Background Transfers with Minimal Interference

2001-2002 Faculty Research Award

IBM Austin Center for Advanced Studies
Tivoli Center of Excellence

PI: Mike Dahlin
dahlin@cs.utexas.edu
Dept of Computer Sciences
Univ of Texas at Austin

IBM Technical Sponsor:
Jeff Achtermann
jachterm@us.ibm.com
Tivoli Software, Austin, TX

Summary

We propose to develop and evaluate end-to-end mechanisms to permit background transfers of data without interfering with demand requests at clients, servers, or the network between them.

Motivation

Background transfer of data between clients and servers is an important task. In recent work, we have developed optimal algorithms for prefetching data to replicated storage locations [1,2] and demonstrated their potential effectiveness for web workloads [2]. More generally, pre-positioning data at distributed servers and clients has the potential to improve performance for a number of applications, and important system management functions such as system upgrades and backups. In the future, as network bandwidths increase and latencies do not, background traffic will become increasingly important as it becomes economical to "waste" more network bandwidth to hide latency. In many cases, background transfers may equal or exceed foreground traffic [3].

In order to support large amounts of background traffic without interfering with interactive traffic, we propose to develop a middleware module to schedule and issue large volumes of background transfer requests without interfering with foreground requests.

Technical Challenges

A number of operating systems scheduling techniques appear to provide the technologies needed to prevent background requests from delaying foreground requests at the CPU, network, and disk. Our primary goal here will be to assemble these isolated techniques and to assess (1) their end-to-end effectiveness and (2) their ease of deployment.

Our primary research focus will be on network scheduling. In particular, large-volume background transfers should not interfere with demand network traffic---including both transfers between the hosts doing the background transfer and cross-traffic between other hosts that share bottleneck routers with the background transfer. Ideally, one could modify routers to support different service priority levels. In that case, foreground traffic would never wait for background traffic. Unfortunately, modifying the router infrastructure is not a feasible deployment strategy.

The research question is how closely more easily deployed transport-level or application-level techniques can approximate this optimal goal. In particular, we will examine replacing standard TCP congestion control with (a) replacing loss-based feedback with delay-based feedback to react to cross-traffic more quickly and (b) replacing standard additive-increase, multiplicative-decrease congestion control control with a generalized version of the algorithm that allows a wide-range less aggressive variations of the algorithm [4].

This evaluation will consider deployability as a major constraint. At one extreme, modifying routers can provide near-optimal performance but is not deployable in practice. At the other extreme, using unmodified TCP and application-level heuristics (e.g., sending only at night or rate-limiting) is easy to deploy, but has limited performance and requires significant hand-tuning. We believe there is a significant middle ground ranging from user-level congestion control over UDP transport, new in-kernel TCP implementations deployed at the server but not the client, and new in-kernel TCP implementations deployed at both the client and server.

Our goals are to (1) develop quantitative measures of the "damage" inflicted by prefetch traffic on demand traffic, (2) offer tunable algorithms that place a specified upper bound on their interference, and (3) determine the limits of the benefits that can be achieved as a function of the deployment constraints. For some applications, multicast provides a second strategy for reducing the network resources consumed by background traffic. We will also examine recent efforts that quantify the increased efficiency of multicast versus unicast transfers. We will use these techniques to include multicast as a technique for our tunable background traffic system.

References

[1] "Bandwidth constrained placement in a WAN," A. Venkataramani, Phoebe Weidmann, M. Dahlin. Twentieth ACM Symposium on Principles of Distributed Computing (PODC 2001). August 2001. http://www.cs.utexas.edu/users/dahlin/papers/bw-constrained01.pdf

[2] "Potential costs and benefits of long-term prefetching for content-distribution," A. Venkataramani, P. Yalagandula, R. Kokku, S. Sharif, M. Dahlin. The 2001 Web Cache Workshop. June 2001. http://www.cs.utexas.edu/users/dahlin/papers/lt-pref.pdf

[3] "Web workloads influencing disconnected services access," B. Chandra, MS Thesis. UTCS Techical Report. May 2001. http://www.cs.utexas.edu/users/dahlin/papers/chandraMS.pdf

[4] "Congestion Control," Y. Yang. Ph.D. dissertation. UTCS. August 2001.