Session Based Admission Control: a Mechanism for Web QoS

Ludmila Cherkasova and Peter Phaal
Hewlett-Packard Laboratories
1501 Page Mill Road, Palo Alto, CA 94303
e-mail: {cherkasova,phaal}@hpl.hp.com

As the Internet matures, companies are implementing mission critical Internet applications. These applications provide dynamic content, integrate with databases and offer secure commercial transactions.

Customers are becoming increasingly reliant on these complex business applications for services such as banking, product purchases and stock trading. These new services make greater demands on web servers at a time when traffic is increasing rapidly, making it difficult to ensure adequate level of service.

Evaluation of web server performance generally focuses on achievable throughput and latency for request-based type of workload as a function of traffic load. SpecWeb96 benchmark is an industry standard for measuring Web Servers performance. It is based on generating HTTP requests to retrieve different length files accordingly to a particular distribution. The server performance (throughput) is characterized as a maximum achievable number of connection per second while maintaining the required file mix.

However, commercial applications impose a set of additional, service level expectations. Typically, access to a web service occurs in the form of a *session* consisting of many individual requests. Placing an order through the web site involves further requests relating to selecting a product, providing shipping information, arranging payment agreement and finally receiving a confirmation. So, for a customer trying to place an order, or a retailer trying to make a sale, the real measure of a web server performance is its ability to process the entire sequence of requests needed to complete a transaction.

We introduce a new model of workload based on sessions. Session-based workload gives a new interesting angle to revisit and re-valuate the definition of web server performance. It naturally proposes to measure a server throughput as a number of successfully completed sessions.

Let us consider the situation when a server is processing a load that exceeds its capacity. If a load consists of single, unrelated requests then the server throughput is defined by its maximum capacity, i.e. a maximum number of connections the server can support. Any extra connections will be refused and extra load-requests will be dropped. Thus, once a server has reached its maximum throughput, it will stay there, at a server maximum capacity.

However, if the server runs a session-based workload then a dropped request could occur anywhere in the session. That leads to aborted, incomplete sessions. Using a simulation model, we show that an overloaded web server can experience a severe loss of throughput when measured in completed sessions while still maintaining its throughput measured in requests per second. As an extreme, a web server which seems to be busily satisfying clients requests and working at the edge of its capacity could have wasted its resources on failed sessions and, in fact, not accomplishing any useful work. Statistical analysis of completed sessions reveals that an overloaded web server discriminates against the longer sessions. Our analysis of a retail web site showed that sessions resulting in sales are typically 2-3 times longer than non-sale sessions. Hence discriminating against the longer sessions could significantly impact sales and profitability of the commercial web sites.

Quality of service is a way of describing the end to end performance requirements and conditions that a particular application imposes to be successfully executed. For a web server running a commercial application the following *web quality of service* requirement is crucial:

- a fair chance of completion for any accepted session, independent of session length.

We introduce *session based admission control* as a way to provide a web quality of service guaranties for a server running a session-based workload.

The main goal of a session based admission control is the prevention of web server from overload. An admission control mechanism will accept a new session only when a server has the capacity to process all future requests related to the session, i.e. a server can guarantee the successful session completion. If a server is functioning near its capacity, a new session will be rejected (or redirected to another server if one is available).

We introduce a simple implementation of session based admission control based on server CPU utilization (for more details and results see [CP98]). Deferring a client at the very beginning of their transaction (session) rather than in a middle - is another desirable web quality of service property for an overloaded server. It will minimize an amount of wasted server work.

We show that a web server augmented with session based admission control is able to provide a fair guarantee of completion, for any accepted session, independent of a session length. This provides a predictable and controllable platform for web applications, and is a critical requirement for any e-business.

On May 11, 1998, Hewlett-Packard, as a part of its "How to succeed in E-Business" press event, announced the intrduction of the HP Service Control product [HP-98]. This product deploys the session based admission control mechanism, described in this paper, in order to ensure the high levels of service required to successfully complete commerce transactions on the web.

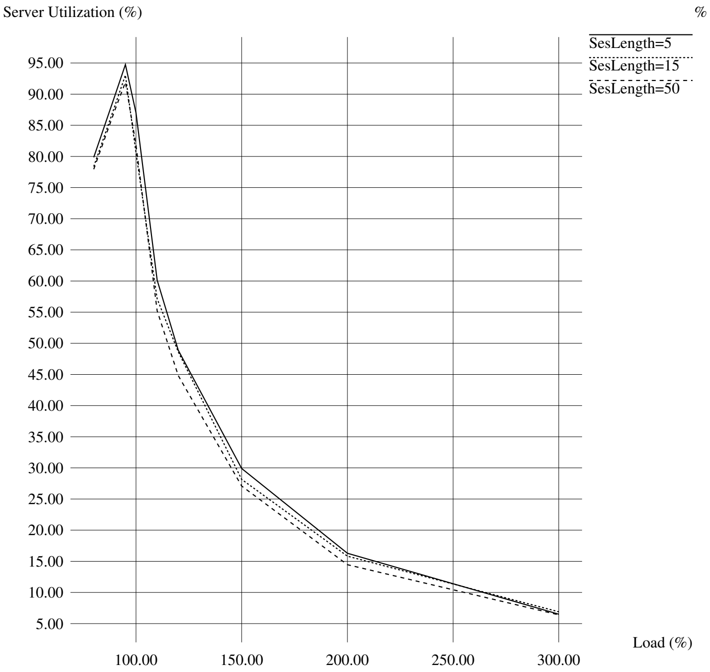

Figure 1: Server Useful Utilization of Processing Sessions which Complete.

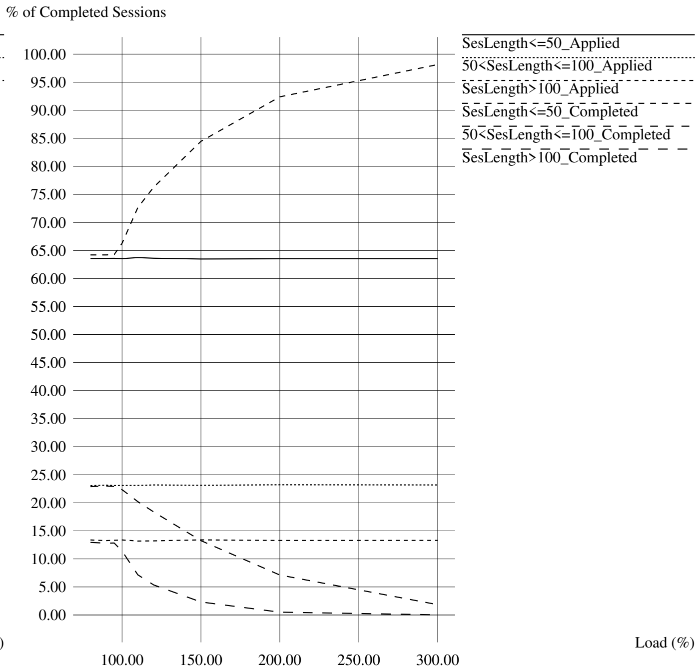

Figure 2: Percentage of Completed Sessions in Three Bins by Length, Session-Based Workload with Mean = 50.

1 Overloaded Web Server: Behavior and Characteristics

We introduce a notion of a session as a unit of session workload. *Session* is a sequence of clients individual requests. The individual requests retrieve the files defined by a SpecWeb96 file mix distribution: 0 class files are 100bytes-900bytes (35%), 1 class files are 1Kb-9Kb (50%), 2 class files are 10Kb-90Kb (14%), 3 class files are 100Kb-900Kb (1%).

A session workload generator produces a new session request accordingly to specified input model parameters: session load and sessions length distribution.We assume the session lengths to be exponentially distributed with a given mean: we have performed experiments for session lengths with a mean of 5, 15 and 50. The client issues the next request only when it receives a reply for the previous request.

For web server running a session-based workload we introduce a new performance measure: *useful server utilization:* server busy time spent processing only sessions which complete. Figure 1 shows useful server utilization as a function of load and session length. The results are overwhelming: the overloaded, "busy looking" server produces an amazingly small amount of useful work: around 15% for a 200% load; less than 7% for a 300% load.

Figure 2 shows the percentage of original and completed sessions in three bins by length for an overloaded server running a session-based workload with a mean of 50.

Indeed, the overloaded web server discriminates against the long sessions in a quite severe way: almost all the completed sessions fall in the first bin (short sessions), the sessions from the second and the third bins (medium and long sessions) are practically absent.

This raises rather serious question: is such server behavior expected and acceptable for commercial sites? Since the answer is rather obvious, the next question to ask is: can a web server be augmented with session based admission control mechanism to prevent the server from becoming overloaded and to ensure that longer sessions are completed?

We introduce a simple admission control mechanism based on the server CPU utilization.

The basic idea of a session based admission controller is as follows: the server utilization is measured during predefined time intervals (say, each second). Using this measured utilization (for the last interval) and some data characterizing server utilization in the resent past, it computes an "observed" utilization. If the observed utilization gets above a specified threshold then for the next time interval (i.e. the next second), the admission controller will reject all the new sessions and will only serve the requests from already admitted sessions. Once the observed utilization drops below the given threshold, the server (controller) changes its policy for the next time interval and begins to admit and process new sessions again.

The admission control mechanism dramatically improves the "quality" of the web server output compared with the similar results for a web server with no admission control. Web server meets the desired quality of service requirement: nearly zero aborted sessions from those accepted for service. It minimizes an amount of wasted server work. For sessions with a mean of 15 and 50 the useful server utilization is improved almost an order of magnitude in overloaded area comparing with the similar results for a web server with no admission control.

References

[CP98] L. Cherkasova, P. Phaal: Session Based Admission Control: a Mechanism for Improving the Performance of an Overloaded Web Server. HP Laboratories Report No. HPL-98-119, June, 1998. URL: http://www.hpl.hp.com/techreports/98/HPL-98-119.html

[HP-98] HP Enterprise Computing. URL http://www.hp.com/esy/go/hp_domain.html