https://doi.org/10.48009/2_iis_2005_168-174

ANALYZING SCALABILITY: A RISK FACTOR FOR EBUSINESS DISCONTINUITY

Dr. Cretson L. Dalmadge, Winston-Salem State University, dalmadgec@wssu.edu
Dr. Roman M. Wong, Barry University, rwong@mail.barry.edu

ABSTRACT

Dalmadge and Wong [1] identified an array of risk factors that contribute to the increase in system vulnerability, which may in turn trigger discontinuity in eBusinesses. Among others, scalability has been identified in the information systems literature as one of the most prominent risk factors for eBusinesses discontinuity. In this paper we use a popular network simulation tool to test the impacts of scalability on a network based student registration system. The objectives of the simulation are to: (i) evaluate the extent to which scalability affects system performance, and (ii) identify individual attributed of scalability that are most significant in determining the likelihood that an information system or eBusiness will perform at a desired level under increasing load. Our results supported the fact that scalability serves as a risk factor for eBusiness discontinuity but our number of simulation runs failed to deliver conclusive results on the individual attributes of scalability that serve as discriminator for eBusiness discontinuity.

Key Words: Business continuity, discontinuity, risk management, enhancers, suppressors

INTRODUCTION AND BACKGROUND

Prior research found that risk factors serve as more effective indicators of discontinuity than triggers do. We adapted the model on eBusiness discontinuity from Dalmadge and Wong's [1] work and present the modified model in Figure 1. Risk factors serve as the primary indicators of the likelihood of discontinuity in the system. They indicate the level of vulnerability inherent to the system. A system with state-of-the-art security tools, for example, is deemed to be less vulnerable than one with very little or no security.

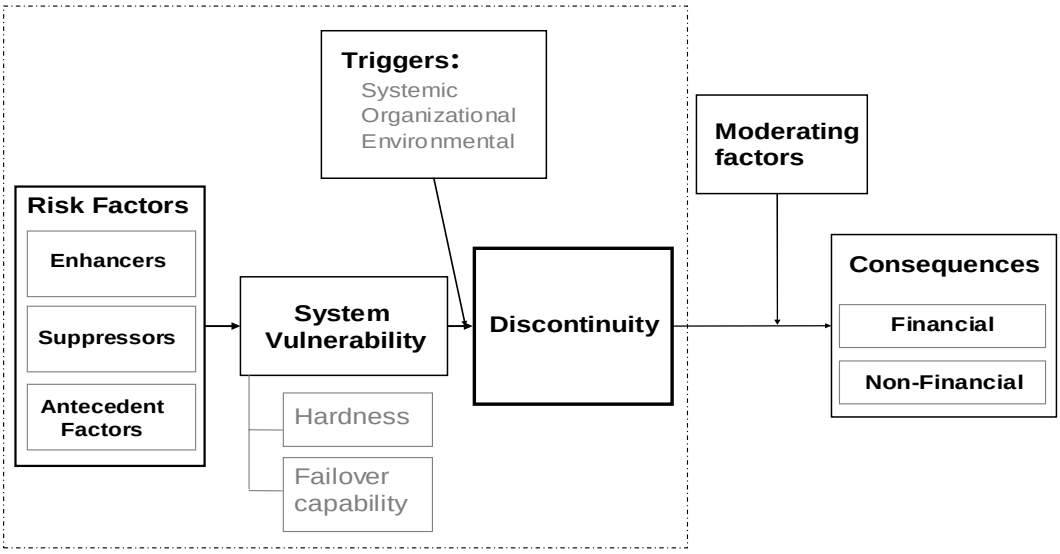

Figure 1: The eBusiness Discontinuity Model

Triggers cause discontinuity (see Figure 2). When vulnerable systems are exposed to certain triggers they suffer discontinuity. An eBusiness with weak security tools will suffer discontinuities in the face of persistent hack attacks. Another business with stronger tools may face the same level of hack attacks with little or no downtime.

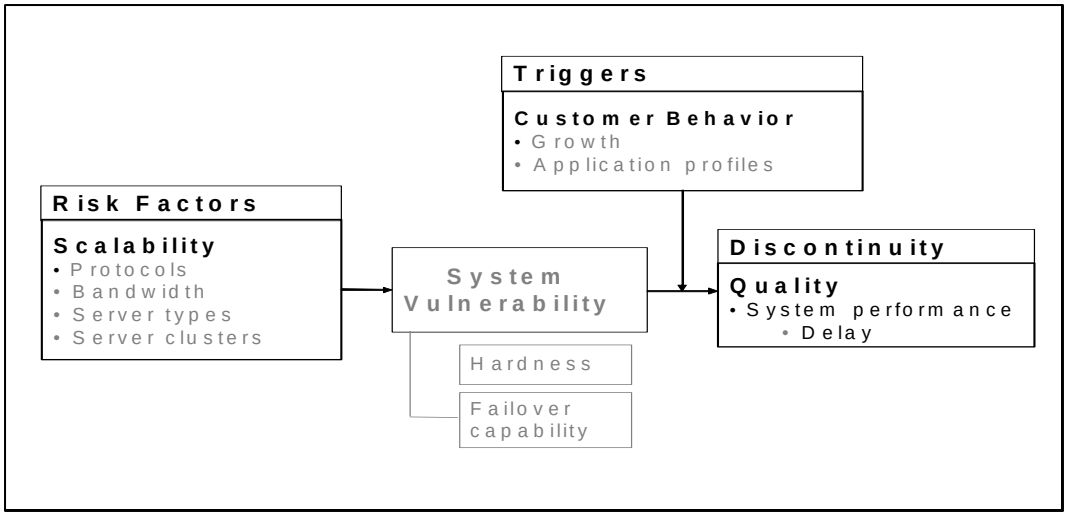

Figure 2: Scalability as a risk factor for discontinuity

eBusiness discontinuity. An eBusiness discontinuity occurs when its customers cannot interact with the business in a satisfactory manner. Conversely, eBusiness continuity is maintained as long as the business and its customers can interact satisfactorily. An eBusiness discontinuity may occur in three ways. First, an eBusiness discontinuity has occurred if the eBusiness becomes unavailable. When an eBusiness server goes down without notice, there is naturally a discontinuity. This discontinuity exists as long as the system is unavailable to process customer requests [2, 3].

Second, an eBusiness discontinuity has occurred if the eBusiness becomes inaccessible. When the volume of traffic is so high that it is unmanageable, an eBusiness may become inaccessible and thus suffer a discontinuity [4, 5]. Third, an eBusiness discontinuity has occurred if the eBusiness fails to deliver adequate quality of service. Even if an eBusiness' systems are available and accessible to its customers, there is a discontinuity if the response time is very slow i.e., below the satisfaction threshold of its customers [6]. We assess the attribute failure to deliver adequate level of quality of service as our indicator of discontinuity in this study.

The research question. The research addresses two primary issues. First, does scalability serve an effective predictor of eBusiness discontinuity? Second, how do the factors/elements of scalability individually influence the greater vulnerability of en eBusiness?

The Current Study

A review of anecdotal evidence in the popular practitioner literature points to a prominent shortlist of risk factors for eBusiness discontinuity. At the very top of this list are scalability and security. While security problems are typically associated with eBusinesses unavailability and inaccessibility (e.g., the effects of denial of service attacks), scalability is usually associated with

quality related issues such as performance. In this paper, we chose to study scalability and its effects on system performance and hence quality of service seen by customers.

Scalability. Scalability is a measure of the information systems ability to handle increased workload [7] (see Table 1). Scalable systems are able to adapt to increasing workload caused by changes in application profiles or increased in customer base. Systems with low measure of scalability fail to adapt to increasing load. The result is decreasing performance, manifested as increasing latency and delays on the system. In extreme cases these delays translate to discontinuities for eBusinesses.

Table 1: Implementation of scalability as a risk factor

Risk Factors			
Scalability			
Protocols	**Bandwidth**	**Server Type (Operating System)**	**Clustering**
Single	10Mbps	Windows	No
Multiple	100Mbps	Solaris	Yes

A combination of anecdotal evidence and theorization lead to four components of scalability for a networked information system. These are (i) the number and choice of protocols, (ii) network bandwidth, (iii) the decision to implement dependent servers (in clusters) so that they can dynamically share load, and (iv) the type of servers used in the networked system. We addressed the variables as follows. First, there is a direct relationship between the applications needed and the protocols that run on a given networked information system. Web-related traffic demands hyper-text transfer protocol (http), user datagram protocol (UDP). Also networked information systems utilize TCP/IP. A greater number of protocol demands more protocol conversion as data is moved across the system; hence, greater load is generated on the system. Second, network bandwidth serves as the channel for moving data across the system. Greater bandwidth will more effectively facilitate movement of data and allow for greater ease in growing the system. Third, server clusters are more effective at picking up extra load than single server operations. Both configurations are tested in the simulations. Fourth, we test the notion that one type of server scales more effectively than another. The simulation tool allows us to manipulate the choice of operating system. Changing the server type (e.g., Windows to Solaris) affects the nature of the server hard drive partitioning as well as the server software used in the simulation.

The combination of these four attributes lead to sixteen unique profiles (of risk factors) to be tested against the changing customer related load. It should be noted that while we chose to test these scenario, there are other possibilities that could be addresses. For example network bandwidth now includes a Gigabit Ethernet option and server types also include Linux, OpenVMS and AIX. We modeled to most popular options to add practical value to the study.

Customer behaviors. Customer behaviors influence traffic to eBusinesses (IBM high Volume Design Team 2001). As customers become savvier they utilized greater number of functionalities provided by the eBusiness. Further, application profiles typically change over time. Today's emails utilize much richer information contents than those of a few years ago. Also users are becoming increasingly comfortable with the use of richer communication tools such as video.

This coupled with the growth in number of customers results the significant increased in traffic to most websites. We model customer behaviors in our study by manipulating two variables. First, we manipulate the application profile. The simulation software allows us to add several types of application. These includes: Oracle database function, email, printing, web surfing, and video conferencing. In addition to simply adding applications we were also able to manipulate the application profile. Users may be programmed to use light, medium or heavy email; video conferencing may be set to low resolution, high resolution or VCR quality video. These along with several others allowed us to manipulate the application profiles and as such simulate greater volumes of traffic for a given number of customers. Utilizing this ability to change the application profile we were able to establish light, medium and heavy load scenario for each computer lab.

As shown in Table 2, customer related load was also manipulated by adding more customers to the system. We used an initial setup that had ten labs accounting for a maximum of 1158 simultaneous connections using very rich applications. We accounted for varying number of users by disconnecting some of the labs to produce simulation runs with four, seven or ten labs. The combination of three different lab setups and three application profiles produces nine distinct load settings for the simulations.

Trigger	
Customer Behavior	
No of Customers (implemented by changing the # of labs)	**Applications Load** (implemented by changing the app profile)
4 Labs	low
7 Labs	medium
10 Labs	high

Table 2. Implementation of Customer Profile

METHODOLOGY – THE SIMULATION

Software simulation was seen as an effective way of testing the validity of the factors that are deemed effective indicators of scalability. This choice was made for two reasons. First, simulation software allows us to manipulate and study the effects of the risk factors under varying loads. Second, any attempts to perform initial detailed studies on a 'live network' would lead to the kinds of network disruptions that few if any network administrators would tolerate. Initial calibration of the model requires that risk factors are carefully manipulated and loads are varied in a controlled manner.

Simulation does however have its risks. According to Bhargavan and colleagues [8] simulations are often plagued by two problems: (i) simulation software may possess problems with the basic logic – resulting in results that are not necessarily meaningful, and (ii) simulation code are often 'buggy'. We addressed these issues by choosing a well established tool and that is widely used in the U.S. networking market. According to Dubie [9] OPNET is the leading networking configuration/simulation program in the North American market. It leads in both the categories of (i) number of individuals and firms using the software and (ii) level of satisfaction among users of simulation tools. Given these facts, we have made the assumption that the simulation logic is at acceptable level for testing a theoretical model and the code should have little or no

Figure 4: Sample output form Opnet

errors. Figure 3 shows the network layout being tested. We modeled a typical university network with computer lab use by students to perform tasks that range from self registration, to research, class assignments, email and general web surfing. Each of the ten labs is modeled as a subnet with its own local switching. The ten labs are then connected to a core switch to provide connectivity across the broader university campus.

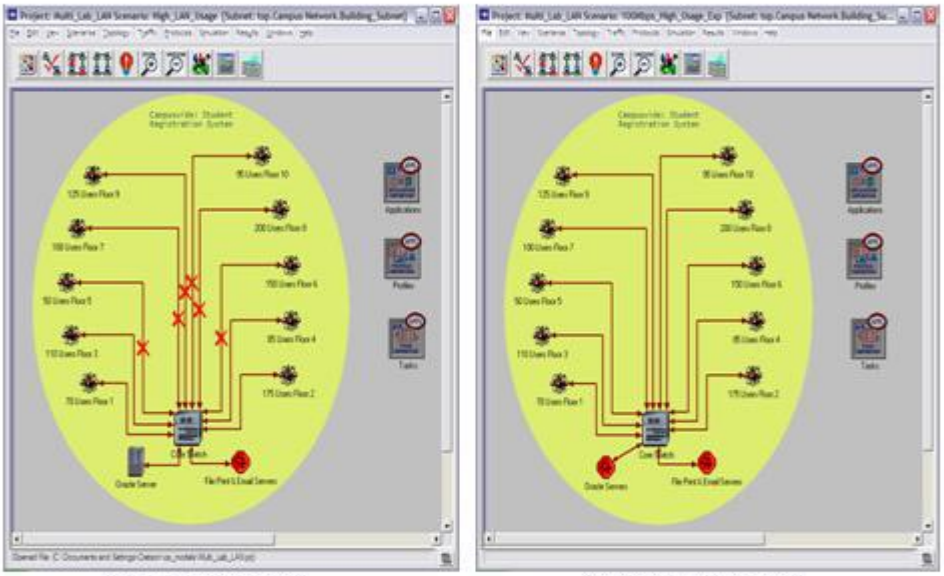

Figure 3. The network layout being tested

RESULTS AND IMPLICATIONS

The 16 risk factor profiles established above were used alternately as based properties for the system. Each of the combination of triggers was then applied to the system and quality/performance related results were recorded. Figure 4 shows a sample output of the simulation. Application delay is mapped against time as the simulation runs. This is reported for all labs in the current simulation. The average delay is obtained by exporting the results to MS Excel and calculating the average delay for that run.

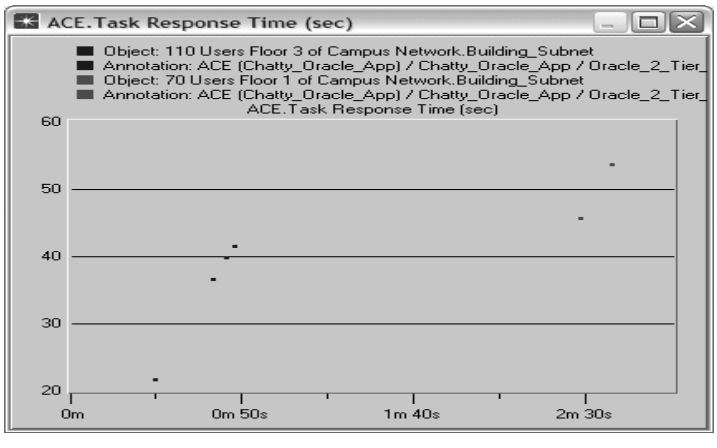

Figure 4. Sample Output From Opnet

The combination of 16 risk factor profiles and nine discrete states for the trigger resulted in 144 runs of the simulation. As the runs progressed we found that the system was overloaded and failed to produce much useful information for greater than seven labs (738 users). This largely resulted from the load generated by the Oracle operation. Failure to achieve conclusive results for all 144 runs prevented us from performing full statistics analysis of the data. As such we were unable to assess the individual contribution of the four indicators of scalability.

We proceeded by graphically analyzing the results from the runs that generated useful data. This allows us to assess whether certain unique configurations were more effective at adapting to increasing customer load than other system configurations. Figure 5 shows a sample of the graphical representation of delays produced by the simulation. The outputs for four risk profiles were graphed. These were: (i) multiple protocols, high bandwidth, Windows servers and clusters (MBWY); (ii) multiple protocols, high bandwidth, Solaris servers and clusters (MBSY); (iii) single protocol, high bandwidth, Windows servers and clusters (SBWY); and (iv) single protocol, high bandwidth, Solaris servers and clusters (SBSY). The acceptable measure of delay for the system is also shown on the graph. As shown in the figure, certain configurations resulted in systems that were better able to adapt to increasing traffic load than others. The configuration with single protocol (i.e., database data only – no email and voice traffic), broadband (i.e., Fast Ethernet connections) and servers clustered scaled better than configurations. Those configurations showed some decrease in performance – average delays increasing form the order of 20 seconds to 62 seconds as load was increased three folds. Other configurations especially those with multiple protocols and regular Ethernet connections had far greater problems adapting to increased load. Configurations with multiple protocol but having Fast Ethernet connections, and server clusters has modest performance results. Application delays increased from 25 seconds to approximately 100 seconds as the load was increased 3 folds.

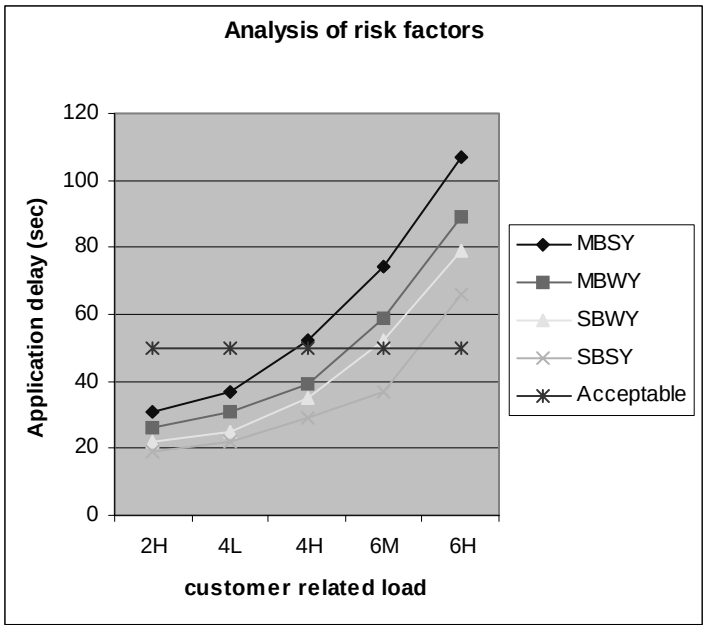

Figure 5. Application Delay for Different Load Profiles

In addition to the difference in the rate at which performance deteriorated for the different configurations, the graphic also shows that two configurations (i) single protocol, high bandwidth, Windows servers, clusters and (ii) single protocol, high bandwidth, Solaris servers, clusters) were able to pickup four of the five load settings without performance deteriorating below acceptable levels. These two configurations were very scalable. The other two had much lower measures of scalability.

FUTURE DIRECTIONS AND CONCLUSIONS

The software produced results consistent with the theoretical position that eBusinesses and networked information systems in general possess certain key system properties that influence their ability to effectively adapt to increasing network traffic. Manipulation of these properties showed that some configuration resulted in unsatisfactory system performance as load increased while other configurations provided satisfactory measure of performance. Future research needs to statistically validate the individual items that define the construct of scalability. This needs to be addressed at two levels. First, greater number of simulation runs, starting with smaller load and allowing for greater increasing in load will help to address the failure to attain statistically significant results. Second, the results need to be tested in a 'live network' scenario. This will demand the identification of networks that face dramatic difference in load over relatively short periods of time – to facilitate data collection. Student registration system was modeled in this experiment because university systems face significant differences in system load over the course of the first couple weeks of a semester as students move from a registration mode and adapt to normal campus life.

REFERENCES

1. Dalmadge, C.L. & Wong, R.M. (2004). Towards a Proactive Model for Managing eBusiness Continuity. *Issues in Information Systems*, *5*(2).
2. Boritz, J.E. & Hunton, J. E. (2002). Investigating the Impact of Auditor-Provided Systems Reliability Assurance on Potential Service Recipients. *Journal of Information Systems*, *16*(1), 69-87.
3. Jackson, C.B. (2002). The Changing Face of Continuity Planning. *Information Systems Security*, *10*(6), 18-21.
4. Chen, E. (1996). IBM not winning medals at Olympics, *Electronic News*. p. 2.
5. Fitzgerald, M.(1996). *Who's on First?*, *Editor & Publisher*, 13-15.
6. Krause, M. & Brown, L. (1996). Information security in the healthcare industry. I*nformation Systems Security*, *5*(3), 32-40.
7. Weyuker, E.J. & Avritzer, A. (2002). *A Metric for Predicting the Performance of an Application Under a Growing Workload. IBM Systems Journal*, *41*(1), 45.
8. Bhargavan, K. & Gunter, C. (2002). Verisim: Formal Analysis of Network Simulation. *IEEE Transactions on Software Engineering*, *28*(2).
9. Dubie, D. (2004). Opnet Tackles Configuration Management. *Network World*,17.