Scalable Web Server Design for Distributed Data Management *

Scott M. Baker *Bongki Moon*
Department of Computer Science, University of Arizona, Tucson, AZ 85721
{bakers,bkmoon}@cs.arizona.edu

Extended Abstract

With the explosive popularity of the internet and the world wide web (WWW), there is a rapidly growing need to provide unprecedented access to globally distributed data sources through the internet. Web accessibility will be an essential component of the services that future digital libraries should provide for clients. This need has created a strong demand for database access capability through the internet, and high performance scalable web servers. As most popular web sites are experiencing overload from an increasing number of users accessing the sites at the same time, it is desired that scalable web servers should adapt to the changing access characteristics and should be capable of handling a large number of concurrent requests simultaneously, with reasonable response times and minimal request drop rates.

A collection of web documents may be viewed as a directed graph, where each document is a node and each hyperlink (or image reference) is a directed link from one node to another. If there is a way to distribute this graph amongst many server computers in such a way that the load is evenly distributed despite the dynamically changing web access patterns, then the problem of load balancing, one of the most important issues of creating a distributed web server, has been solved. Our solution will take this *graph-based* approach and will be based on the hypothesis that most web sites only have a few *well-known entry points* (*e.g.,* `www.washingtonpost.com`) from which users start navigating through the site's documents.

The proposed solution is to dynamically modify the web documents to change their hyperlink connectivity, and thereby distributing the document graph adaptively amongst several servers. The dynamic modifications will be performed automatically by the web servers and will require no user intervention. All the well-known entry points will be maintained at the *home servers* where the web documents originate, while less known internal documents may be migrated to alternate server computers which we call *co-op servers* for load balancing purposes. The home servers and co-op servers can serve collectively as a *distributed cooperative web server (DCWS)* for the need of web request processing with great flexibility and scalability.

There may be many possible situations where the distributed cooperative web server can be deployed to handle highly fluctuating web requests. Any stand-alone web server can be supported by several computers connected together by a local area network in the same organization. When the stand-alone web (home) server is overloaded, some of the computers can act as co-op servers by offloading documents from the home server and delivering them on behalf of the home server. For another example, two or more departmental web server machines which work independently in the usual operational mode, can become a distributed cooperative web server; since the relative load may be different on each departmental web server depending on the time of year, project deadlines and so on, any of the lightly loaded servers can be a co-op server for any of the heavily loaded servers. The server machines can be geographically distributed. If an organization runs a number of independent web servers for branches in the east and west coasts of the United States and Asian and European countries, then the DCWS approach enables the web servers to adapt to the changes in geographic distribution of document requests and the changes due to different time zones. It also enables the web servers to take advantage of geographic caching of documents.

The distributed cooperative web server solution poses the following benefits over traditional systems based on packet-level manipulation, or domain name services (DNS) and distributed file systems:

- Network or packet level manipulation is not necessary. There is no entity (such as a router) that needs to touch every packet that is transferred between client and server. This eliminates a significant bottleneck present in traditional systems.
- Instead of implicit load balancing by using custom DNS servers, the cooperating servers make use of the connectivity of hyperlinks to directly control load balancing at the fine-grained level of documents.
- The cooperating servers do not need to be located within the same administrative domain or local area network. They may be geographically distributed and can distribute network traffic over multiple networks.
- Adding a new server is easy, flexible, and cost effective. Any available machine may be added as a cooperating server, without consideration as to the location of the machine relative to other existing servers.

For the design principles of the DCWS system and the detailed issues of its prototype implementation, readers are referred to a full paper of this abstract [1].

References

[1] Scott M. Baker and Bongki Moon. Scalable web server design for distributed data management. Technical Report TR 98-8, University of Arizona, Tucson, AZ 85721, August 1998. http://www.cs.arizona.edu/research/reports.html.

*This work was sponsored in part by National Science Foundation Research Infrastructure program EIA-9500991. The authors assume all responsibility for the contents of the paper.