object's URL to contain a checksum of the object's contents. At the same time, it maintains or establishes a short lifetime setting for the HTML page that references the objects.

When the browser wants to see if there is updated content, it re-requests the main HTML page from WebAccelerator since that page has a short expiration time. If nothing has changed, the browser receives the same main HTML page as before and then loads all objects from its cache since they were served with a long expiration time. However, if an object has changed, it is assigned a new object URL, which is reflected in the main HTML page. The browser receives a new HTML page with a different URL only for the object that has changed. The browser requests only that single object and then loads the remaining objects from its cache since they were served with a long expiration time.

Key Advantages

This process has two very important advantages over setting long expiration times alone:

Content safety: Because the URLs have been altered to contain a checksum of the object, there is now a means of retroactively expiring updated content. HTTP objects can be safely given lifetimes as long as six months. If the content is updated, the browser will detect a new URL immediately and begin using the new information. This means the browser is never at risk for using stale content.

Ease of configuration: With no risk of using stale content, there is no need to worry about categorizing the HTTP objects of the web application into static and dynamic data. This allows the use of a very simple configuration such as simply looking for all mime-types that match images. There is no longer a need to have any knowledge of the structure or usage of the web application, or to consider how long to set the lifetime for the different classes of objects. A broadly applied six-month lifetime setting can be used. Even if the content changes every minute, the new content will instantly be available for all users to download as soon as the change is made.



For additional guides from our User Experience/Acceleration series, go to <http://learn.f5.com>, email us at resources@f5.com, or call 888-88BIGIP (888-882-4447).

Guide 1 – Overview | Guide 2 – Network Application | Guide 3 – Server Offload