Proc. 2000 Computer Measurement Group Conference, Orlando, FL, December 10-15, 2000.

CHALLENGES IN SCALING E-BUSINESS SITES

Daniel A. Menascé
Department of Computer Science, MS 4A5
George Mason University
Fairfax, VA 22030-4444
Menasce@cs.gmu.edu

Virgilio A. F. Almeida
Department of Computer Science
Federal University of Minas Gerais
Belo Horizonte, MG 3000, Brazil
virgilio@dcc.ufmg.br

One of the challenges in designing and maintaining e-business sites is to ensure its scalability as the workload increases. This paper discusses a multi-layer reference model that can be used for capacity planning and analysis of e-business sites. The paper shows how to characterize the workload of e-commerce servers taking into account customer behavior patterns. It further explains how to use performance models to identify and analyze problems in e-commerce sites. Finally, a discussion on the performance impacts of authentication protocols is given.

1. Introduction

E-commerce sites have been developed and launched with very little time for planning. A large number of these sites go on line in six to eight weeks. Rapid development and deployment of e-commerce sites is important because time is of the essence in the very competitive area of e-business. However, many e-business sites discover right after launch that the number of visitors to the site far exceeds expectations and that the hardware infrastructure and/or software architecture is not adequate to handle the load. Bad performance has shut down many sites at launch time as reported by the general media!

While the managers of these sites have downplayed these incidents as being a good thing that many people went to their sites, the truth is that lousy performance can be embarrassing and bad for the business. In fact, a Zona Research report published in April 1999 found that the US lost $43.5 billion in the previous year in e-commerce due to bad performance. Some statistics about the Holiday season of 1998 are revealing: over 1/3 of customers gave up due to slowness, 44% turned to conventional stores, and 14% moved to another site.

But, what is bad performance in e-business anyway? In traditional mainframe and/or client/server environments, the notion of bad performance is associated to service level agreements (SLAs). When SLAs are not met, the system performance is deemed to be inadequate. In e-business, there are no explicit SLAs. Customers of an online store do not meet managers of the store to reach an agreement on how long they are willing to wait for the execution of each function. When the time to download a Web page exceeds eight seconds, customers tend to become very frustrated. This de-facto standard on Web page download is called the ``eight-second rule.'' Clearly, customers may be willing to wait longer for some type of pages than others. For instance, while customers may abort a search after waiting for its result for more than eight seconds, they may be willing to wait 20 or even 30 seconds for a page that confirms a payment made by credit card.

Performance problems in e-business tend to escalate for many reasons. The first is the proliferation of mobile devices such as Personal Digital Assistants (PDAs) that offer wireless Web access. The same is true for cell phones that are already starting to offer Web access. The reliability, security, and speed of these connections will increase dramatically in the near future as third generation cellular technologies such as **C**ode **D**ivision **M**ultiple **A**ccess **(CDMA)** become more widely used. What this means is that many people will be interacting with e-business sites even when they are away from their desktops. It is estimated that CDMA-based wireless services will

enable more than 500 hundred million people to access multimedia content anytime, anywhere by the year 2005 [DRC00]. Some airplane manufacturers have recently announced that they will be equipping their airplanes with Web and e-mail access. Passengers will be able to interact with e-business sites while traveling over the Atlantic or flying across the US. This means more traffic to e-business sites.

The second factor is the development of easier to use interfaces based on speech recognition. Some car manufacturers have already announced that their cars will come equipped with e-mail and Web access using a voice-based interface. In the near future, you may be able to "ask" your favorite online trading site how some of the stocks in your portfolio are doing and "tell" the site to buy or sell, without taking your hands off the steering wheel. Again, these new technologies will make Web access more ubiquitous and increase traffic to Web and e-business sites.

Another factor to consider is the increased load to be placed on e-business sites by software agents that will buy and sell on behalf of customers [MAES99]. These agents will roam the Web, looking for items in the profile of the customer, negotiate with merchant agents on issues ranging from price, delivery time, and shipping and handling options. Software agents will free customers from having to search and compare available options.

Finally, one has to consider the impact of the use of authentication protocols (e.g., SSL and TLS) and payment protocols (e.g., SET) on e-commerce site performance [MA00]. More detail will be provided in section 7.

This paper discusses issues that are important in answering questions such as:

- Is the online trading site prepared to accommodate a 75% increase of trades/day?
- Do I have enough servers to handle a peak demand 10 times the average value?
- How fast can the site architecture be scaled up? What components should be upgraded? Database servers? Web servers? Application servers? Bandwidth?
- How can I design a site that will meet its business goals?

Section two discusses the notions of scalability. Section three presents a multi-layer reference model that can be used to analyze e-business sites. The next section discusses issues related to workload characterization in e-commerce. Section five discusses some of the issues that are relevant when modeling e-commerce servers and section six discusses performance modeling issues including software contention problems. Section seven addresses the performance impacts of authentication protocols and, finally, section eight presents some concluding remarks.

2. Scalability

What do people mean when they say that a system is scalable? We consider a system to be scalable if there is a "straightforward'' way to upgrade the system to handle an increase in traffic while maintaining adequate performance. By straightforward we mean that no system or software architectural changes should be required to scale the system. Example of straightforward changes are: adding more servers to a system that already employs multiple servers, adding more CPUs to a multiprocessor, replacing existing servers with faster servers that use the same architecture. Devlin, Gray, Laing, and Spix have introduced the notions of scaling out and scaling up [DGLS99]. Scaling out means that more servers of the same type are added, while scaling up means replacing the existing servers with faster ones. Figure 2.1 illustrates these notions. The top part of the figure shows a system with *n* servers, each with a speed of *S* (operations/sec). The middle part of Fig. 2.1 shows a scaled-up version of the original system. This version was obtained by upgrading each of the *n* servers to a greater speed *k* x *S* (*k* > 1). Finally, the bottom part of the figure shows a scaled-out system obtained by adding more of the original servers to the original system.

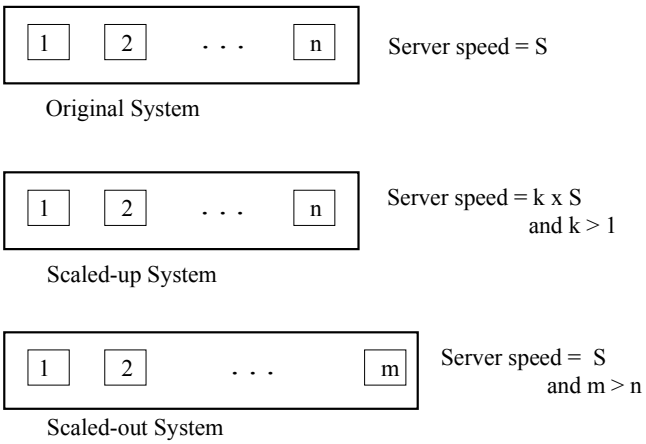


Figure 2.1 – Scaling Up and Scaling Out

Let us compare the performance of these three alternatives using standard M/M/m queuing models [KLEI75] as depicted in Fig. 2.2. As shown in Fig. 2.2, the original system, the scaled up, and the scaled out systems can be modeled as M/M/m queues with arrival rate equal to λ requests/sec and service rate of μ requests/sec. The response time for these three cases can be computed using the formulas shown in "Queuing Systems, Vol I: Theory" [KLEI75] or in "Capacity Planning for Web Performance: metrics, models, and methods" [MA98].

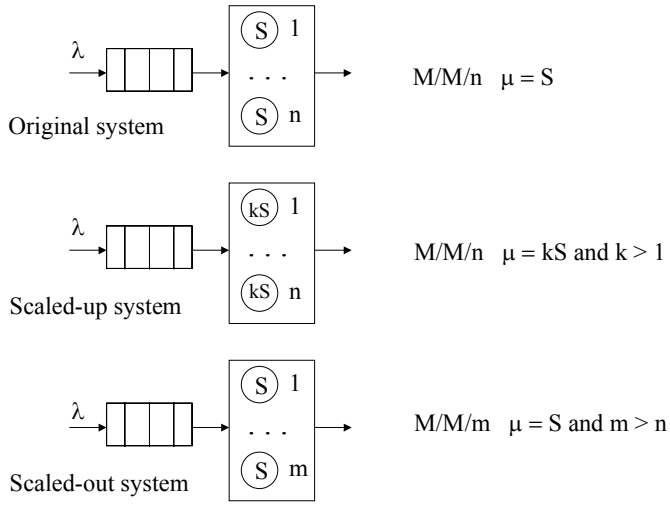


Figure 2.2 – Queuing Models for Scaling Up and Scaling Out.

Figure 2.3 illustrates a numerical example showing the variation of the response time as a function of the arrival rate λ for the case in which the original system has three servers with a processing rate of 2 requests/sec each. So, the aggregate processing rate is 6 requests/sec. Suppose we want to double the aggregate processing capacity. One alternative is the scale up approach. In this case, we double the speed of each of the three servers to 4 requests/sec. Another approach, the scaling-out approach, is to add three more servers with processing rate of 2 requests/sec each. In both scaling-up and scaling-out cases, we have an aggregate processing capacity of 12 requests/sec. However, as shown in Fig. 2.3, the scaling-up approach exhibits a better response time.

Figure 2.3 suggests that if we want to achieve the same level of performance as in the scaling up case using a scaling out approach, we need to add even more servers. In other words, we need to provide an aggregate capacity for the scaling out approach that exceeds that of the scaling up one. In some cases, it may be cheaper to do just that if smaller machines have a better cost x performance ratio than high-end servers.

Scalability is a key issue for e-commerce sites. To assess their scalability, the next section presents a multilevel reference model that can be used as a framework to understanding and analyzing e-commerce sites.

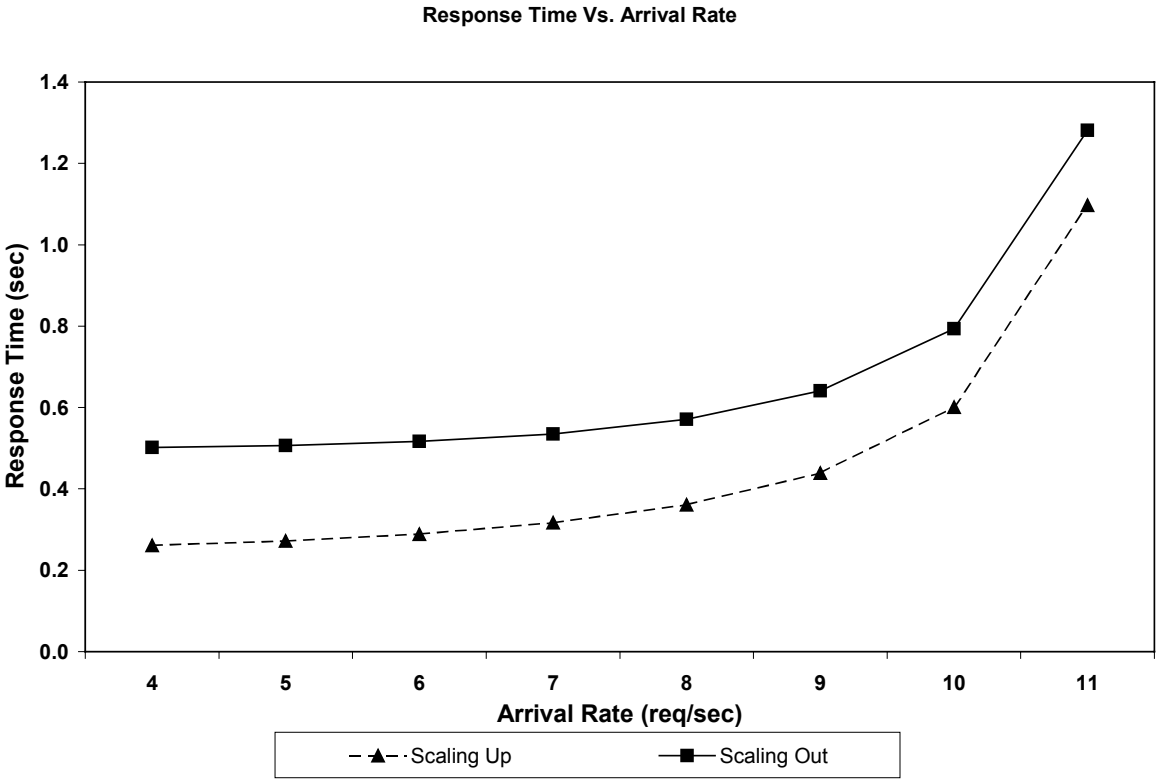


Figure 2.3 - Performance of Scaling-up vs. Scaling-out Approaches.

3. A Reference Model for E-Commerce Analysis

Traditional capacity planning approaches tend to be IT resource-centric. However, in e-business systems, one has to consider all aspects of the problem: business, functional, customer behavior, and IT resources. In this section, a *reference model* is presented that can be used to analyze e-business sites and plan their capacity and scalability properties. This reference model is depicted in Fig. 3.1.

The *business model* describes the nature of the e-business through *business descriptors* such as type of delivery (e.g., electronic, physical), number of items in the catalog, number of registered customers, traffic statistics, supply-chain integration and order-fulfillment processes, type of items sold, sale and revenue statistics, availability, security, and performance requirements.

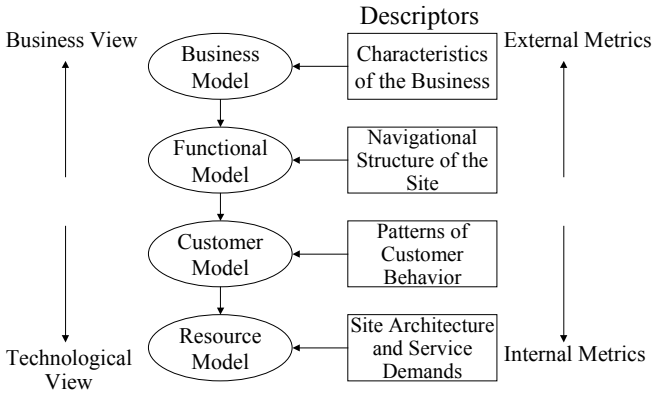


Figure 3.1 – A Reference Model for E-Business. Reprinted from [MA00] with authorization from Prentice Hall.

The *functional model* describes the functions offered by the e-business site as well as its the navigational structure. Examples of e-business functions include Browse, Search, Select Item, Register, Login, Add to Cart, Order Product, and Pay.

The *customer behavior model* describes patterns of user behavior, i.e., how users navigate through the site, which functions they use and how often, what is the frequency of transition from one e-business function to the other, etc. Clearly, several patterns of user behavior may be detected for the same site. For example, a fraction of the visits may come from occasional buyers who spend a lot of time window-shopping but very seldom buy anything. Other visits to the site 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. It is important to realize that different customer behavior patterns generate different loads on the IT resources that support the site.

Finally, the *resource model* describes the hardware and software architecture along with the service demands for every type of transaction at each resource (e.g., processors, I/O devices, network, segments, and routers).

4. Workload Characterization for E-Commerce

The reference model described in section 3 can be utilized as a framework for workload characterization of e-business sites. As the model suggests, workload characterization has to be done at multiple levels. At the highest level, the business level, one needs to establish quantitative business descriptors (QBDs) that can be used to guide the workload characterization process. These QBDs can also be used for workload forecasting and as scalability analysis parameters.

At the functional and customer behavior level, the following steps should be taken:

1. Determine the e-business functions made available by the site. Associate URLs or URL patterns to each e-business function. This allows one to map HTTP requests found in HTTP logs into e-business functions.

2. Analyze the site's HTTP logs to determine *customer sessions.* A customer session is defined as a sequence of related requests made by a customer during a single visit to the site. Session boundaries are usually identified in a log by associating a threshold of time inactivity by a customer. Thirty minutes is a standard threshold value [MA00].

3. Cluster customer sessions into groups of "similar" customer sessions, i.e., sessions that exhibit similar behavior. The notion of similarity is usually based on some form of Euclidean distance as in many clustering algorithms. Examples of such clustering analyses for e-commerce are given in "A Methodology for Workload Characterization of E-commerce Sites" [MAFM99]. The result of these analyses is a set of graphs, called Customer Behavior Model Graphs (CBMGs) that describe patterns of user behavior. Figure 4.1 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 the states. The numbers along the transitions indicate the probability of making the transition. A state not explicit represented in the figure is the Exit state. Transitions to the Exit state are indicated by arrows leaving a state and not going to any other state in the figure. For example, the probability of going to the Exit state out of the Browse state is 0.15. As shown in "Scaling for E-business: technologies, models, performance, and

capacity planning" [MA00] and "A Methodology for Workload Characterization of E-commerce Sites" [MAFM99], one can obtain several interesting metrics from a CBMG. One is the average number of visits V_j to any state *j* of the CBMG. The other interesting metric is the average session length *S*, defined as

$$S = \sum_{j=2}^{n-1} V_j$$

where *n* is the number of states of the CBMG. We assume, without loss of generality, that the entry state is 1 and the exit state is *n*. Finally, the other metric of interest is the buy to visit ratio defined as the average number of visits to the Pay state per visit to the site.

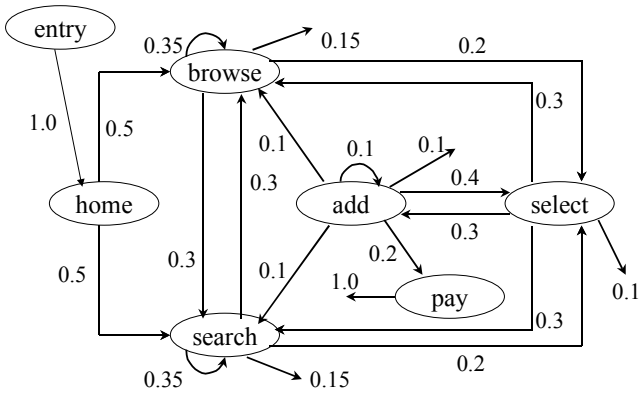
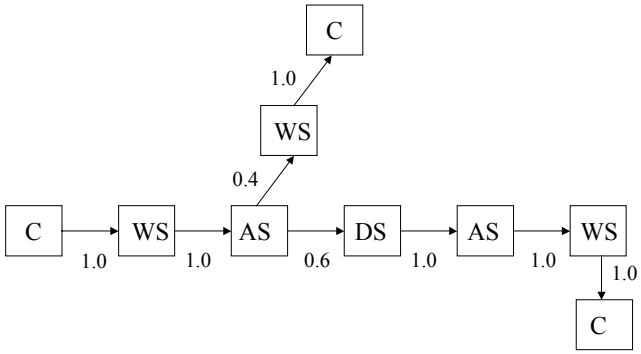


Figure 4.1 – Example of a Customer Behavior Model Graph (CBMG)

4. Map e-business functions into client/server interactions displaying the servers (e.g., Web server, application server, database server) involved in the execution of each interaction. These interactions are described using diagrams such as the one depicted in Fig. 4.2. The various nodes of the diagram correspond to clients or servers. The paths starting at a client node and ending at another client node represent a complete interaction with the site. Probabilities are usually associated with the arcs to indicate the probability of a specific interaction between two elements. For example, Fig. 4.2 indicates the client/server interaction for the Search function. A Search requests starts at the client, goes to a Web server, which sends the request to an Application Server. If the result of the search is cached at the Application Server, this happens with probability 0.4 in the figure, then the Application Server creates an HTML page with the answer and sends it to the Web server, which sends the result to the client. With probability 0.6, the result of the query is not cached at the Application Server, which needs to contact the Database Server to obtain the result of the query.

   These diagrams allow one to obtain the average number of times a server is activated during the execution of an e-business function. For example, the Application Server is activated once with probability 0.4 and twice with probability 0.6. Then on average, it is activated 1.6 (=1 x 0.4 + 2 x 0.6) times. Assume that, based on CBMG analysis, on average, each customer executes 3.5 Search functions per session. Then, the Application Server will be activated 5.6 (= 1.6 * 3.5) times per session. If the site starts 5 session/sec, the Application Server will be activated at a rate of 28 (=5 x 5.6) activations/sec.



C: client
WS: Web Server
AS: Application Server
DS: Database Server

Figure 4.2 – Example of a Client/Server Interaction Diagram for the Search Function

5. Obtain the service demands at each resource (e.g., processor, I/O device) of each of the servers for each activation of the server. These service demands can be obtained using the conventional approaches described in "Capacity Planning for Web Performance: metrics, models, and methods" [MA98] and in "Capacity Planning and Performance Modeling: from mainframes to client-server systems" [MAD94]. Suppose for example that the service demand at the CPU of the Application Server for each activation of the server for the Search function is 0.010 sec. Then, the CPU utilization of the Application Server due to the execution of the Search function alone would be, using the numbers in the previous item, 28% (=0.010 x 28).

Once the workload has been characterized, one can develop performance models to answer what-if questions, as described in the next section.

6. Modeling E-commerce Servers

E-business sites have a multi-tiered architecture as shown in Fig. 6.1. There are typically three layers of servers. The first is composed of Web and authentication servers. Incoming requests are sent to any of these servers based on policies implemented by load balancers. The second tier is composed of transaction or application servers that implement the business logic of the application. Finally, the third tier is where DB servers reside. They can be existing mainframes running legacy applications and storing enterprise data or dedicated servers.

Performance modeling of e-commerce environments can be done using analytic, simulation, and/or hybrid methods. In the case of analytic methods, queuing networks (QNS) can be used to analyze the scalability and predict the performance of e-business sites as indicated in "Scaling for E-business: technologies, models, performance and capacity planning" [MA00]. Queues in the QN are associated with the various components of a server (e.g., processors, disks), LAN segments connecting the servers, routers, firewalls, and links connecting the site to the ISP.

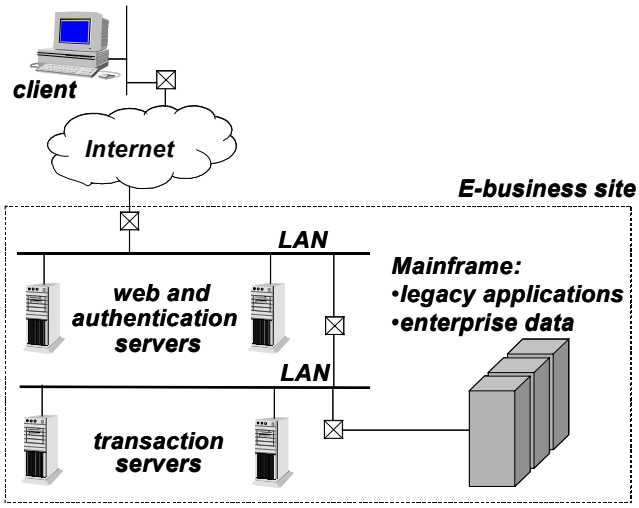


Figure 6.1 – Multi-tiered Architecture of an E-Business Site.

An important aspect to consider when modeling e-business sites is software contention. It is very common for the software architecture of the various software servers to be multi-threaded. Maximum threading levels are set with the purpose of ensuring a certain quality of service to the threads in execution. One common performance problem which might occur is elongated response times with associated low utilization of application server hardware resources, essentially due to idle server threads, and queued requests waiting on DB server response. Because an application server thread is not available to handle other requests while the DB server is processing a request, we may see large queues for application server threads. Caching of DB query results at the application server may alleviate the problem.

7. Performance Impacts of Authentication Protocols

It has been observed through measurements [KIM99] and analytic modeling [MA00] that the throughput of e-business sites can be significantly reduced when authentication operations are performed. One of the most popular authentication protocols is Secure Sockets Layer (SSL), which is being superceded by the Transport Layer Security (TLS) protocol. TLS is essentially similar to SSL version 3 and is now an Internet Engineering Task Force (IETF) standard.

TLS has two phases: a handshake phase and a data transmission phase. During the handshake phase, client and servers authenticate to one another. Client authentication to the server is optional. Public-key encryption is used during this phase to exchange secrets to be used to generate a secret key to be used in the data transfer phase, which uses symmetric key encryption. The performance impacts of the handshake phase are due to several message exchanges that add several Roundtrip Times (RTTs) and additional processing at client and server (especially for Public-Key encryption). During the data transfer phase, additional delays are due to the processing load involved in encryption/decryption and compression and decompression.

Figure 7.1 shows the throughput of secure vs. insecure connections versus the number of concurrent requests in the system. As it can be seen, for the case of insecure connections, the maximum throughput is 100 requests/sec. When TLS is used, the throughput drops. The larger the key size, and therefore the higher the security level, the lower the throughput. For example, for a key size of 1024 bits, the maximum throughput is about 20% of the maximum throughput observed in the case of insecure connections.

8. Concluding Remarks

There are several alternatives to scaling e-business sites. Some may be far more expensive than others. These days, shareholders of dot com companies are looking forward to seeing profits and not just potential profit. This means that the cost per transaction has to be lowered as much as possible. In other words, e-commerce sites have to provide the best possible quality of service with the minimum possible cost. The days of throwing plumbing at the problem are gone. This paper offers a perspective on some of the problems to be considered when dealing with scalability issues of e-business sites. Additional

information on this topic may be found at “Scaling for E-business: technologies, models, performance, and capacity planning” [MA00].

When analyzing the scalability of e-business sites, one has to consider the business, functional, customer behavior, and IT resource aspects of the problem. This paper presents a reference model that covers these four aspects.

References

[DGLS99] Devlin, B., J. Gray, B. Laing, and G. Spix, Scalability Terminology: Farms, Clones, Partitions, and Pack: RACS and RAPS,”Technical Report MS-TR-99-85, Microsoft Research, Dec. 99.

[DRC00] Global CDMA Business Opportunities, August 2000, Datacomm Research Company, www.datacommresearch.com/reports.html

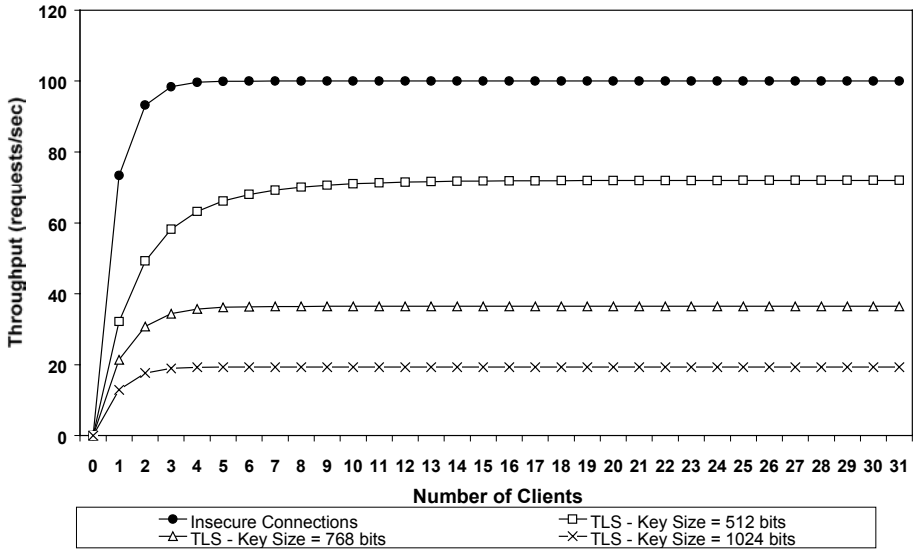


Figure 7.1 – Throughput of Secure vs. Insecure Connections Using TLS with various Key Sizes. Reprinted from [MA00] with authorization from Prentice Hall.

[KIM99] K. Kant, R. Iyer, and P. Mohapatra, “Architectural Impact of Secure Socket Layer on Internet Servers,” Technical Report, Intel Corporation, Beaverton, OR, 1999.

[KLEI75] Kleinrock, L., *Queuing Systems*, Vol I: Theory, John Wiley, 1975.

[MAES99] Maes, P., Agents that Buy and Sell, *Communications of the ACM*, Vol. 42, No. 3, March 1999, pp. 81-91.

[MA00] Menascé, D. A. and V. A. F. Almeida, *Scaling for E-business: technologies, models, performance, and capacity planning,* Prentice Hall, Upper Saddle River, NJ, 2000.

[MAFM99] Menascé, D. A, V. Almeida, R. Fonseca, and M. A. Mendes, “A Methodology for Workload Characterization of E-commerce Sites," *1999 ACM Conference on Electronic Commerce*, Denver, CO, November, 1999.

[MA98] Menascé, D. A. and V. A. F. Almeida, *Capacity Planning for Web Performance: metrics, models, and methods*, Prentice Hall, Upper Saddle River, NJ, 1998.

[MAD94] Menascé, D. A., V. A. F. Almeida, and L. W. Dowdy, *Capacity Planning and Performance Modeling: from mainframes to client-server systems*, Prentice Hall, Upper Saddle River, NJ, 1994.