Communications of the IIMA

Volume 7 | Issue 2 Article 7

2007

Performance Analysis for Shared Services

Hai Wang
Sobey School of Business, Saint Mary's University

Performance Analysis for Shared Services

Hai Wang
Sobey School of Business,
Saint Mary's University, Canada
hwang@smu.ca

ABSTRACT

Shared services have widely spread in the government and private sectors as an alternative to outsourcing. Unlike outsourcing, shared service is the standardization and consolidation of common functions across the multiple organizations to reduce operational cost and to increase information and knowledge sharing. One critical aspect to successful shared services is to maintain specified service levels and response times for each individual organization. This paper presents a technique for performance analysis for shared services, and demonstrates the usefulness of this technique in implementing shared services.

INTRODUCTION

Recently, shared services have widely spread in government and private sectors as an alternative to outsourcing (Janssen and Joha, 2006; Leavell, 2006; Rolia *et al*., 2006; Williams, 2006). Shared service is the standardization and consolidation of common functions across multiple organizations to reduce operational cost and to increase information and knowledge sharing. The cross-organizational dimension of shared services makes up distinctive characteristics in contrast to other business processes (Davenport *et al.*, 2004; Ulbrich, 2005).

One of the important aspects of successful shared services is to maintain specified service levels and response times for each individual organization. In this paper, we present a technique for performance analysis for shared services. This technique is based on multi-class product form queuing network models and the mean value analysis algorithm (Basket *et al.*, 1975; Reiser and Lavenberg, 1980). We demonstrate the usefulness of this technique in implementing shared services.

The remainder of this paper is organized as follows. Section 2 describes the unique characteristics and implementation issues of shared services. Section 3 presents the technique for performance analysis for shared services, and Section 4 shows the usefulness and effectiveness of this technique. Finally, Section 5 provides a summary and conclusions.

SHARED SERCIVES

Characteristics of Shared Services

Services for common functions in individual organizations can be shared to reduce operational cost and to increase information and knowledge sharing through standardization and consolidation of these service processes. Generally, information systems services, accounting and financial services, customer relationship management, and human resource management are the designated lines of business processes for shared services.

Shared services require significant transformation of business processes. Cost saving through business process reengineering is the strategic opportunity of shared services (Davenport *et al.*, 2004; Davenport and Short, 1990). By sharing common non-core business functions, such as information systems services, accounting and financial services and human resource management, would dramatically improve the business performance of each individual partner organizations of shared services.

Goals of shared services are:

- Lower costs of business processes and improve services.
- Form a long-term strategic alliance with other organizations to share information and knowledge.
- Establish leadership through a focus on core business functions.

Implementation of Shared Services

As more organizations have engaged in shared services, there has been a growing realization of the importance of implementing shared services and redesigning organizations. Many organizations have discovered that it requires considerable efforts to make shared services suitable for their specific situations (Baron *et al*., 2005; Park and Kim, 2005; Rison, 2005).

Research has indicated that successful implementation of shared services requires new organizational structures (Gulati and Singh, 1998; Kakabadse and Kakabadse, 2000). Shared services drive all partner organizations to form a network with the nucleus of shared service providers. A shared service provider provides shared services and knowledge to each partner organization. Each individual organization is a stockholder of the shared service provider and acts as part of its governance. Such a strategic alliance actualizes value-adding, resource and knowledge sharing, as well as risk sharing partnerships. Successful implementations of shared serviced has been reported in the many fields, including information systems services, accounting and financial services, and human resource management (Marcolin and Ross, 2005; Park and Kim, 2005; Ulbrich, 1995).

Information technology creates fertile soil for shared services (King and Malhotra, 2000; Marcolin and Ross, 2005; Park and Kim, 2005). The rational behind shared services is that organizations tend to do fewer things better and with lower costs in order to be strong competitors. In practice, business process reengineering (BPR) is often used to re-examine shared services related to business processes and the relationship between business units (Ulbrich, 2006). Organizations perform only those core functions for which they have expert skills, and share with partner organizations those non-core services that can be performed more economically by the shared service providers.

Although shared services are often mistakenly implemented as outsourcing, there is a fundamental difference between shared services and outsourcing. For shared services, the shared service provider is formed and governed by the partner organizations (Gulati and Singh, 1998). For outsourcing, the relationship between the service provider and the outsourced organization is usually defined by a bilateral contract.

One of the important aspects of successful implementation of shared services is to maintain specified service levels and response times for each individual partner organization. As a shared service center provides services to partner organizations, different organizations have different requirements about service levels and response times. It is hard, but necessary, to predict the appropriateness of these requirements at the operational level. Currently, there is no analytical tool used for judging the appropriateness of service levels and accurately predicting the performance for shared services. In the next section, we present a technique for performance analysis of shared services.

PERFORMANCE ANALYSIS FOR SHARED SERVICES

Multi-class Product Form Queuing Network Models

Multi-class product form queuing network models have been widely used in performance analysis of resource sharing systems, such as computer systems and communication networks (Bolch *et al.*, 2006). They are capable of providing accurate predictions of service levels of different types of requests in a resource sharing system. As shared services may be treated as resource sharing systems, multi-class product form queuing network models can form a useful basis for judging the appropriateness of service levels for shared services.

A queuing network model is a collection of service centers representing the system resources that provide service to a collection of customers that represent the tasks. The tasks' competition for the service of a resource corresponds to customers' queuing into the service center. Once a customer is served by a service center and departs from it, the customer may join the queue of another service center in the network. For multi-class product form queuing

network models (Baskett *et al.*, 1975), customers in the network are classified into different classes. Different classes of customers correspond to different types of tasks with different characteristics. Moreover, for multi-class product form queuing network models, the capacity of all queues is infinity and all service centers belong to one of the following four types:

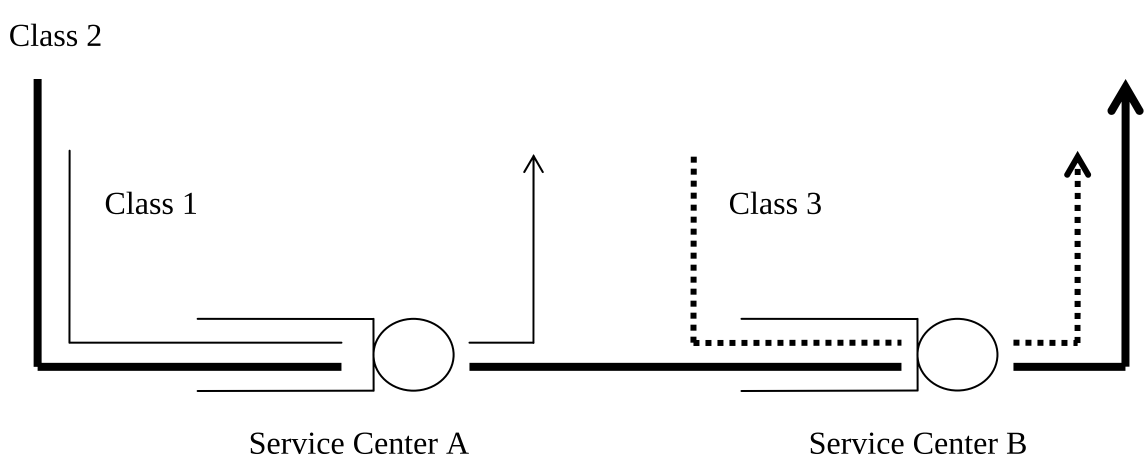
1. Single-server service centers with First Come First Served scheduling and exponential customer service time distribution, identical to each customer class.
2. Single-server service centers with Processor Sharing scheduling and arbitrary customer service time distribution.
3. Single-server service centers with Last Come First Served with Preemption scheduling and arbitrary customer service time distribution.
4. Infinity-server service centers with arbitrary customer service time distribution.

For the first three types of service centers, there is only one server to serve customers, and customers have to wait in the queue before they are served. For Type 4 service centers, there are an infinite number of servers to serve customers, and customers will be served immediately when they arrive at the service center. There is no queuing for this type of service centers. For example, a huge computer lab with many computers to serve very few students can be modeled as a Type 4 service center.

Multi-class product form queuing network models is capable of providing fairly accurate representations for many real systems. Moreover, many non-product form queuing network models can be accurately approximated by larger and more complex multi-class product form queuing networks (Bolch *et al.*, 2006). In our view, multi-class product form queuing network models is sufficient to accurately model shared services. Tasks from different partner organizations are modeled as different classes of customers in the queuing network. Resources of the shared service provider are modeled as service centers in the queuing network. The fact that each task must be accomplished eventually corresponds to the infinite queuing capacity for all queues in the network.

For example, a shared service provider with two employees, A and B, provides bookkeeping and auditing services to two different organizations, X and Y. Organization X has two different types of tasks. One requires only employee A to perform bookkeeping. The other requires employee A to perform bookkeeping first, and then employee B to perform auditing. Organization Y has only one type of tasks which requires employee B to perform auditing. We can model this shared service provider using a multi-class product form queuing networks with 3 classes of customers and 2 single-server service centers. Figure 1 shows such a model.

Figure 1: A Queuing Network Model.



The Mean Value Analysis Algorithm

The average performance measures of multi-class product form queuing network models can be evaluated by the mean value analysis algorithm (Reiser and Lavenberg, 1980).

Consider a multi-class product form queueing network (Baskett *et al.*, 1975) with C customer classes and K single-server service centers. These K service centers can be any of Type 1, 2, 3 single-server service centers described in Section 3.1. The customer classes are indexed as classes 1 through *C,* and the service centers are indexed as centers 1 through K. The customer population of the queueing network is denoted by the vector $\mathbf{N} = (N_1, N_2, ..., N_C) = [N_c]$, where N_c denotes the number of customers belonging to class c for c=1,2,...,C. The average service demand of a class c customer at center k is denoted by $D_{c,k}$ for c=1,2,...,C, and k=1,2,...,K. This average service demand is defined as the product of two factors: the average service time of a class c customer during each visit to center k, and the average number of visits by a class c customer to center k. The sum of the average service demands of all infinite-server (Type 4) service centers for class c is denoted by Z_c.

For queuing networks with population vector **N**, we are interested in the following average performance measures:

$R_{c,k}(\mathbf{N})$ = the average residence time of a class c customer at center k.
$R_c(\mathbf{N})$ = the average response time of a class c customer in the network.
$Q_{c,k}(\mathbf{N})$ = the average queue length of class c at center k.
$Q_k(\mathbf{N})$ = the average total queue length at center k.

The average residence time of a class c customer at center k, $R_{c,k}(\mathbf{N})$, is the sum of the average service demand and queuing waiting time of a class c customer at center k. Intuitively, $R_{c,k}(\mathbf{N})$ is the average response time of center k for class c customers, and $R_c(\mathbf{N})$ is the average response time of the entire system for class c customers.

The mean value analysis algorithm (Reiser and Lavenberg, 1980) involves repeated applications of the following four equations, in which $\mathbf{n} = (n_1, n_2, ..., n_C)$ is a population vector ranging from $\mathbf{0} = (0, 0, \ldots, 0)$ to $\mathbf{N} = (N_1, N_2, ..., N_C)$; $\mathbf{1}_c$ is a C-dimensional vector whose c^{th} element is one and whose other elements are zeros; and $\mathbf{n}\text{-}\mathbf{1}_c$ denotes the population vector **n** with one class c customer removed:

$$R_{c,k}(\mathbf{n}) = D_{c,k} \cdot [1 + Q_k(\mathbf{n}-\mathbf{1}_c)]$$

$$R_c(\mathbf{n}) = \sum_{k=1}^{K} R_{c,k}(\mathbf{n})$$

$$Q_{c,k}(\mathbf{n}) = \frac{n_c \cdot R_{c,k}(\mathbf{n})}{Z_c + R_c(\mathbf{n})}$$

$$Q_k(\mathbf{n}) = \sum_{c=1}^{C} Q_{c,k}(\mathbf{n})$$

with initial conditions $Q_k(\mathbf{0}) = 0$ for k=1,2,...,K.

The key component of the mean value analysis is the recursive dependence of the performance measures for one population on those for lower population levels. The performance measures of population **n** is derived from those of population $\mathbf{n}\text{-}\mathbf{1}_c$ for c=1,2,...,C.

An implementation of the mean value algorithm is shown as follows.

Algorithm Input:

C: the number of classes of customers
K: the number of single-server service centers in the network
$\mathbf{N} = [N_c]$: the population vector of customers in the network, where each element N_c is the number of class c customers in the network for c = 1,2,…,C

Z = [Z_c]: the service demand vector for infinite-server service centers, where each element Z_c is the sum of the average service demands of all infinite-server service centers for class c customers for c = 1,2,…,C

D = [$D_{c,k}$]: the service demand matrix for single-server service centers, where each element $D_{c,k}$ is the average service demand of class c customers at center k for c = 1,2,…,C and k=1,2,…,K

Algorithm Output:

$R_{c,k}(\mathbf{N})$: the average residence time of a class c customer at center k for the population vector **N,** where c=1,2,...,C and k=1,2,…,K

$R_c(\mathbf{N})$: the average response time of a class c customer in the network for the population vector **N**, where c=1,2,...,C

$Q_{c,k}(\mathbf{N})$: the average queue length of class c at center k for the population vector **N**, where c=1,2,...,C and k=1,2,…,K

$Q_k(\mathbf{N})$: the average total queue length at center k for the population vector **N**, where k=1,2,…,K

Algorithm Description:

FOR k = 1 TO K
  $Q_k(\mathbf{0}) = 0$
END FOR

FOR n_1 = 0 TO N_1
    FOR n_2 = 0 TO N_2
      ……
        FOR n_C = 0 TO N_C
            n = the population vector (n_1, n_2, ..., n_C) = [n_c] for c=1,2,…,C
            FOR c = 1 TO C
                FOR k = 1 TO K
                    IF **n** == **0** THEN
                        $R_{c,k}(\mathbf{n}) = 0$
                        $Q_{c,k}(\mathbf{n}) = 0$
                    ELSE
                        $R_{c,k}(\mathbf{n}) = D_{c,k} \cdot [1 + Q_k(\mathbf{n}-\mathbf{1}_c)]$
                        $Q_{c,k}(\mathbf{n}) = \dfrac{n_c \cdot R_{c,k}(\mathbf{n})}{Z_c + R_c(\mathbf{n})}$
                    END IF
                END FOR
            END FOR
            FOR k = 1 TO K
                $Q_k(\mathbf{n}) = \sum_{c=1}^{C} Q_{c,k}(\mathbf{n})$
            END FOR
            FOR c = 1 TO C
                $R_c(\mathbf{n}) = \sum_{k=1}^{K} R_{c,k}(\mathbf{n})$
            END FOR
        END FOR
      ……
    END FOR
END FOR

AN ILLUSTRATIVE EXAMPLE

We present an illustrative example of applying the proposed technique to the implementation of shared services.

Consider in the same example described in Section 3.1. A shared service provider with two employees, A and B, provides shared services to two different organizations, X and Y. Organization X has two different types of tasks, T_1 and T_2. On average, Type T_1 tasks only require employee A to process for 0.5 days, while Type T_2 tasks first require employee A to process for 0.2 day and then require employee B to process for another 0.2 day. Organization Y has only one type of tasks, T_3, which requires employee B to process for 2.0 days on average. As shown in Figure 1 in Section 3.1, the queuing network model consists of 3 classes of customers, 2 single-server service centers, and no infinite-server service centers. The workloads are as follows:

$$D_{1,1} = 0.5,\ D_{1,2} = 0,\ D_{2,1} = 0.2,\ D_{2,2} = 0.2,\ D_{3,1} = 0,\ D_{3,2} = 2.0,\ Z_1 = Z_2 = 0.0.$$

Suppose that organization X has 5 T_1 tasks and 3 T_2 tasks, and organization Y has 7 T_3 tasks. Hence, the population vector $\mathbf{N}$ = (5, 3, 7). We want to know whether or not the shared service provider is capable of meeting the following requirements:

Requirement I: Type T1 tasks will be accomplished within 5 days on average.
Requirement II: Type T2 tasks will be accomplished within 4 days on average.
Requirement III: Type T3 tasks will be accomplished within 20 days on average.

Applying the mean value analysis algorithm, we obtain

$$R_1(\mathbf{N}) = 3.077,\ R_2(\mathbf{N}) = 3.2,\ R_3(\mathbf{N}) = 17.231,$$

which means the average response times for Type T_1, T_2, T_3 tasks are 3.077, 3.2 and 17.231 days respectively. Hence, we are able to conclude that the shared service provider meets the requirements and delivers satisfactory services to both partner organizations.

Suppose that organization X still has 5 T_1 tasks and 3 T_2 tasks, but organization Y has 10 T_3 tasks. Hence, the population vector $\mathbf{N}$ = (5, 3, 10). We want to know whether or not the shared service provider still meets the same requirements mentioned above. Applying the mean value analysis algorithm, we obtain

$$R_1(\mathbf{N}) = 2.969,\ R_2(\mathbf{N}) = 3.8,\ R_3(\mathbf{N}) = 23.75,$$

which means the average response times for Type T_1, T_2, T_3 tasks are 2.969, 3.8 and 23.75 days respectively. Hence, we conclude that the shared service provider fails to meet Requirement III, and need to be re-designed. One solution could be that the shared service provider hires more employees to do the same job as employee B. Another solution could be that organization Y or the shared service provider hires more employees to process Type T_3 tasks that cannot be handled on time. The re-designed model should then be evaluated using the same technique to ensure that all requirements are met.

CONCLUSIONS

This paper discusses the potentials and characteristics of shared services. To provide a tool for performance analysis for shared services, this paper presents a technique based on multi-class product form queuing network models and the mean value analysis algorithm. This technique is capable of evaluating response times of shared services for each individual partner organization at the operational level, and it is useful to the successful implementation of shared services.

REFERENCES

Baron, D. B., and Bitran, G. R. (2005). Pricing of shared computer services. *Journal of Revenue and Pricing Management*, 4(1), 54-65.

Baskett, F., Chandy, K. M., Muntz, R. R., and Palacios, G. (1975). Open, closed and mixed networks of queues with different classes of customers. *Journal of the ACM*, 22(2), 248-260.

Bolch, G., Greiner, S., Meer, H., and Trivedi, K. S. (2006). *Queueing Networks and Markov Chains: Modeling and Performance Evaluation with Computer Science Applications*, 2nd edition, Wiley-Interscience, Hoboken, New Jersey.

Davenport, T. H., Harris, J. G., and Cantrell, S. (2004). Enterprise systems and ongoing process change. *Business Process Management Journal*, 10(1), 16-26.

Davenport, T. H., and Short, J. (1990). The new industrial engineering. *Sloan Management Review*, 31(4), 11-27.

Gulati, R., and Singh, H. (1998). The architecture of cooperation: managing coordination costs and appropriation concerns in strategic alliance. *Administrative Science Quarterly*, 43(4), 781-814.

Janssen, M., and Joha, A. (2006). Motives for establishing shared service centers in public administrations. *International Journal of Information Management*, 26(2), 102-112.

Kakabadse, A., and Kakabadse, N. (2000). Sourcing: New face to economies of scale and the emergence of new organizational forms. *Knowledge and Process Management*, 7(2), 107-118.

King, W. R., and Malhotra, Y. (2000). Developing a framework for analyzing IS sourcing. *Information & Management*, 37(6), 323-334.

Leavell, H. (2006). International working capital management. *The Business Review*, 5(1), 233-239.

Marcolin, B. L., and Ross, A. (2005). Complexities in IS sourcing: equifinality and relationship management. *Data Base for Advances in Information Systems*, 36(4), 29-46.

Park, J.-Y., and Kim, J. S. (2005). The impact of IS sourcing type on service quality and maintenance efforts. *Information & Management*, 42(2), 261-274.

Reiser, M., and Lavenberg, S. S. (1980). Mean value analysis of closed multichain queueing networks. *Journal of the ACM*, 27(2), 313-322.

Rison, R. P. (2005). How to reduce the cost of HR and continue to provide value. *Human Resource Planning*, 28(1), 14-17.

Rolia, J., Cherkasova, L., Arlitt, M., and Machiraju, V. (2006). Supporting application quality of service in shared resource pools. *Communications of the ACM*, 49(3), 55-60.

Ulbrich, D. (1995). Shared services: from vogue to value. *Human Resource Planning*, 18(3), 12-23.

Ulbrich, F. (2006). Improving shared service implementation: adopting lessons from the BPR movement. *Business Process Management Journal*, 12(2), 191-205.

Williams, C. A. (2006). Advancing federal financial management. *The Journal of Government Financial Management*, 55(4), 52-57.