December 1996

Pricing of Information Services Using Real-Time Databases: A Framework for Integrating User Preferences and Real-Time Workload (Best Paper Runner Up)

Prabhudev Konana
University of Texas, Austin

Alok Gupta
University of Connecticut

Dale Stahl
University of Texas, Austin

Andrew Whinston
University of Texas, Austin

PRICING OF INFORMATION SERVICES USING REAL-TIME DATABASES: A FRAMEWORK FOR INTEGRATING USER PREFERENCES AND REAL-TIME WORKLOAD

Prabhudev Konana
University of Texas, Austin

Alok Gupta
University of Connecticut

Dale O. Stahl
Andrew B. Whinston
University of Texas, Austin

Abstract

Many revolutionary information products are being offered or envisioned in electronic commerce setting. Since an economic paradigm and mass customization are implicit in electronic commerce, these products must be produced and delivered at appropriate prices with user desired service characteristics such as response time, correctness, and completeness. In this research, we investigate the information services pricing with response time (or delay) as the only service characteristic since response time can implicitly characterize other quality attributes such as correctness. In order to recognize customers' preferences, real-time databases, where transaction processing is time-cognizant, are central to information providers and can be thought of as "manufacturers" of customized products. We propose to capture user preferences by a priority pricing mechanism based on economic theory. This pricing is concerned with database access and is independent of content pricing. Our approach has a natural overload[1] management and admission control[2] techniques that can potentially increase collective benefits. Our model is evaluated using simulation and is shown to outperform a system without access pricing mechanism with respect to both system wide benefits and RTDB performance.

1. INTRODUCTION

An *economic paradigm* and *mass customization* are implicit in electronic commerce. Therefore, we need to rethink and redevelop how information products are produced and delivered over the Internet. In this paper, we provide an information services framework in the context of electronic commerce, and integrate issues in economics and databases to form one seamless issue. Real-time database (RTDB), where transaction processing is time cognizant, is central to our framework since it allows us to recognize user preferences. Using an economic paradigm, we price services for database access of various information classes with different values,[3] and priority classes. Our approach simplifies RTDB functions such as deadline assignment, admission control, overload management, scheduling, and data management.

[1]A system is said to be in overload state if transactions miss their time deadlines.

[2]Admission control is required in RTDB to selectively block transactions from executing to avoid overloading the system.

[3]Value may be based on demand and/or information content.

1.1 Information Services Framework

The demand for information has created new business opportunities on the Internet involving collecting, analyzing and providing value-added customized services to consumers. Some of the examples of customized services are financial information services, travel/tourism related services (Branding and Buchmann 1996), and medical information services. Hamalainen, Whinston and Vishik (1996) propose a revolutionary model for education brokerages that restructures the delivery of education and training.

With increasing demand for customized information products, most of the quality/service characteristics found in traditional manufacturing and service industries will also apply to information services. Some of these characteristics include on-time delivery (expected time delay), relative order, correctness, accuracy, reliability, completeness, multimedia delivery, security/privacy, mobile delivery, and ease of use. For example, "correctness" in information services may imply that information provided must be most recent (e.g., stock quotes[4]), and "completeness" refers to delivering all pertinent information (e.g., picture/medical images, expert's analysis in financial services).

However, there are subtle differences in information services, such as in financial information services, where the value of information decays rapidly with elapsed time. Furthermore, each user values information differently based on how critically it affects his/her profits (net benefits) and when it is received relative to others. Even when an information service has only entertainment value (e.g., video on-demand), customers view delays differently. We believe that, in the electronic commerce environment, customers will seek information products at a particular price with a set of service parameters that information providers must satisfy.

Figure 1 provides a generalized information service framework that includes pricing and payment systems.[5] In this framework, customers can submit a request either directly to an original information provider (OIP) or approach through an electronic information broker (EIB). An EIB is a directory service that is expected to reduce the search cost of users dramatically by matching user preferences with that of OIPs.

1.2 Basic Model: Value as a Function of Delay

One of the critical service characteristics in our framework is response time or on-time delivery. The motivation for our argument is best described with an example from financial markets.

1.2.1 Example: Financial Markets

Long before the Internet became available for electronic commerce, financial data providers operated on their own private networks (e.g., Reuters network) or leased lines using proprietary hardware (terminals) and software. Such services usually cost thousands of dollars per month. However, many new financial information services have emerged over the Internet who provide information using WWW, e-mail, Telnet or FTP (e.g., InterQuote, QuoteCom, etc.) at a remarkably low price. This has allowed a large number of small and individual investors into direct users category who until recently were dependent on the brokers for information. Further, with on-demand access to information, even the trading behavior of large traders has changed. Consider an example where a trader, after hearing favorable reports of a few Internet related technology stocks, likes to recommend certain investments to his/her clients. The trader seeks information such as management reports, earning statements, news/reports and industry outlook of these Internet related companies. Such information may be available from different databases distributed geographically. The sequence of information requirement is shown in Figure 2 (Barney 1995). The trader may submit requests sequentially to each independent OIP after identifying it or choose to submit a request to an

[4]Some data providers deliver delayed quotes of various times. These quotes must be as promised, that is, a five minute delayed quote must not be greater than five minutes delayed.

[5]For a detailed framework, readers are referred to Konana, Gupta and Whinston (1996).

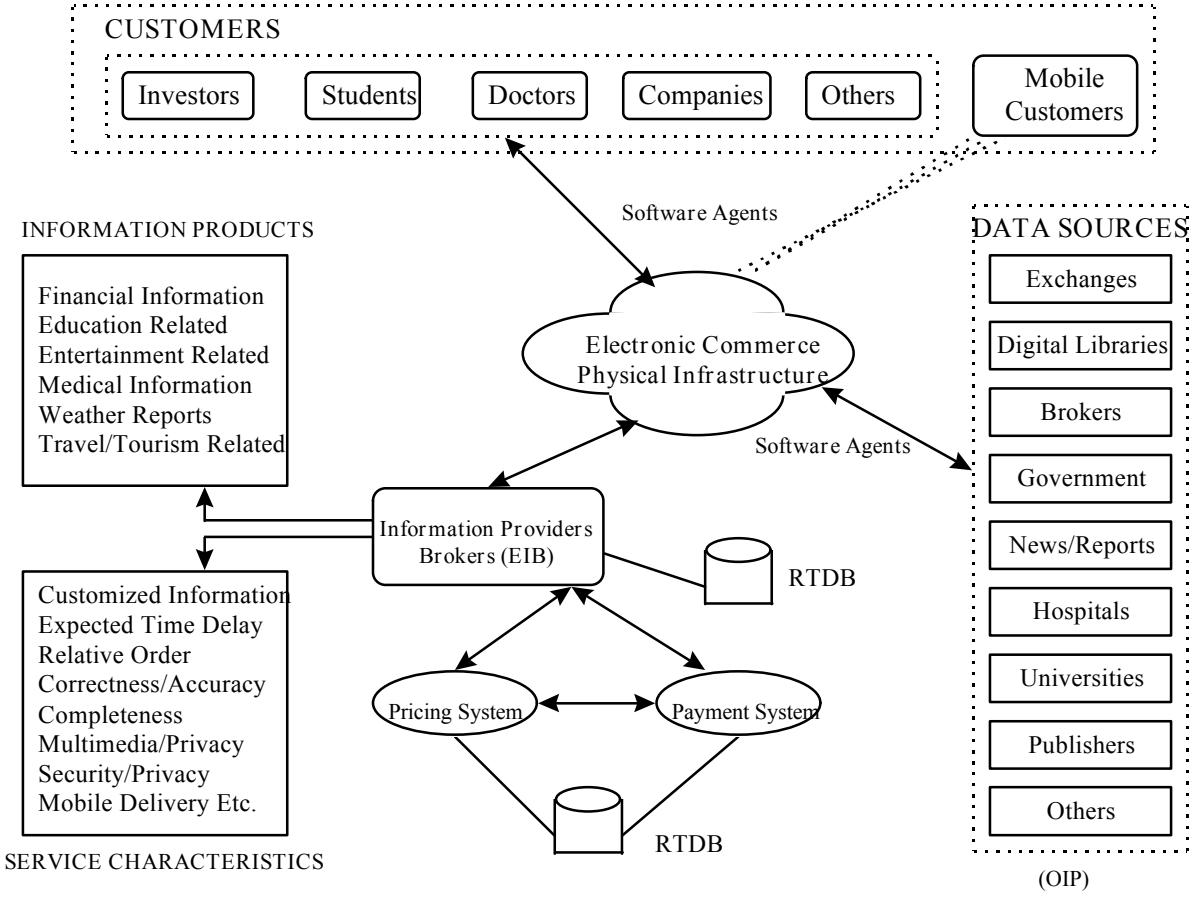

Figure 1. Information Services Framework

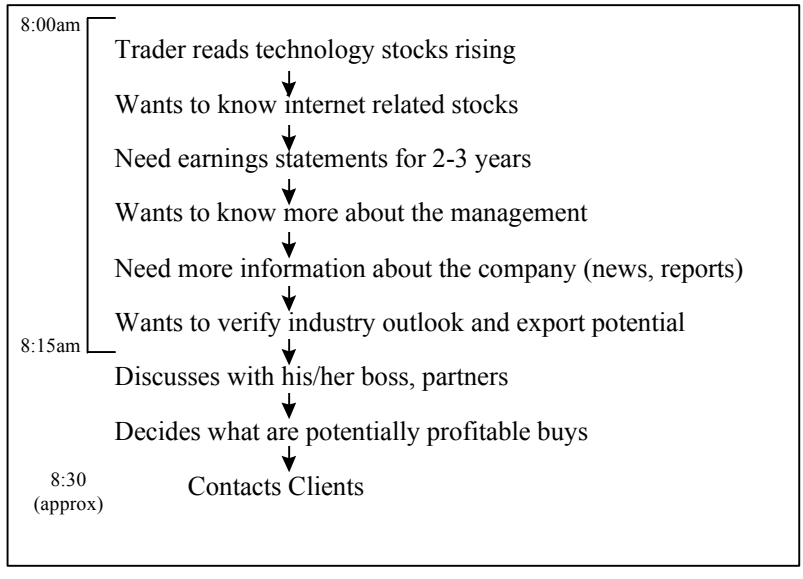

Figure 2. Information Requirement of a Trader

EIB to perform the information gathering (information bundling) efforts. The latter option may potentially save substantial time and allow the trader to invest more time in the actual analysis. Furthermore, the information retrieval becomes transparent to the trader. Since in an efficient market profit opportunities vanish quickly, a trader must analyze, suggest and execute trades in a relatively short period of time (e.g., fifteen to thirty minutes).

The issue of time has become even more critical with current communication technology where one can initiate a trade in a matter of seconds even when geographically distant. A number of studies in financial literature have found that the time windows in which major price adjustments occur are extremely small, and those who trade first based on new information benefit the most (Ederington and Lee 1993, French and Roll 1986, Patell and Wolfson 1984). For example, major price adjustments in interest rates and foreign exchange futures markets occur within one minute of scheduled macroeconomic news releases (Ederington and Lee 1993), while in equity markets it takes five to ten minutes for trading profits to disappear (French and Roll 1986). In the future, these time windows are expected to shrink further with large scale computerization, advanced high-speed communication networks, and proprietary automated trading systems. Therefore, information services must consider these market dynamics in designing their services.

1.3 Economic Modeling in Electronic Commerce

The information services framework lends itself to economic analysis where producers are the OIPs, and EIBs are middlemen arranging for supplies to satisfy consumer demands. However, several inefficiencies exists in the current systems. The existing information providers treat all customers equally[6] and, therefore, how customers value information does not affect their cost. Even if providers allow users to declare their urgency there is no incentive for users to declare their true preferences. Furthermore, some users appear to subsidize the information cost of others. This can lead to many problems: inefficient utilization of resources, excessive congestion resulting in significant delays, lack of timely information for users to make appropriate decisions, and complicated overload management techniques at providers' databases. Economic theory provides a strong basis for pricing and resource allocation that will maximize net benefits of both producers and consumers. An appropriate priority based pricing mechanism for database access acts as an admission control policy for providers.

1.4 Real-Time Databases in Electronic Commerce

Many factors contribute to the delay in providing services on the Internet: the Internet bandwidth, routers/gateways, modem capacity, transaction size, processing requirement and information server capacity. While most prior research on service delay has focused only on the Internet bandwidth as the bottleneck, we focus on information servers' capacity as the bottleneck. In order to capture user preferences in terms of delay and then process requests within that delay expectation, we need time cognizant processing at the servers. That is, information servers must have real-time database (RTDB) functionality. We assume transactions have soft deadlines, that is, transactions will continue to execute until completion even when the deadline has expired. RTDBs have vast applications in financial stock trading, network management, and manufacturing process control (Ramamritham 1993). Such databases are particularly important for modeling a real-world environment where the state of the environment changes rapidly. For example, financial markets state changes rapidly, and sensors (exchanges, tickers and other market monitoring agencies) transmit massive amounts of data every second (thousands of transactions per second) (Konana 1995). Most of the data are temporal data that arrive periodically, aperiodically and sporadically, and need to be reflected in the database in real-time. Traditional databases lack the capabilities to manage temporal data and process transactions with time deadlines. Further, unlike traditional databases where performance is evaluated based on throughout and average response time, in RTDBs performance is evaluated based on the percentage of requests that satisfy time deadline.[7]

---

[6] Even though information providers appear to separate customers by pricing, it is not based on how customers value information, but rather on the number of requests allowed within some time frame.

[7] In this research, we also have other performance metrics such as net system benefits and average tardiness.

2. RESEARCH ISSUES

In this research, we have two sets of distinct, yet related, research issues. The first set of issues is concerned with how to price information services for database access[8] that will capture the urgency (delay) and value of information to the users. These prices may be used to prioritize transactions for resource allocation in RTDBs. We provide a mechanism to associate a time deadline to users' requests by capturing their preferences. The objective in this issue is to maximize benefits to both users and service providers while at the same time maximize the number of requests that satisfy response time requirements. This research issue is unique in that it combines research issues from both economics and RTDBs. Unlike previous studies in RTDBs, where admission control is treated as a separate issue, our first research issue forms a natural admission control technique by pricing out requests based on users' preferences. The other set of issues, not considered in this study, is related to RTDB where we investigate how to allocate resources for various categories of workload and transactions such as update, triggered, read-only transactions in a dynamic environment (The research issues in RTDBs in the context of electronic commerce are discussed in Konana, Gupta and Whinston 1996).

Existing resource allocation policies suggested in the computer science literature for RTDBs do not consider the expected value of information to the user. Even though it may be argued that users' preferences and benefits are reflected in their desired response time window, the issue of how correct required response time window (user preferences) may be elicited from users' is never addressed.[9] We believe that user preferences can be elicited using an appropriate pricing mechanism. As discussed in Section 1.3, an inadequate pricing scheme may lead to serious database overload situations that often leads to performance degradation similar to the notion of "thrashing" in operating systems[10] (Kim and Srivatsava 1991). The resource allocation for time constrained transactions in databases under dynamic workload is computationally intractable (Korth, Soparker and Silberscatz 1990).

Congestion or overload situations in databases are, in general, managed by blocking overloading transactions or controlling the number of active transactions. Such overload management schemes do not consider, from an economic perspective, the well being of the users. It is a rationing mechanism without any consideration to relative importance of user needs. In this paper, we argue that an appropriate transaction pricing mechanism acts as a natural rationing mechanism.

3. RELATED WORK

Branding and Buchmann experimented using real-time database functionality for last minute trip booking in travel related services. Otherwise, not much has been reported on using RTDBs for electronic commerce. An economic paradigm, however, has been suggested for solving many traditional database problems in computer science literature. Significant among them are wide-are distributed management system — MARIPOSA (Stonebraker et al. 1994, 1996), file search problem (Moore, Richmond and Whinston 1990), optimal database design (Mendelson and Saharia 1986), and control of computing resources (Mendelson 1987). Mendelson provided a methodology for setting prices, capacity and utilization taking into account the value of users' time using a microeconomics framework. Dewan and Mendelson (1990), provide a static internal pricing mechanism considering user delay costs for a service facility. In MARIPOSA (Stonebraker et. al 1996), query processing is based on cost-delay curves where the cost is based on the load at the time of bidding and neglects future arrival of higher valued queries. Our research is different from the previous studies in that we price requests in a dynamic environment taking into account both current and expected future arrival rates for various information objects classified by importance (that is information content is priced differently for each information class).

---

[8]This is different from pricing of information content, which is beyond the scope of this research. Our research focuses on pricing database access as opposed to providing free access.

[9]Why would users reflect their true desired response time if there are no incentives to do so?

[10]Overload can occur due to both resource and data conflicts (Konana 1995).

Naor (1969) suggested controlling the steady state length of a single-server queue by introducing prices for a service. That is, when arrival rate increases, prices are increased and vice-versa, until the system reaches an equilibrium. This notion is different from studies in computer science literature where arrival rate is assumed to be fixed and the system is expected to manage the resulting workload. A number of studies have investigated resource allocation techniques in RTDBs with some success. Ramamritham (1993) and Yu et al. (1994) provide a good overview on RTDBs resource allocation techniques. However, most of these studies assume transaction deadlines are known a priori, and neglects users' value.

4. PRIORITY BASED PRICING MODEL

4.1 Real-Time Database Model

We view RTDB as a business entity manufacturing information products to customers' expected response time specification. The RTDB is assumed to be a disk-resident shared memory multiprocessor system, and consists of a large number of objects with different demands. Therefore, we classify data into many *Information Classes* of some granularity based on the importance or demand.[11] An information class with a very high demand may be cached or maintained in main memory that will improve database performance significantly. Let $C = \{1,2,....N\}$ be the set of information classes. The users' requests fall into a known set of *transaction sizes* $S = \{S_1 , S_2 ,....S_p\}$ which is measured in terms of number of pages or data-items.[12] Theoretically, there can be $|C|\times|S|$ categories of requests that we call *Transaction Classes*, $T = \{1,2,....|C|\times|S|\}$. Each transaction class j has s_j number of pages/data-items. Assume there are M priority classes, $K = \{1,2,....,M\}$, in any given transaction class j.

We assume that users' requests are pre-analyzed[13] and, therefore, data requirements (read-set -*RS*) are known a priori. Let the time required for pre-analysis be π. Since frequently accessed information can be cached or placed in main-memory, every data item in *RS* does not require disk access, and hence, we can estimate the expected number of disk accesses per transaction as $EC = \alpha \times BC + (1 - \alpha) \times WC$, where *EC*, *BC* and *WC* are expected, best case and worst case disk accesses respectively (Konana 1995; Datta et al. 1996). The parameter α is a control variable that takes a value between 0 and 1.

We assume that each read-item corresponds to a page and there is fixed amount of time, ω, to process and transmit a page. If a transaction of size *s* executes in isolation in the system then the expected response time is $Response\_time = \pi + EC \times disk\_access\_time + \omega \times s$. . The capacity of the database is a function of the CPU processing (we assume the processor being the bottleneck and not the disk access or main memory) measured in terms of pages processed and transmitted per unit time.

Let $\bar{\omega} = \{\omega_{jk} , j \in T, k \in K\}$ be the vector of waiting times for each transaction class j and priority class k. A transaction may have to wait at CPU and Disk queues at multiple instances due to disk access, time sharing etc. This waiting clearly depends on the size, *s*, of transactions. We assume that ω_{jk} represents the total waiting time at CPU disk queues. Therefore, the total expected time to process a request in transaction class j, priority class k and of size s_j is:

$$\tau(j, k) = \omega_{jk} + \pi + \omega \times s_j \qquad (1)$$

[11]Even though classifying data is not an issue in this study, it will have significant effect on database performance.

[12]Transaction size has a direct impact on the processing/response time. The higher the number of pages/data items to be fetched, the higher may be the number of I/Os.

[13]Pre-analysis can be performed off-line using a dedicated processor and involves identifying the data requirements.

4.2 Users, Preferences and Demands

Let I denote a set of users with each user subscribing to $T_i \subseteq T$. Assume service needs for user i as a stochastic process with a specific arrival rate. Let $x_i = \{x_{ijk}, j \in T, k \in K\}$ denote the vector of average flow rates for user i for transaction class j at priority class k. A user i may represent a company or a group of individuals or an individual. Therefore, the average flow rate x_i is an aggregate of usage of all individual entities within the user group for specific transaction class and priority.

Table 1. Notation

Notation	Description	Index
C	Information class	
T	Transaction classes	j
I	Set of users	i
K	Priority Classes	k
ω	Vector of waiting times for j and k	
τ	Response time	
Γ	Database processing capacity in pages unit time	
Ψ	Matrix of job arrival rates	
δ	Delay cost	
V_i	Instantaneous value of information for user i	
u_i	Net benefit for user i	
c_i^*	Cost of accessing information for user i	

We represent the instantaneous value of a user i with flow rate x_i by a continuously differentiable concave function $V_i(x_i)$. While it may appear that associating a value is a formidable task, we can use the "rule-of-thumb" approach captured in client software (agents).[14] For example, in financial applications an investor may have an amount X in a portfolio of stocks and may anticipate an expected return n%, then the instantaneous value is $(X \times n)/100$. However, the net value to a user is less than $V_i(x_i)$ because the value of the information diminishes with elapsed time and there is a cost to retrieve this information. If δ_{ij} is the delay cost per unit time for a user i for transaction class j , then the expected delay cost for information class j of priority k is $\delta_{ij}\tau(j, k)$. Again, the estimation of delay cost appears to be a difficult task. This cost may be determined based on the investment outlook, that is, whether the expected return is based on long-term (as in mutual funds) or short-term (as in securities trading) outlook. Let the monetary cost to access information from the database for transaction class j and priority k be $r_j = \{r_{ji}, k \in K\}$. Then the minimum cost of accessing information in transaction class j from the database with a given waiting time and rental cost is

$$c_{ij}^{*}(r,\omega) + \min_{k}[\delta_{ij}\tau(j,k) + r_j] \tag{2}$$

The task of finding the cost and priority can be automated using client agents[15] Given that there is a monetary and non-monetary costs for retrieving information, the overall net benefit to the user i is

[14]In our model, only users need to know the value, V, and it is not required for actual price computation.

[15]This issue is a subject of future research and not discussed in detail due to space limitation.

$$u_i(x_i,r,\omega) = V_i(x_i) - \sum_j \sum_k x_{ijk} c_{ij}^*(r,\omega) \tag{3}$$

We stress the fact that the waiting times are fixed for a period of time, while in reality it may differ depending on the current demand and expected future demands from other users. Accordingly, we assume each user *i* will choose x_i to maximize u_i taking (r, ω) as fixed.

4.3 Optimal Resource Allocation

Disregarding problems such as deadlock resulting from concurrency control we can say an equilibrium would be that the demand $D = \sum_i \sum_j \sum_k x_{ijk} s_j$ is less than the capacity of the database, Γ, in any given time interval. If $D > \Gamma$, then, clearly, the expected waiting times cannot be satisfied, and over a longer run, the waiting times $\to \infty$. The pricing mechanism provides a natural rationing mechanism to insure that the demand is less than the capacity.

Let the entire array of demands for all users, transaction classes, priority classes and sizes be denoted by $\underline{x} = \{x_{ijk}, i0I, j0T, k0K\}$. In general, the expected waiting times at the database for transaction class *j* before being serviced will depend on the distribution of job arrival rates by priority class and transaction class. This distribution is given by

$$\Psi_{jk} = \sum_i x_{ijk} \tag{4}$$

Let $\Psi_j = \{\Psi_{jk}, k0K\}$ denote the matrix of job arrival rates to the database for each transaction class by priority class. The aggregate flow to the database in priority class *k* for any transaction class is $D_{ij} = s_j \Psi_{jk}$. We approximate the expected waiting time for transaction class *j* at the database given priority class *k* as a function of the distribution matrix Ψ_j and capacity Γ.

$$\omega_{ij} = \Omega_{ik}(\Psi(\underline{x});\Gamma) \tag{5}$$

where $\Omega_{jk}(\bullet;\Gamma)$ is continuously differentiable, strictly increasing and convex as long as $\sum_j \sum_k D_{jk} < \Gamma$ and $\Omega_{jk}(0;\Gamma)$ = 0. Further $\Omega_{jk}(\Psi(\underline{x});\Gamma) \to \infty$ as $\sum_j \sum_k D_{jk} \to D$. We assume that $\partial\Omega_q/\partial\Psi_{jk} \geq \partial\Omega_q/\partial\Psi_{jk}$, for all $k < k\mathcal{N}$, that is, the incremental waiting time imposed on priority *q* transactions is greatest for transactions arriving with the highest priority.

To derive the optimal trade-off, we need to define a system-wide welfare function. It is natural to take the sum of non-pecuniary user benefits, that is, benefits of all users minus the delay cost for all users:

$$W(x,\omega) \equiv \sum_i \sum_j (V_i(x_i) - \delta_{ij} \sum_k x_{ijk} \tau(j,k)) \tag{6}$$

We now seek an allocation of demands, $\underline{x} = \{x_{ijk}, i0I, j0T, k0K\}$ and waiting times w_{jk} that maximizes $X(x, \omega)$ subject to equation 5 (Gupta, Stahl and Whinston 1996). To solve the global maximization problem using the Lagrangian method, we define the Lagrangian function:

$$L(x,\omega,\gamma) \equiv W(x,\omega) + \sum_k \gamma_k[\omega_k - \Omega_k(\Psi(\underline{x});\Gamma)] \tag{7}$$

Then the Kuhn-Tucker conditions for optimality are:

$$\partial V_{ij}/\partial x_{ijk} - \delta_{ij}\tau(j,k) \le \sum_p [\partial\Omega_p/\partial\Psi_{jk}]\gamma_p \qquad \forall i,j,k \tag{8}$$

$$\partial V_{ij}/\partial x_{ijk} - \delta_{ij}\tau(j,k) < \sum_p [\partial\Omega_p/\partial\Psi_{jk}]\gamma_p \rightarrow x_{ijk} = 0 \qquad \forall i,j,k \tag{9}$$

and the Lagrangian multiplier is:

$$\gamma_k = \sum_i \sum_j \delta_{ij} x_{ijk} \tag{10}$$

The interpretation of conditions 8, 9 and 10 becomes clear in subsequent paragraphs. Users will maximize their utility by making appropriate submission decisions. From equation 3, if a user, *i*, has a positive flow for service *j* (i.e., $x_{ijk} > 0$), then $\partial V_{ij}/\partial x_{ijk} = c_{ij}^*(r,\omega) = \delta_{ij}\tau(j,k) + r_{jk}$. Then, substituting for γ_p in equations 8 and 9 using equation 10, x_{ijk} will satisfy equations 8 and 9 if

$$r_{jk} = \sum_p [\partial\Omega_p/\partial\Psi_{jk}] \sum_i \sum_j {}' \delta_{ij} x_{ij}{}'_p \tag{11}$$

The interpretation of this rental price is simple: the welfare maximizing rental price for database access in priority *k* must equal the average cost of aggregate delays $(\sum_i \sum_j \delta_{ij} x_{ijk})$, weighted by the waiting-time/throughput tradeoff $(\partial\Omega_p/\partial\Psi_{ij})$.

Condition 8 states that marginal increase in value per unit increase in flow rate minus the delay costs (marginal benefit) must be less than or equal to the rental price. Condition 9 states that if the marginal benefit is less than rental price then the flow rate will tend toward zero.

Given our assumptions about the waiting time function, Ω, the rental prices are highest for highest priority class (priority 1) and decreasing as the priority class decreases, i.e., $r_{\bullet,k} > r_{\bullet,k+1}$. Hence one could think of $r_{\bullet,M}$ as the base price, and $(r_{\bullet,k} - r_{\bullet,M})$ as the premium for accessing the higher priority.

Note that equation 11 is not an explicit formula for rental price r_{jk} since r_{jk} enters the right hand side through x_{ijk} and the resulting arrival rate matrix Ψ. Instead of using traditional fixed-point methods of computing prices, we favor an approach

that is motivated by the classical tatonnement process (Hahn 1982). Our approach has the benefit of not requiring the knowledge of the demand functions and providing an adaptive mechanism to compute and implement prices in real-time.

In summary, every transaction class is associated with a price given some time delay. Therefore, when a user's request arrives with service characteristics and price range, the server agent will identify the best priority for that transaction. This priority is then used for resource allocation (CPU and Disk) in the RTDB. Transactions in the same priority class are prioritized by a first-come-first-served (FCFS) basis. A request may not be serviced if the price a user is prepared to pay is infeasible with the service requirements.

4.4 Overload Management and Price Recomputation

The problem of missing expected deadline (overload) implies that the producer (database) is unable to meet the service requirements (requests that miss their deadlines will continue to execute). Therefore, once the system notices that transactions miss their deadlines, it activates the overload mechanism. In our case overload implies that the prices must be adjusted to reflect the true demand for system resources which in turn changes the admission control policy.

In our price computing approach, we measure the average flows at the database queues and predict the expected demand and waiting times. Let $x_{=ijk}(t)$ denote the current time-averaged estimate of x_{ijk}, and let $T_k(t)$ denote the current time-averaged waiting time. These estimates can then be used to estimate the function Ω_k and its derivatives. By monitoring the current state of the database (prices and waiting times), the users' delay costs can be estimated by monitoring their choices of priority. Let $r_{jk}(t)$ be the estimate of rental price by using the estimates $x_{ijk}(t)$ and $w_k(t)$ after a certain time period t, $r_{jk}(t)$ be the actually implemented price in period t, and $\beta \in (0,1)$; then to lessen the chances for instabilities due to over-responsiveness, the prices for period $t + 1$ can be set to

$$r_{ji}(t + 1) = \beta r_{jk}(t) + (1 - \beta) r_{jk}(t) \tag{12}$$

5. SIMULATION MODEL

We evaluate the performance of RTDB using priority pricing through simulation. The analytical model we provided in previous sections to compute optimal prices for database access requires the knowledge of demand functions and their variability with time. However, in electronic commerce framework, the demand for services may be extremely dynamic and unpredictable. Since price recomputation using analytical modeling may be computationally very expensive, we prefer to manage overload using the mechanism discussed in section 4.3; the only appropriate mechanism to study such a pricing system is through simulation.

We implemented a computer simulation model that captures the main elements of a real-time database system in CSIM, a discrete event simulation language (Schwetman 1991). Our simulation uses a queuing model of a single-site shared-memory disk-resident multiprocessor database system. We neglect concurrency control since it is not an issue. We compare our model with that of a system without any database access cost and overload management, and without access cost and no overload management.

The simulator consists of six modules: transaction generator, price generator, transaction manager, resource manager, database and statistics collector. We do not explicitly model memory pages but rather use probabilistic method as in Abbott and Garcia-Molina (1992). If the page is not in the memory, then I/O service request is created to the appropriate disk. The database consists of N information classes with a total of *Dbsize* pages. The pages are uniformly distributed across all disks and we can map a page in an information class to a specific disk. The probability of whether a page is in the disk depends on the importance of the information class to which it belongs. The transaction generator generates transactions to the system from a Poisson process. A transaction belongs to an information class with a probability PI_i (where $i = 1, 2,...N$). The size (number of pages) of the transaction is chosen uniformly from a range *SizeInterval*. Every incoming transaction will be associated with a *Value* and *DelayCost* drawn from normal distributions of mean and standard deviations (μ_v, σ_v) and (μ_d, σ_d) respectively. The scheduling discipline is assumed to be FCFS.

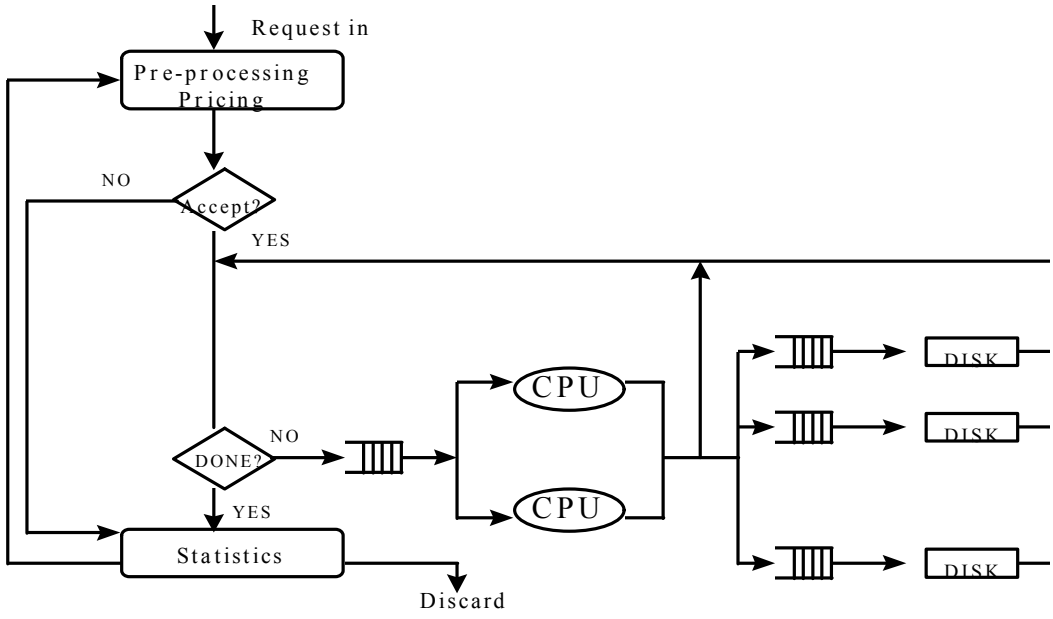

Figure 3. Physical Queuing Model

Our simulation differs significantly from those discussed in Abbott and Garcia-Molina and in Lee and Son (1993) on how deadlines are associated with each transaction. Rather than assigning deadlines to each transaction based on number of pages and slack time, we associate priorities and expected deadlines using the expected value, delay cost and rental cost. Our approach is based on the fact that every transaction class and priority is associated with an expected deadline. Tables 2 and 3 provide resource and transaction parameters. These parameters have been adopted from published simulation literature in real-time databases (Abbott and Garcia-Molina 1992; Konana 1995; Kim Srivatsava 1991) and economics (Gupta, Stahl and Whinston 1996) modified for our study. Figure 3 provides the physical queuing model.

Table 2. Resource Parameters

Parameter	Meaning	Value
NumCPU	Number of CPUs	1
NumDisks	Number of disks	2
CPUTime	CPU time for each data page	10msec
DiskTime	Disk access time for each data page	20msec
DBSize	Number of pages in the database	1000
N	Number of Information Classes	2
K	Number of Priority Classes	[1-2]

Table 3. Transaction Parameters

Parameter	Meaning	Value
ArrivalRate	Transaction arrival rate per second	[0-50]
PI_i	Probability belonging to Information Class *i*	0.50
μ_v, σ_v	Parameters for transaction value	(25,7)
μ_d, σ_d	Parameters for transaction delay cost	(4,1)
SizeInterval	Number of pages accessed per transaction	[1-20]

5.1 Performance Metrics

We use several metrics to evaluate our model. Since we try to maximize social welfare, the key performance metric is ***the net system benefits accrued per unit time*** (*NB*). This is computed as follows:

$$NB = \frac{\sum_i (Value_i - (delay\_cost_i \times t_i))}{\sum_i t_i}$$

where $Value_i$, $delay\_cost_i$, and t_i represent the instantaneous value, delay cost per unit time, and time delay of transaction *i*.

We also use other performance metrics used in the RTDB literature such as the *MissRatio* and *Average tardiness*. *Average tardiness* is defined as the average lateness of transactions that completed after the deadline. The *MissRatio* is computed as follows:

$$MissRatio = \frac{Number\ of\ transactions\ missing\ deadline}{Number\ of\ transactions\ arriving}$$

6. SIMULATION RESULTS

The base parameters for resources and transactions are provided in Tables 2 and 3. Our primary objective is to show the validity of our pricing model in a heavily loaded system rather than a lightly loaded system. Therefore, we opted to have a single processor in order to reach overload situations quickly. Further, since no real data are available for transaction values and delay costs, we have assumed certain distributions based on similar studies in network pricing (Gupta, Stahl and Whinston 1996). This experimental conditions will in no way diminish the validity of the results, but will support our pricing based transaction management.

The simulation was run on HP-UX operating system on HP workstations. In these experiments, we varied the arrival rate from ten transactions/second to fifty transactions/second in increments of ten. Statistics gathered were based on the replication-deletion approach (Law and Kelton 1991). Each experiment consisted of ten runs with a transient period of 1,000 time units and length 10,000 time units each. Statistics collected were averages of these ten runs.

6.1 Effect on Net Benefits

Figure 4 graphs the net system benefits (collective benefits of consumer and producer) with and without pricing mechanism. At low arrival rates, pricing is not an issue since the system in not overloaded and every request's response time is satisfied. However, at higher arrival rates, the collective benefits with pricing (that is, with database access cost) is significantly higher than that without pricing (that is, without database access cost) at the 95% confidence level. The experiment was also repeated for a system with no admission control and pricing. In fact, the system provided significant negative benefits, as expected, and hence not shown along with other results. The reason for higher system benefits is that, at higher arrival rates, only those requests with higher value, delay costs and rental price were admitted. This value based admission control effectively blocks requests with lesser value and those requiring significant resources.

6.2 Effect on Miss Ratio and Average Tardiness

Figure 5 shows the miss ratio with pricing and without pricing mechanism. In both cases, requests were admitted based on whether the customer gained a positive net benefit. At higher arrival rates, the miss ratio for the system without pricing was significantly higher than that with pricing. Surprisingly, the miss ratio in both cases decrease at higher arrival rates. This behavior became apparent on analyzing the actual miss ratios at various sizes of transactions. In fact, at very high arrival rates, a sufficient number of smaller size jobs were present and provided significantly more benefits than larger sized requests. The larger sized transactions consume more resources while adding little to the collective benefits. At arrival rates between ten and twenty transactions/second some large sized transactions were allowed to execute that affected execution of smaller sized transactions resulting in higher miss

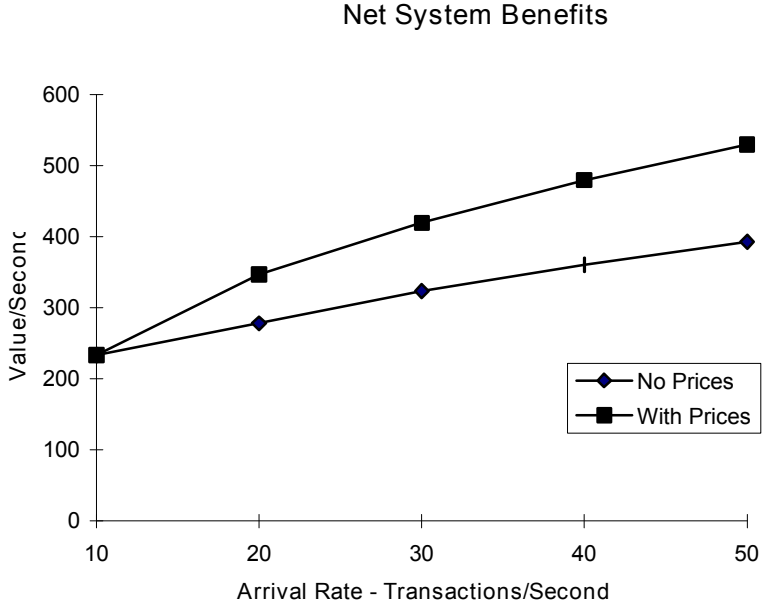

Figure 4. Net System Benefits

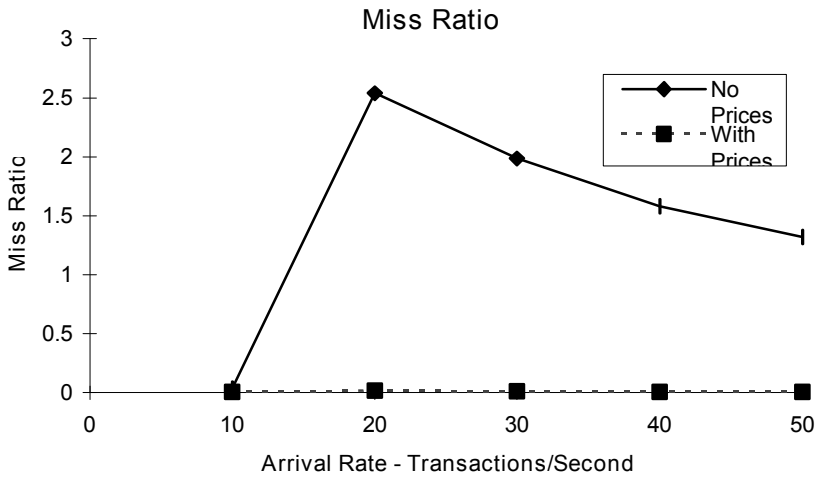

Figure 5. Miss Ratio

ratios. This result is consistent with the average tardiness of late jobs shown in Figure 6. The average tardiness actually reduced at higher arrival rates since large transactions were blocked and smaller transactions were executed. It may, however, appear that the system favors requests of smaller size (bias against larger sized requests). We can argue from an economic perspective that blocking larger requests is, in fact, beneficial for both producer and consumer. We also conducted experiments without both pricing and admission control. The miss ratio hits over 90% even with an arrival rate twenty transactions/second.

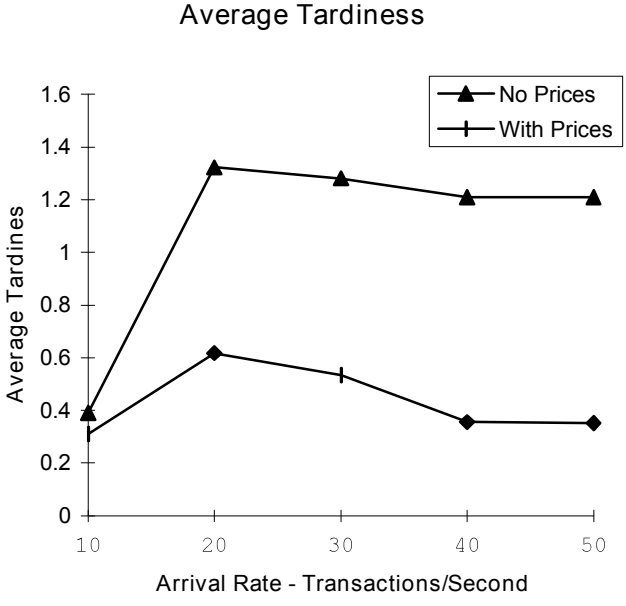

Figure 6. Average Tardiness

7. CONCLUSION AND FUTURE RESEARCH

We provided a framework for information services on the Internet in the electronic commerce setting. The issue we investigated involves how to price services for database access based on priorities, users' preferences (e.g., response time), and how users value information. In this study, we considered response time as the only service characteristic. To incorporate users' preferences, such as response time or time delay, RTDBs are required since transaction processing in these databases is time cognizant. We derive rental prices for each priority class that support efficient resource allocation, admission control and overload management in databases. This study integrates issues in economics and RTDB for maximizing collective benefits. We evaluated our pricing model against a system without prices for database through simulation studies. We show that our model outperforms a system without access prices both from an economic perspective and database performance.

In future research, we will evaluate our model under a multiprocessor environment, other scheduling algorithms such as earliest deadline first (EDF), and least slack first (LSF) and various resource and transaction parameters. Our database access pricing model will be extended to include other service characteristics such as completeness and correctness. We need to include other database workload such as update only and triggered transactions into our model since these transactions also consume resources. There are other issues that we need to explore further, such as developing a framework for client agents to embed user preferences in a specific domain (e.g., financial services).

8. REFERENCES

Abbott, R., and Garcia-Molina, H. "Scheduling Real-Time Transactions: Performance Evaluation." *ACM Transactions on Database Systems*, Volume 17, Number 3, September 1992, pp. 513-560.

Barney, L. "Training for the Big Time." *Wall Street and Technology*, Volume 13, Number 11, 1995, pp. 38-40.

Branding, H., and Buchmann, A. P. "Unbundling RTDBMS Functionality to Support WWW Applications." In *Proceedings of the First International Workshop on Real-time Databases*, March 1996, pp. 45-47.

Datta, A.; Mukkerjee, S.; Konana, P.; Viguier, I.; and Bajaj, A. "Multiclass Transactions Scheduling and Overload Management in Firm Deadline Real-Time Database Systems." *Information Systems*, Volume 21, Number 1, March 1996, pp. 29-54.

Dewan, S., and Mendelson, H. "User Delay Costs and Internal Pricing for a Service Facility." *Management Science*, Volume 36, Number 12, December 1990, pp. 1502-1517.

Ederington, L. H., and Lee, J. H. "How Markets Process Information: News Releases and Volatility." *The Journal of Finance*, Volume 38, Number 4, September 1993, pp. 1161-1191.

French, K. R., and Roll, R. "Stock Return Variances: The Arrival of Information and the Reaction of Traders." *Journal of Financial Economics*, Volume 17, 1986, pp. 5-26.

Gupta, A.; Stahl, D. O.; and Whinston, A. B. "An Economic Approach to Networked Computing with Priority Classes." *Organizational Computing and Electronic Commerce*, Volume 6, Number 1, 1996, pp. 71-95.

Hahn, T. *Stability*. Amsterdam: North-Holland, 1982.

Hamalainen, M.; Whinston, A. B.; and Vishik, S. "Electronic Markets for Learning: Developing Education Brokerages on the Internet." *Communications of the ACM*, Volume 39, Number 6, June 1996.

Kim, W., and Srivastava, J. "Enhancing Real Time DBMS Performance with Multiversion Data and Priority Based Disk Scheduling." *Proceedings of the IEEE Real-Time Systems Symposium*, 1991, pp. 212-231.

Konana, P. "A Transaction Model for Active and Real-time Databases." Unpublished Ph.D Thesis, University of Arizona, 1995.

Konana, P.; Gupta, A.; and Whinston, A. B. "Research Issues in Real-Time DBMS in the Context of Electronic Commerce." Submitted for publication. Also available as Technical Report, Center for Information Systems Management, The University of Texas at Austin, 1996.

Korth, H. F.; Soparkar, N.; and Silberschatz, A. "Triggered Real-Time Databases with Consistency Constraints." *Proceedings of the Sixteenth Conference on Very Large Data Bases*, August 1990, pp. 71-82.

Law, A. M., and Kelton, W. D. "Simulation Modeling and Analysis." New York: McGraw Hill, 1991.

Lee, J., and Son, S. H. "Using Dynamic Adjustment of Serialization Order for Real-Time Database Systems." *Proceedings of IEEE Real-Time Systems Symposium*, December 1993, pp. 66-75.

Mendelson, H. "Pricing Computer Services: Queuing Effects." *Communications of the ACM*, Volume 28, Number 3, March 1987, pp. 312-321.

Mendelson, H., and Saharia, A. N. "Incomplete Information Costs and Database Design." *ACM Transactions on Database Systems*, Volume 11, Number 2, June 1986, pp. 159-185.

Moore, J. C.; Richmond, W. B.; and Whinston, A. B. "A Decision-Theoretic Approach to Information Retrieval." *ACM Transactions on Database Systems*, Volume 15, Number 3, September 1990, pp. 311-340.

Naor, P. "On the Regulation of Queue Size by Levying Tolls." *Econometrica*, Volume 37, 1969, pp. 15-24.

Patell, J. M., and Wolfson, M. A. "The Intra-Day Speed of Adjustment of Stock Prices to Earnings and Dividend Announcements." *Journal of Financial Economics*, Volume 13, 1984, pp. 223-252.

Ramamritham, K. "Real-Time Databases." *International Journal of Distributed and Parallel Databases*, Volume 1, Number 2, 1993, pp. 199-226.

Schwetman, H. Microelectronics and Computer Technology Corporation (MCC), Austin, Texas, 1991.

Stonebraker, M.; Aoki, P. M.; Pfeffer, A.; Sah, A.; Staelin, J. C.; and Yu, A. "MARIPOSA: A Wide-Area Distributed Database System." *Journal of VLBD*, Volume 5, Number 1, 1996, pp. 64-84.

Stonebraker, M.; Devine, R.; Kornacker, M.; Litwin, W.; Sah, A.; and Staelin, C. "An Economic Paradigm for Query Processing and Data Migration in Mariposa." *Proceedings of Third International Conference on Parallel and Distributed Information Systems*, September 1994.

Yu, P. S.; Wu, K-L.; Lin, K-J.; and Son, S. H. "On Real-Time Databases: Concurrency Control and Scheduling." *Proceedings of the IEEE*, Volume 82, Number 1, 1994, pp. 140-157.