

TPC-W: A Benchmark for E-commerce

Daniel A. Menascé • George Mason University • menasce@cs.gmu.edu

The design of e-commerce sites presents many challenges, including scalability.⁷⁻⁹ One needs to know how the performance—measured in terms of response time and throughput—varies as more and more users access an e-commerce site. Will performance degrade significantly beyond a given load level or will the site be able to deliver acceptable performance even as the load surges? What is the maximum number of transactions that can be processed per second? Can the site be upgraded in a straightforward way (e.g., by adding more servers or replacing existing servers by more powerful ones) to support higher traffic volumes or is an architectural change required? These are some typical questions that must be answered when analyzing the scalability of transactional Web sites.

When choosing the hardware and software configuration of an e-commerce site, one needs to know how a specific combination of Web servers, commerce servers, database servers, and the supporting hardware will handle a desired load level. Benchmarks can be used to compare competing alternatives. The only available benchmark for e-commerce is TPC™ Benchmark W (TPC-W), designed by the Transaction Processing Performance Council (www.tpc.org). This article describes its main features, advantages, and limitations.

Main Components of a Benchmark

The specification of any benchmark deals with three main components as illustrated in Figure 1. The first is the system to be evaluated, called System Under Test (SUT). For TPC-W, the SUT is an e-commerce site. The second aspect of a benchmark is the workload submitted to the SUT. The benchmark has to specify the type of requests submitted to the SUT as well as the intensity of the workload. Finally, a benchmark defines one or more metrics to be obtained by some type of performance monitoring while the SUT is being evaluated. These three aspects of TPC-W are discussed here.

Workload Characterization for E-commerce

The workload of Web sites that provide information has been extensively studied and characterized at the level of HTTP requests.^{1-2, 4-5} Studies have been

conducted with the purpose of understanding the workload of e-commerce sites and searching for invariants that cut across more than one type of e-commerce site.¹⁰ The workload imposed by robots (e.g., price bots and crawlers) was characterized recently and their impact on Web caching was analyzed.³

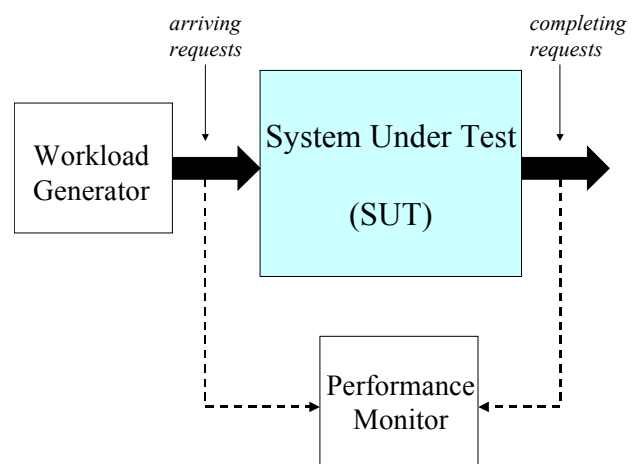


Figure 1 – Components of a Benchmark

In e-commerce, customers interact with the site through *sessions*, which are sequences of consecutive requests to execute e-business functions (e.g., search, browse, select, add to cart, login, register, and pay) during a single visit to the site. One way to capture the navigational pattern within a session is through the Customer Behavior Model Graph (CBMG)^{8,11}, which describes patterns of user behavior, i.e., how users navigate through the site, which functions they use and how often, and the frequency of transitions from one e-business function to another. Figure 2 depicts an example of a CBMG showing that customers may be in several different states—Home, Browse, Search, Select, Add, and Pay—and they may transition between these states as indicated by the arcs connecting them. The numbers on the arcs represent transition probabilities. A state not explicitly represented in the figure is the Exit state. Transitions to this state are indicated by arrows leaving a state and not going to any other state. For example, the probability of going to the Exit state from the Browse state is 0.15.

A slightly modified version of this paper will appear in IEEE Internet Computer, May/June 2002.

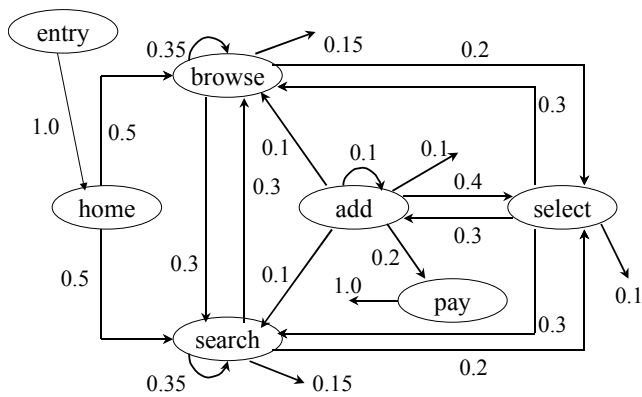


Figure 2 – Example of a Customer Behavior Model Graph (CBMG)

Several patterns of user behavior may be detected at a site. For example, some visits may come from occasional buyers who spend most of their time “window-shopping” but very seldom buy anything. Other visits may exhibit a pattern of heavy buyers, i.e., customers who know what they want and with a few clicks select one or more items, order and pay for them. Different customer behavior patterns generate different loads on the IT resources that support the site.

TPC-W: An Overview

TPC-W mimics the activities of an e-tailer, a bookstore in this case, in a controlled Internet commerce environment. The main features of the benchmark include:

- Multiple on-line browser sessions
- Dynamic page generation with database access and update
- Authentication through Secure Socket Layers (SSL) version 3 or Transport Layer Security (TLS)
- Payment authorization through an emulated Payment Gateway Emulator (PGE), not part of the SUT.
- Databases consisting of many tables with a wide variety of sizes, attributes, and relationships
- Database transaction integrity with ACID property, i.e., atomicity, consistency, isolation, and durability
- On-line transaction execution resulting in contention on data access and update

TPC-W handles scalability by establishing a relationship between the number of concurrent sessions and the size of the store, called the *scale factor*, measured in terms of the number of items in the inventory. The main performance metric specified by TPC-W is Web Interactions Per Second

(WIPS) at a tested scale factor, denoted by WIPS@scale-factor (e.g., 6,045WIPS@100,000).

TPC-W: The SUT

An SUT as specified by TPC-W comprises all Web servers, commerce servers, database servers, load balancers, internal networks, and network interfaces, required to implement the e-commerce application being emulated. Caching products can be used in the SUT, provided that they are commercially available and that data consistency is maintained when caching is used.

The TPC-W database consists of a minimum of eight tables:

- CUSTOMER: Customer personal and session data.
- ADDRESS: Customer address data.
- COUNTRY: Country name and exchange rate information.
- ORDER: Order information, including total amount and shipping information.
- ORDER_LINE: One order line data per order.
- CC_XACTS: Credit card transaction data.
- ITEM: Description of each item in the inventory.
- AUTHOR: Author data.

The use of additional tables required to support shopping cart transactions and state preservation is left to the individual implementation. The entity relationship (ER) diagram in Figure 3 illustrates the logical database design of the TPC-W database.

The database is required to support look-up, insert, and update functionality as well as the capability to commit and rollback transactions.

TPC-W: Workload Generation

TPC-W uses the concept of a group of Emulated Browsers (EB) to generate requests to the SUT. An EB emulates a user communicating with the SUT using a browser by sending and receiving HTML content using HTTP and TCP/IP over a network connection. The number of EBs used for a given test is determined by the size and scaling factor of the SUT, as described later. A Remote Browser Emulator (RBE) creates and manages one EB for each emulated user. The number of EBs is constant throughout an experiment.

Each EB is responsible for submitting a series of requests within a user session. The workload generated by the EBs is specified by i) the navigational patterns within a session, represented by a CBMG, and ii) the workload intensity, specified by the number of EBs and the *think time*,

i.e., the time interval elapsed between a result page is received by an EB and the submission of its next request.

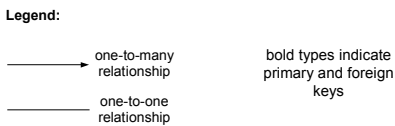
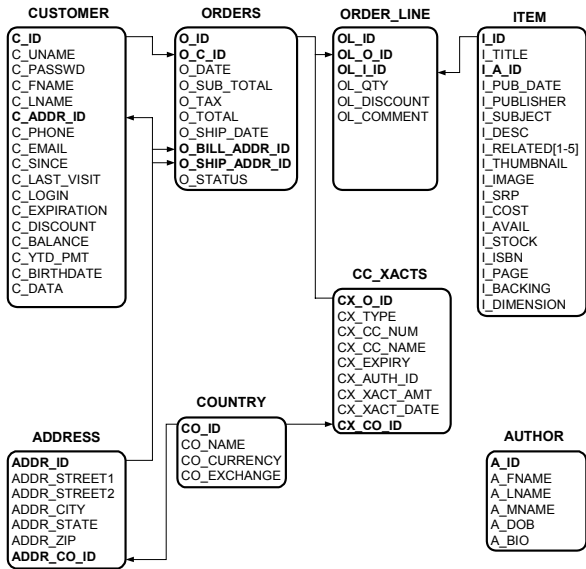


Figure 3 – Entity Relationship Diagram for the TPC-W Database

The full CBMG, specified by TPC-W, consists of 14 unique pages, shown in Table 1. These pages are divided into two categories: Browse and Order. The Browse pages typically consist of non-secure requests with few transaction processing requirements. The Order pages are distinguished from the Browse pages by their greater processing requirements in terms of database access and/or secure transaction processing. Figure 4 shows a slightly simplified version of the CBMG for TPC-W, in which the states Admin Request and Admin Confirm, used by site administrators, were omitted. Transitions to the Exit state (not shown in the figure) can occur from any state except the Entry state.

Transition probabilities are not shown in the CBMG of Figure 4, because TPC-W defines three different types of Web interaction mixes by varying the ratio of browse to buy activities: primarily shopping (WIPS), browsing (WIPsb), and web-based ordering (WIPSo), as indicated in Table 1. The table shows the percentage of browse interactions and product order interactions for the three Web interaction mixes, as well as the breakdown within each category.

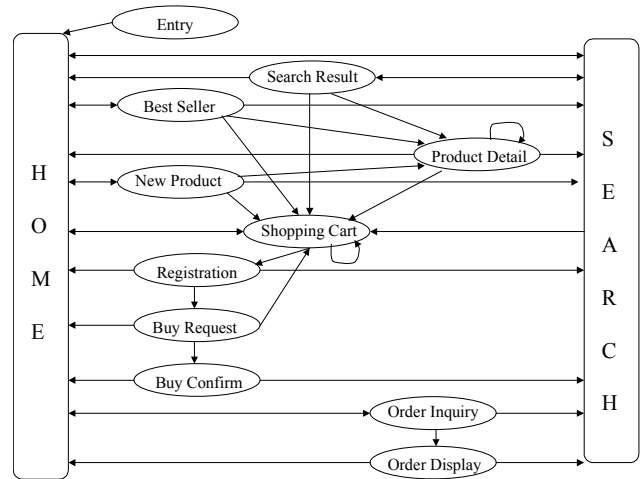


Figure 4 - CBMG for TPC-W

TPC-W specifies the transition probabilities for the CBMG of each of the three Web interaction mixes so that the percentages indicated in Table 1 are enforced.

Table 1 - Web Interaction Mixes

Web Interaction	Browsing Mix (WIPsb)	Shopping Mix (WIPS)	Ordering Mix (WIPSo)
Browse	95%	80%	50%
Home	29.00%	16.00%	9.12%
New products	11.00%	5.00%	0.46%
Best sellers	11.00%	5.00%	0.46%
Product detail	21.00%	17.00%	12.35%
Search request	12.00%	20.00%	14.53%
Search result	11.00%	17.00%	13.08%
Order	5%	20%	50%
Shopping cart	2.00%	11.60%	13.53%
Registration	0.82%	3.00%	12.86%
Buy request	0.75%	2.60%	12.73%
Buy confirm	0.69%	1.20%	10.18%
Order inquiry	0.30%	0.75%	0.25%
Order display	0.25%	0.66%	0.22%
Adm. request	0.10%	0.10%	0.12%
Adm. confirm	0.09%	0.09%	0.11%

A slightly modified version of this paper will appear in IEEE Internet Computer, May/June 2002.

TPC-W defines a user session duration as the time elapsed between the first transaction executed by an EB and the current time. The minimum session duration, called User Session Minimum Duration (USMD), is obtained from an exponential distribution with mean equal to 15 minutes truncated at 60 minutes. At the end of a session, the EB closes any open SSL sessions and TCP connections and starts a new user session. Each EB must generate a new USMD for each new session.

Each user session, is composed of multiple requests to the SUT. These requests are separated by a unique think time, Z , defined as the time elapsed since the last byte of the current page requested is received by the EB (this includes all requests for in-line images) and the time that immediately precedes sending the first byte of the next request to the SUT. Think times generated by EBs are derived from an exponential distribution with a mean between 7 and 8 seconds truncated at 10 times the mean value.

TPC-W Metrics

The two main TPC-W metrics are:

- WIPS: Web Interactions Per Second during a shopping mix. When reporting a value for the WIPS metric, the cardinality of the ITEM table has to be specified. For example, a WIPS value measured for a database with 100,000 rows in the ITEM table would be reported as 6,045@100,000.
- \$/WIPS: the ratio between the total price of the SUT (includes hardware, software, and 3-year hardware/software maintenance charges), and the WIPS value.

The secondary metrics are WIPSB and WIPSO, which measure the number of Web interactions per second during browsing and ordering mixes, respectively. The three Web interaction mixes provide diverse buy to visit ratios and may be used to represent different usage patterns.

Some examples of actual values of the metrics posted on the TPC site as of April 4, 2002 are shown in Table 2, with company names omitted. The table was obtained for databases with 100,000 items and shows that the best system (system A) is able to process 6,045 Web Interactions per Second at the price of \$76.67/WIPS. So, the total cost (software + hardware + maintenance) of System A is \$464 thousand dollars. System D costs about 2.65 times as much as system A and has a performance, measured

in WIPS, which is only 3.8% better than that of system A.

Table 2 – Top Four TPC-W Results for 100,000 Items in the Catalog (as of April 4, 2002)

Rank	Company	WIPS	\$/WIPS
1	A	6,045	76.67 US \$
2	B	10,439	106.73 US \$
3	C	7,554	136.80 US \$
4	D	6,272	195.59 US \$

TPC-W specifies a required 90-percentile for the response time on each of the 14 pages. For example, 90% of the Buy Confirm pages must return in at most 5 seconds whereas 90% of Search Requests must have a response time not exceeding 3 seconds.

Using the Response Time Law⁶ to establish a relationship between the average response time R , the number of Emulated Browsers, and the average think time \bar{Z} , we get

$$R = \frac{\text{No. EBs}}{\text{WIPS}} - \bar{Z}. \quad (1)$$

Using $\bar{Z} = 7$ sec, a possible value specified by TPC-W for the average think time, and considering system A in Table 2, we would get an average response time of 1.27 sec for 50,000 concurrent users ($1.27 = 50,000 / 6,045 - 7$). System D in Table 2 would provide an average response time of 0.97 sec for the same number of concurrent users and the same average think time ($0.97 = 50,000 / 6,262 - 7$).

TPC-W places upper and lower bounds on the value of the WIPS metric that can be reported. Let us examine these bounds by rewriting Eq. (1) as

$$\text{WIPS} = \frac{\text{No. EBs}}{R + \bar{Z}}. \quad (2)$$

The maximum possible theoretical value for the WIPS value is obtained by making the response time equal to zero and \bar{Z} equal to 7, the minimum allowed average think time, in Eq. (2). So,

$$\text{WIPS} < \frac{\text{No. EBs}}{7} \quad (3)$$

On the other hand, if the response time is too high, the value of WIPS will be very small, according to

A slightly modified version of this paper will appear in IEEE Internet Computer, May/June 2002.

Eq. (2). To prevent over-scaling the system SUT, the reported WIPS must be at least 50% of its maximum theoretical value of No. EBs/7. So, the value of WIPS must satisfy the following constraint:

$$(\text{No. EBs} / 14) < \text{WIPS} < (\text{No. EBs} / 7). \quad (4)$$

TPC-W Scalability

TPC-W is designed with scalability in mind. The idea is to maintain the ratio between the transactional load placed on the site, indicated by the number of concurrent EBs, and the initial database size, represented by the cardinality (number of rows) of the various database tables. Figure 5, indicates how the cardinality of each table is obtained as a multiple of the cardinality of another table or, in the case of the CUSTOMER table, as a multiple of the number of EBs. The cardinality of the COUNTRY table is fixed at 92 rows. The cardinality of the ITEM table, TPC-W's scaling factor, can be one of the following values: one thousand, ten thousand, one hundred thousand, one million, and ten million rows.

For example, according to Figure 5, if 100 EBs are used, the CUSTOMER table should have 288,000 rows. The ORDER table should have 259,200 rows and the ADDRESS table should have 576,000 rows. The ORDER_LINE table should have 777,600 rows and the CC_XACTS table should have 259,200 rows. If the database has 100,000 items, then there should be 25,000 authors.

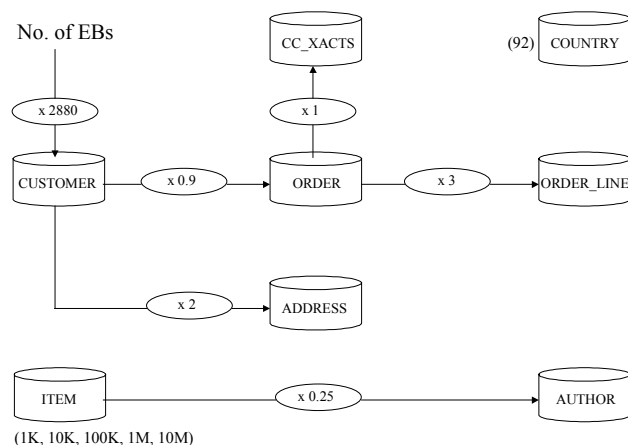


Figure 5 – Database Scalability

The idea behind the scaling method is that as the on-line store supports a higher number of concurrent users, the size of the database that supports the store operation has to scale up accordingly.

TPC-W Limitations

As any benchmark, TPC-W may not accurately represent a specific application. For example, the workload characteristics of an auction site may be much different from that of an online bookstore. One may expect the workload of an auction site to exhibit a surge of requests for a specific item near the closing time of an auction.

TPC-W is oriented towards business-to-consumer e-commerce, and does not model well business-to-business online transactions. Another limitation is that TPC-W is designed for workloads composed of requests generated by human beings who expect HTML pages as a result. As Web services¹² become more prominent, there will be a need for benchmarks of Web service providers, where the users are computer programs and not human beings. Another aspect not contemplated by TPC-W is the presence of robots in the workload of actual e-business sites.³

It should also be realized that TPC-W results are obtained in a controlled internet environment, in which the EBs are typically connected to the SUT via very high-speed (e.g., 1 Gbps) switches. When users access Web sites through WANs and low speed connections, resources at the Web site (e.g., TCP connections and threads) remain tied up for much longer periods of time and the site throughput decreases.

Despite its limitations, TPC-W is a very well designed benchmark with many of the important elements of most e-commerce applications and therefore, should be seriously considered by providers of e-commerce solutions.

References

1. V. Almeida, A. Bestavros, M. Crovella, and A. de Oliveira. "Characterizing Reference Locality in the WWW," *Proc. IEEE Conference on Parallel and Distributed Information Systems*, Miami Beach, FL, December 1996.
2. M. Arlitt, R. Friedrich, and T. Jin, "Workload Characterization of a Web Proxy in a Cable Environment," *ACM Performance Evaluation Review*, 27 (2), Aug. 1999, pp. 25--36.
3. V. Almeida, D. A. Menascé, R. Riedi, F. Ribeiro, R. Fonseca, and W. Meira, Jr., "Analyzing Web Robots and their Impact on Caching," *Proc. Sixth Workshop on Web Caching and Content Distribution*, Boston, MA, June 20-22, 2001.

A slightly modified version of this paper will appear in IEEE Internet Computer, May/June 2002.

4. M. Arlitt and C. Williamson, "Web Server Workload Characterization: the Search for Invariants," *Proc. 1996 ACM Sigmetrics Conf. Measurement & Modeling of Computer Systems*, Philadelphia, PA, May 23-26, 1996, pp. 126—137.

5. M. Crovella and A. Bestavros, "Self-Similarity in World Wide Web Traffic: evidence and possible causes," *Proc. 1996 ACM SIGMETRICS Conf. Measurements Comp. Syst.*, Philadelphia, PA, May 1996.

6. P. J. Denning and J. P. Buzen, "The Operational Analysis of Queuing Network Models," *ACM Computing Surveys*, vol. 10, no. 3, Sept. 1978, pp. 225-261.

7. F. Douglis and M. F. Kaashoek, "Scalable Internet Services," *IEEE Internet Computing*, July/Aug. 2001, pp. 36-37.

8. D. A. Menascé and V. A. F. Almeida, *Scaling for E-Business: technologies, models, performance, and capacity planning*, Prentice Hall, Upper Saddle River, NJ, 2000.

9. D. A. Menascé and V. A. F. Almeida, *Capacity Planning for Web Service: metrics, models, and methods*, Prentice Hall, Upper Saddle River, NJ, 2002.

10. D. A. Menascé, V. Almeida, R. Riedi, F. Ribeiro, R. Fonseca, and W. Meira Jr, "In Search of Invariants for E-Business Workloads," *Proc. Second ACM Conference on Electronic Commerce*, Minneapolis, MN, October 17-20, 2000.

11. D. A. Menascé, V. Almeida, R. Fonseca, and M. A. Mendes, "A Methodology for Workload Characterization of E-commerce Sites," *Proc. First ACM Conference on Electronic Commerce*, Denver, CO, November 3-5, 1999.

12. Web Services Description Language (WSDL), www.w3.org/TR/wsdl