INTERNATIONAL JOURNAL OF NETWORK MANAGEMENT
Int. J. Network Mgmt 2002; **12**: 117–128 (DOI: 10.1002/nem.427)

Managing computing resources in active intranets

By Ramnath K. Chellappa* and Alok Gupta

The objective of this paper is to present an economic pricing-based resource management technique for Intranets that has the capability of managing Intranet resources from an organizational perspective. We discuss the adoption of this pricing scheme at three stages: application level, node level, and data-stream level implementations. These three levels demonstrate how a pricing based approach can be used with the current technology and also be migrated to evolving network architectures such as active networks. Copyright © 2002 John Wiley & Sons, Ltd.

Introduction

Popular literature has often described Intranets as simply intra-webs or internal corporate Web servers. However, a functional analysis of Intranets reveals that they are predominantly used to implement a company's business protocol.[9] Our approach adopts the definition that Intranets are 'a secure corporate networks with rich functional features of a Local Area Network (LAN) interconnected by the Internet and/or its technologies and applications'. The primary feature of LANs is that they do not encounter problems of congestion and security by virtue of their limited geographical scope. However, Intranets do not enjoy the luxury of providing unlimited bandwidth or foolproof security. Furthermore, LANs are typically controlled by Network Operating Systems (NOS) that provide a form of centralized administration of the network resources. This leads to the question: How can resources be managed in an Intranet, that is geographically dispersed and cuts across functional and system boundaries? In addition, the complexity of network management and resource allocation grows exponentially with the advent of active networks[8] and mobile computing technology. For example, competing requirements of different users where each user's mobile agent is trying to grab more resources for its own use could result in serious degradation of service quality for everyone.

The current role of network management applications is often a passive management of hardware rather than a proactive acquisition and management of valuable resources—a necessity for effective Intranet management. We argue that network management for the Intranets needs to be proactive. This involves: allocation of resources, secure communications, replication of business processes with natural priorities (which may be as simple as delivering the CEO's email first or as complex as determining which of several computing jobs should be allocated a higher priority), and consideration of organizational objectives. This calls for the integration of technology with business objectives to manage Intranet resources.

Economists and MIS academics have argued that economic pricing can be used for effective network resource management[1–4] that can provide

Dr Ramnath K. Chellappa is Assistant Professor of Information Systems and Co-director of ebizlab at the Marshall School of Business at the University of Southern California. His research papers have been published in leading journals and conferences.

Dr. Alok Gupta is Assistant Professor of Information Systems and Co-director of Treibick Electronic Commerce Initiative at the Department of Operations and Information Management, University of Connecticut. Dr Gupta serves on the editorial board of Decision Support Systems and Brazilian Electronic Journal of Economics.

**Correspondence to: Dr. Ramnath Chellappa, Assistant Professor of Information Systems, USC, Marshall School of Business, Bridge Hall 401, Los Angeles, CA 90089, USA.*

Published online 29 January 2002

the required capabilities of proactive network management. However, as Shenker *et al.*[5] point out, it is not clear how these economic pricing models can be implemented and integrated with networking technology and tools. Indeed, a main drawback of most pricing approaches has been the lack of understanding of the Internet's mode of operation. The Internet operates on a purely co-operative basis, i.e., intermediate nodes such routers and gateways (that may be privately owned) forward IP packets on the understanding that their packets will similarly be forwarded at other nodes. Thus the concept of pricing seems alien to the Internet. On the other hand, an Intranet that runs on top of the Internet *can be built with partnerships with adjoining routers by selectively casting or routing packets only through these particular nodes*. Therefore, a pricing implementation on Intranets can, more easily, be designed to provide active administration of computing resources.

In the next section we describe priority pricing-based network management that results in a greater overall benefit for the network. This pricing mechanism can be implemented with relatively small computational overhead since the prices at each individual network element depend upon the load experienced by that element and are not dependent on other network elements directly. However, the computation of prices requires interpretation of users' preference and requirements. Thus, the question arises as to how and where users' preferences are interpreted in the network. Chellappa[10] describes 'cloud-computing' as a dynamic computing paradigm where the boundaries of computing are determined by rationale provided by technological, economic, organizational and security requirements. Using this computing paradigm, we propose that the priority pricing information can be implemented at the following three different levels (or a combination of them) depending upon the nature of the network and where the network management wants to interpret the required information:

(1) Application level: Applications can be designed to use pricing information by creating an external access control. For example, many organizations today have adopted the use of a secure Web server that uses the HTTPS protocol with certificates issued by internal certificate authorities. This involves presenting digital certificates to the Web server to authenticate the client. The pricing information can be included in such a certificate so that the priority can be determined at the application level. Similarly, in a network that employs a third party authentication mechanism, such as Kerberos,[7] the pricing information can be 'carried' on the ticket that is issued.

(2) Node level: Examples of nodes on a network are gateways, firewalls, and proxy servers. Firewalls in Intranets determine the path to be followed by packet based on, for example, its origin, and type. For instance, a firewall may disallow packets from certain domains into its internal network. Along similar lines, a specialized gateway could be embedded with rules to determine what a priority of a certain packet would be based on its origin and type. Node level implementations may work with *Packet level* implementations to provide more flexibility and user level control while maintaining organizational objective as its overall resource management goal.

(3) Packet level: Researchers in the area of active networks have provided guidelines to including 'byte-code' and other self executing programs at the packet level. The 'byte-code' can be dynamically used to determine the best network service options. In a network using pricing as a network management tool, the 'byte-code' will optimize its use of network resources by using the combination of service quality requirements, cost, and the available budget.

The main objective of the above classification is to describe how the networks can utilize existing protocols and yet manage their resources efficiently, such as with Application level and Node level control. As the networking technology moves into the realm of active networks and mobile programs, the integration of Packet level implementations will enhance the Intranets' capability of catering to its users better by letting them decide the best alternatives for themselves while imposing organizational constraints of resource allocation.

The central objective of this paper is to bridge the gap that exists between economic pricing approaches and networking technology by proposing an economic mechanism* to facilitate resource management from a business perspective. Design of such a mechanism requires careful consideration of available tools, technology, and information. The mechanism presented in this paper is illustrated with a complete set of Intranet management tools, including data security and integrity, dynamic resource allocation, task prioritization, and enforcement of organizational resource constraints.

The paper is organized as follows. The next section provides the background on a dynamic pricing approach that values resources according to current demand and valuation. The third section discusses the technological and operational requirements of implementing a pricing mechanism. The fourth section provides the description of transaction flows with each of the three levels of pricing implementations discussed above. The final section concludes this paper with a brief summary and directions of future research.

Dynamic Pricing as a Network Management Tool

There has been an explosion in the number of commercial organizations joining the Internet particularly in the form of the World Wide Web (WWW). These serve not only as outposts for business-to-consumer interaction but also provide a platform for intra-organizational services for employees. Organizations that have already invested substantially in proprietary data networks are in the process of adopting more open standards (such as the Internet-based technologies) in order to link functional and geographically dispersed units for information sharing. The goal of these Intranets is to harness the Internet and its technologies as a medium for achieving its organizational goals and objectives. These are often achieved by creating collaborative tools and workflow systems designed to operate in the Intranet setting.

However, mere technological innovations will not provide the required competitive edge to organizations unless the resources are used effectively.

*An economic mechanism is implementation of an economic approach.

Intranets may be thought of as providing the infrastructure for intra-organizational electronic commerce.

Intranets may be thought of as providing the infrastructure for *intra-organizational electronic commerce*. While organizational units need to cooperate in achieving the overall goals of the firm, they also have to compete in terms of computational resources to attain their expected performance levels. Therefore, a network management approach for the efficient usage of Intranets and it resources should encompass the following:

- maximal benefit from Intranet resources;
- security and integrity of information;
- availability of information on the status of services and servers;
- appropriate translation of 'traditional' business protocols, including differential data handling, prioritization of organizational tasks, etc., and
- flexibility in implementation to make it adaptable to constantly changing technology (adhering to open standard).

In this section, we address the issue of maximizing organizational benefit of network resource usage. Several researchers have pointed out that economic pricing approaches could achieve the maximization of benefits by forcing the optimal use of the network elements.[1–4] However, many of the pricing approaches fail to account for the most important dimension of managing the network traffic: the necessity to perform this task in real-time, since by definition the data communication services are time constrained, and the demand structure is quite dynamic.

The model for priority pricing for Internet traffic presented in reference 1 goes beyond assessing the theoretical pricing and provides a mechanism to compute the prices in real-time. The key concept being that, there is a charge to a user in an amount that is proportional to his/her usage of the network. Specifically, the paper argues that a user should pay charges equal to the amount that other users suffer as a loss due to degradation in their service. Priority pricing provides economic rationale for providing higher quality of service to the users who need it. In this model a rental price, a priority premium, and an expected waiting time are associated with each network element. The prices are dependent upon the size of services desired. Service requests from a user depend on the service benefits to the user and the costs. Given the prices and the anticipated waiting times, a user evaluates and selects cost-minimizing service schemes. Optimal service demands are then translated into demands on individual processors.

In reference 1 authors prove that a generically unique benefit-maximizing allocation exists, and derive the rental prices that support this optimum as a 'stochastic equilibrium'. That is, (i) user flow rates are optimal for each user given the rental prices and anticipated waiting times, (ii) the anticipated waiting times are the correct ex ante expected waiting times given the flow rates, and (iii) the aggregate average flow rates are equal to welfare-maximizing rates. The advantage of this characterization resides in the fact that this pricing mechanism can be implemented in a decentralized manner with relatively smaller information overhead (as compared to traditional fixed-point price computation approaches in economics). This allows the price computations to be decentralized to each server and users' decision to be decentralized to their client machines. Additionally, priority pricing can take urgent needs into account and provide a rationale for providing faster access to some users.

The objective function in this model maximizes collective benefits of the system and its users. This is a natural objective function for the Intranet environment where it is rational to assume that organizational objective is to maximize the benefits derived from Intranet usage. From the theoretical standpoint these results have significant importance. The rental prices at the Intranet nodes decentralize the management and accounting problems. It gives users or their clients access to an evaluation mechanism to decide when and what kind of service they want and at what priority.* At the server level it will allow the management to assess the queues and delays more accurately and design their systems accordingly. Finally, at the network level it will allow for a better load distribution because users will avoid excessively loaded and thus highly priced servers.

To understand the requirements for price computations we provide a brief description and interpretation of the mathematical model. The price at a particular server for a particular priority class can be represented by the following system:

$$P_{mk}(q) = \sum_h [\partial\Omega_h/\partial\chi_{mkq}] \sum_i \sum_j \delta_{ij} x_{ijhm} \qquad (1)$$

where:

$P_{mk}(q)$ is the price of a job sized q at server m for priority class k

χ_{mkq} is the arrival rate of jobs sized q at machine m in priority class k

Ω_h is a continuously differentiable, strictly increasing function of arrival rate χ_{mkq} and capacity v_m; it provides the waiting time at a server m for priority class h

δ_{ij} is the delay cost parameter of consumer i for service j

x_{ijhm} is the flow rate of service j for consumer i with priority h at server m.

The first term on the right side $[\partial\Omega_h/\partial\chi_{mkq}]$ is the derivative of waiting time with respect to the arrival rate of jobs sized q. Since the waiting time is a strictly increasing function of this arrival rate, an increase in the arrival rate of a certain priority class increases the prices for that priority class. The second $\sum_i \sum_j \delta_{ij} x_{ijhm}$ can be interpreted as the accumulated delay cost of the system; an increase in this cost increases the price. Since the jobs in the highest priority class impose delays on the jobs in all other priority classes, whereas the jobs in lowest priority classes impose very little delay on the jobs in other priority classes, the prices for higher priority classes are higher than that of lower priority classes.

However, these prices cannot be optimally computed since that requires the knowledge of optimal demand. The approximate prices can be computed in real-time by an iterative mechanism, where the parameter estimates can be based on the information obtained from the time-based polling of network elements. Suppose $(t, t+1)$ is the time interval between polling. The following iterative equation can be used to update the prices at any

*By a priority class we mean that jobs in the highest priority class are processed before all the other jobs. At any time if a higher priority job (than the rest in the queue) arrives it is put first in the queue. Thus, jobs in the highest priority class impose delays on the jobs in all other priority classes, whereas the jobs in lowest priority classes do not impose any delay on the jobs in other priority classes.

given time $(t+1)$:

$$p_{mk}^{t+1} = \alpha \hat{p}_{mk}^{t+1} + (1-\alpha) p_{mk}^{t} \qquad (2)$$

where
α is a number between (0, 1)
$\hat{p}_{mk}^{t+1}$ is the estimated new price at time $(t+1)$ using equation (1)
p_{mk}^{t} is the implemented price during the time $(t, t+1)$

Updating the prices this way provides a shield against local fluctuations in demand and in the stochastic nature of the process. A lower value of the parameter α means that the price adjustment will be gradual in time, whereas a higher α will result in potentially large changes in the prices from period to period. Note that, according to equation (1) we are only required to obtain data at each individual network element in order to set prices for that network element.

The implemented prices at each network element are not 'optimal' from a theoretical standpoint since they are based on transient data. However, telecommunication networks rarely (if ever) are in a 'steady-state' and time-variant prices may actually provide better performance than fixed prices that may be based on irrelevant historical data.

The question then might be: Why propose a complicated pricing mechanism at all, why not just monitor the traffic and implement perhaps an aggregate volume based pricing? In fact, most pricing implementations related to the Internet are of this form, such as in the case of fixed access or trunk fee pricing. The problem with this approach is that fixed fee or any aggregate fee does not have any capability of affecting the usage behavior. At best these pricing schemes provide a poor rationing mechanism. In our opinion, a theoretical base for updating prices provides capability of ''tuning'' the prices to correct order that has a benefit maximizing behavior implication on network usage. Figure 1 provides an evidence of this claim of effectiveness of *theory-based* pricing approach as compared to no pricing and fixed pricing approaches. These results are obtained from the simulation testing of this pricing scheme.* These results present the system benefits (net benefits) and private benefits (user benefits). Net benefits are the benefits to the system as a whole and are equal to the aggregated users' value of the services rendered less the loss of value due to the delay, whereas the customer benefits are the net benefits less the price charged by the system.† Clearly, the benefits derived from using our pricing scheme are far superior to those obtained by flat prices or free access. Flat prices do provide higher net benefits than free access but the user benefits are almost identical. More surprisingly, results in Figure 1 suggest that using the pricing mechanism may be a win–win situation on the aggregate level, i.e., both users and the Intranet service providers (such as partner routers) may benefit from using appropriate prices. Users benefit

*The details of this simulation study and a variety of other results related to this pricing mechanism are presented in reference 6.

†Note that in the free access case net benefits = customer benefits.

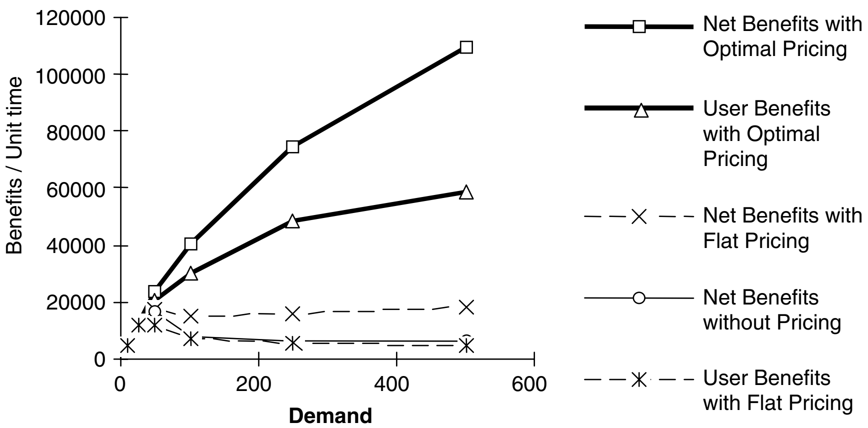

Figure 1. Benefits with different pricing schemes

because their requests are delayed less, resulting in the availability of timely and therefore useful information. The Intranet benefits because of the revenues generated and the optimal usage of the network.

Another important benefit of using dynamic priority pricing is the ability of the pricing mechanism to appropriately replicate the characteristics of traditional business processes.

Another important benefit of using dynamic priority pricing is the ability of the pricing mechanism to appropriately replicate the characteristics of traditional business processes. Ideally, Intranet applications should functionally provide same flexibility (and sometimes restrictions) as provided by the *usual* business protocol it is replacing. For example, one of the most important dimension of information dissemination in an organization is the ability of human operatives to provide differential data handling—some tasks are more important than the others and need their informational requirements met first, i.e., prioritization of tasks. A mechanism that eliminates a traditional business transaction should be able to replicate the important characteristics of that business transaction, such as appropriate priority handling of some information. With current Internet technologies some of the desired qualities of business transactions, such as differential data handling, cannot be replicated.‡ Clearly, appropriate layers of data handling have to be constructed on top of the existing communication protocols besides the development of the applications which support differential data treatment instead of *democratic* best-effort treatment of present.

This priority pricing mechanism can be used in internal electronic commerce environment for automated resource management. Furthermore, since this pricing mechanism maximizes the net benefits derived from the system, it is consistent with the corporate goal of maximizing the benefits from Intranet operations.

‡The next generation Internet Protocol (IPv6)[6] may have priority handling, however, the discussion seems to be geared towards providing application based priority levels rather than usage (or user-need) based priority assignments. An appropriate pricing mechanism, thus, will still be needed to provide need-based priority access.

Computational Requirements for Implementing Pricing

The computational requirements of the different pricing schemes will vary. Figure 2 describes a fictional usage of a pricing system. The illustration points to several requirements of using a pricing system. In this method, either the user explicitly specifies the service he/she wants (routine tasks) or the pricing implementation chooses a service description (what information to access, from which server(s), etc.). Routine tasks can be specified at the user interface level by enhancing client applications to include these options. The pricing implementation scheme has to include a service construction module that can interpret the users' requirements into specific computational tasks. The complexity of such a service construction module depends on the scope

User: *I want to find everything about XYZ method in easy to understand multimedia format*.

Pricing Implementation: *How soon would you want this information*?

User: *In the next half-hour*

Pricing Implementation: *OK! Let me give you the options*

[Pricing Implementation will compute the costs of delivering the information in various formats using different computational and network resources]

1.0 *Complete multimedia (audio/video/Text): cost 200 units—delivery 30 minutes*

2.0 *Audio and Text: Cost 120 units—delivery 30 minutes*

3.0 *Text: cost 35 Units—delivery 30 minutes*

4.0 *Cheapest complete multimedia: cost 95 units—delivery 5 hours (estimated)*

And so forth...

User: *How about delivering the complete multimedia in 2 hours*.

Pricing Implementation: *OK! Let me give you the options...*

Figure 2. Illustration of using a pricing system

of the Intranet applications and organizational goals. While this module will be an integral part of application and node level implementations, in active networks users (who can dynamically assess their needs) will determine this aspect.

Price computation requires the assessment of parameters described above at each network element and then the computation of performance prediction.[1] This pricing information is updated periodically or when a demand pattern shift is detected. However, since the pricing decentralizes the resource allocation mechanism there is no need for a central authority to conduct the estimation task. The prices can be computed at each network element independent of when and how the parameters are estimated at other network elements. However, without active networks where routers do minimal computation the job of price estimation may be left to a central authority, especially with node level (gateways, firewalls, etc.) implementation. In such cases, a Topology Management Application (TMA), such as IBM's NetView™ or HP's OpenView™, may be used to periodically poll the network and update the pricing information. The prices and performance predictions can then be stored in centralized databases in case of application level implementation and at gateways and firewalls in case of node level implementations.

With active networks, the parameter assessment, performance prediction, and price computation can be left to the mobile programs or agents. Further, there would be no need to create a centralized database for performance predictions and pricing information, since each network element can store its own information. Furthermore, the price and performance predictions can be made a part of the routing tables if the pricing is also used for determining the optimal cost routes or optimal cost dynamic multicasting. Thus the task of price computation and resource allocation becomes completely decentralized, while prices still enforce near-optimal usage of network resources from the organizational perspective. Figure 3 provides a generic view of computational requirement of a pricing implementation.

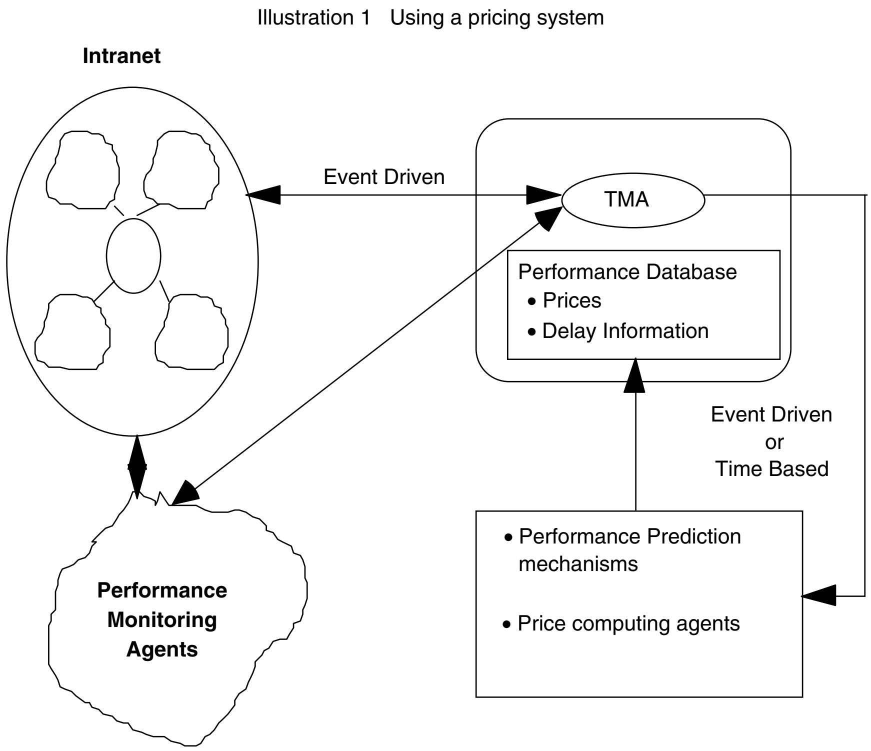

Figure 3. Computing prices

Transaction Characteristics and Pricing Implementations

In this section we illustrate how a transaction may be handled at each of the three levels of the pricing implementation. For the purpose of exposition, we break down the transactions into the following parameters reflecting the different dimensions of a transaction: type and quality of service, authentication and data integrity, acceptable delay or deadlines, and allowable spending limit.

Let us illustrate the role played by these parameters with the help of an example. Consider an Intranet user in California, who wishes to print a document on a printer located in his firm's subsidiary in Malaysia. The type of service requested in this case is that of printing. There could be many printers in the subsidiary each with its own 'price', i.e., the 'cost' of printing the document on a printer with a higher output rate and better resolution may be greater than its inferior brethren. When the user initiates such a transaction, the type of service can be determined by the application invoking it, e.g., the print module of an application. Authorization or permission to schedule such a job needs to be verified based on the user's identity, level of authorization, etc. Further, the user specifies an acceptable delay that he is willing suffer for the completion of his print job, reflecting his urgency. A fourth parameter is the 'maximum price' that a user is willing to pay to get his job completed. Such a transaction may be regulated at any combination of the three levels described later in this section. Before discussing the regulation of a job let us analyze each dimension of a transaction in detail.

(1) Type of service: A service maybe a set of specific well defined processes required to attain the user's goal or they could constitute combinations of generic processes that a user defines dynamically. Common services are often associated with applications, such as a print module of a word processor that invokes a print service, email clients invoking SMTP requests, etc. However, the final form of a service could be a result of combinations of generic processes, e.g., a user printing a document that needs to be converted to a specific printer format and then queued for printing. Such a service can be thought of as a combination two services namely format conversion and print queuing. Another dimension of service types is the quality requirement. For example, a user may be flexible on the quality of audio/video transmission during remote conferencing, however, she might want to enforce a certain *minimum* quality. A detailed treatment of Quality of Service (QoS) considerations is provided in references 11 and 12.

(2) Authorization and Data Integrity: The scheme for authorization may vary across organizations. For example, the IP address of a client machine could be used to construct simple authorization rules in firewall implementations. On the other hand, protocols designed for security, such as Kerberos or SSL, may be implemented to ensure both security and permission levels for users. The method employed for this parameter can be efficiently utilized for implementing the pricing scheme as described earlier.

(3) Acceptable delay: This reflects the urgency desired by a user for a particular service. An average user is not concerned with the specific location of a delay (i.e., at the processors or during data transmission) but is rather concerned about the aggregate delay that includes both processing and network delays. Thus when a user specifies an acceptable delay period, the pricing scheme will take into account the network congestion (therefore the manner in which it will route the packets) and the load at the server end. End-user applications can provide the user with simple interfaces to specify the level of acceptable delay. In fact many clients today already have provisions for including such a parameter. For example many email clients today have an extended MIME type X-priority to indicate if a mail is 'normal' or 'urgent'. It is therefore conceivable to accommodate such modules in any end user client. Depending upon the type of application, delay could be specified in generic terms, such normal, urgent, or no-delay, or it could be specified as a time deadline. For example, users can specify that a service (a) should be

performed in real-time (no delay is tolerable), (b) has a hard or non-negotiable deadline (mandatory completion by the deadline), (c) allows a soft or negotiable deadline (if the request is not completed by the deadline, the user may be able to negotiate another deadline, perhaps at a lower cost); or (d) has no delay restriction (best effort). The granularity of this specification depends upon specific implementations, desired functionality, and organizational goals. Note that a delay can also be mapped into QoS requirements by including more dimensions to the delay specifications, for example by specifying the tolerable variance in intra-stream packet transfer rates.

(4) Allowable spending limit: This is a reflection of the user's perceived value for the services he/she is interested in. This does not translate to the 'amount', the user will be charged but rather indicates the level to which the user is willing to pay for the services. *If the service cannot be provided within the limit specified by the user, then the usage of the Intranet resources cannot be economically justified*. The overall limit for a user as well as the allowable limit for particular services is a function of the organizational structure. For example, a senior manager of a group will perhaps have a greater spending limit, than his/her subordinates.

If the service cannot be provided within the limit specified by the user, then the usage of the Intranet resources cannot be economically justified.

Note that in the case of two competing jobs, it is not the job with higher willingness to pay that necessarily gets executed first, rather it is an overall function of the acceptable delay, service quality requirements and the willingness to pay. Consider the case where two jobs have identical service requirements with job **A** having higher willingness to pay than **B**, but **B** having a tighter deadline. According to our pricing model, described above, the optimal choice might be to give **B** a higher priority than **A**, since the loss in organizational benefits may be higher if the job **A** was given a higher preference. Therefore, spending limits can be determined by the users themselves to evaluate whether there is any economically justifiable option to obtain a given service.

The following section delves into the details of the pricing implementation scheme incorporating the above parameters. Note that the overall implementation of pricing can be a combination of the following methods and the interoperability of these has been discussed wherever it is deemed appropriate.

—Application Level Implementation—

Incorporating pricing information at the application level serves a dual purpose. First, the pricing information such as authorization, delay and spending limit can be 'ridden' on top of application-specific information. For example, applications that use Kerberos for authentication, the tickets issued by the trusted server can serve the dual purpose of carrying pricing information. Second, individual server loads can also be calculated at this level allowing for redirection of process requests.

Figure 4 provides a sketch of transaction flows with application level implementation. Price computation is done in a centralized or decentralized manner as discussed above. However, the task of optimizing costs for each service request is performed by the client itself, i.e., the client receives the pricing information from a centralized database or the service construction module and chooses the best option. Note that, there is a need for some centralized control for the distribution of pricing information.

—Node Level Implementation—

Nodes such as gateways, firewalls, and proxy servers operate on a basis of rules. The pricing approach, discussed earlier, provides a set of rules for prioritizing packets. These rules may be static or constructed dynamically. Static rules that translate

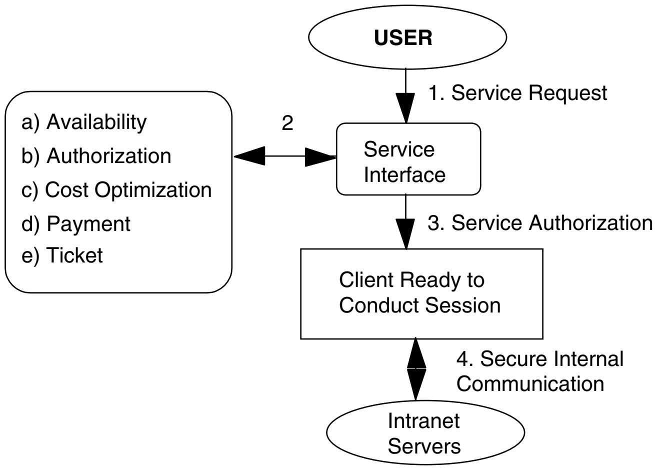

Figure 4. Application level implementation

organizational protocols (e.g., hierarchical structure) can be built into these nodes. When a packet arrives from a certain client machine, these nodes can invoke the default rule set for the packet. For example, when a user of an Intranet has not set any specification for a particular transaction, the nodes can associate the packets arriving from the client machine with the default authorization, acceptable delay and allowable spending limit for that user. This allows for actions to be economically justifiable even when the user has not set any specific parameter.

Note that node level implementations are highly centralized (when used in isolation) and require that these nodes have complete information regarding the sub-network they are protecting or controlling. Thus, the price computation will be done in a centralized manner with help of a TMA's polling mechanism (see above). In addition, the node itself computes decisions regarding the resource allocation rather than the user.

Therefore, a user has to provide his/her preferences (parameters discussed above) to the decision-making node. While, in theory, this may create a problem because users might not be willing to reveal their true preferences, Gupta *et al.*[13] have developed computational schemes that allow revelation of their complete set of preferences from the service type and quality specification.

With active networks, node level pricing applications can be used when organizational goals dictate a greater control over the computational resources. In such cases, mobile programs can provide their choice of resource allocation based on the pricing data available from the nodes, however, node level control can accept, modify, or reject any user request.

—Packet Level Implementations—

Active networks allow individual user, or groups of users, to inject customized programs into the nodes of the network.[14] Commonly these programs are incorporated in the form of byte-codes (e.g., Java) to be interpreted at the level of routers. This capability can provide the maximum amount of flexibility in user control over their service requests. However, as discussed earlier this also has the potential for creating more deadlocks and severe resource allocation problems if the active packets are allowed to obtain their resources purely from a myopic perspective. The role of pricing is such networks is to enforce the organizational objectives while letting the users make their decisions myopically. In our setting, the mobile programs will carry a complete set of service characteristics and decide on the appropriate resources based on the pricing information from all the desired network elements, dynamically.

The active network technology will also allow complete decentralization of price computation and implementation as described in the third section. The mobile programs in active networks will not only be able to customize the service quality according to user specification but will also be able to dynamically adjust their service quality requirements. This will also obviate the need for full-fledged service specification and service construction, since after every subset of computation, the mobile programs can exercise one of many options such as: obtain next set of information, continue computation, or abort and return the current results to the user. Mobile programs can also be used to create dynamic selective casting routes to 'broadcast' organizational wide activities in most efficient manner. For example, at any given time if the computational load is high on a certain part of the Intranet, mobile programs may choose to selectively broadcast to only the nodes where it is economical to do so and can delay the broadcast to other nodes for a later time.

Summary

The focus of this paper is to provide a solution for active management of Intranet resources including those created by active networks. Our approach allocates network resources in real-time based on the relative valuation of organizational tasks while maximizing the benefits of Intranet usage. This is achieved by using an adaptive pricing scheme for network resource allocation and it includes both server and bandwidth level resources. We discuss the pricing implementation at three different levels reflecting the scalability of the approach from current technologies to future innovations such as active networks. With current technology, the pricing scheme, in conjunction with performance prediction (forecasting) tools update a Intranet performance database in real-time to facilitate the optimal decision for the user service requests. With future technologies such as active networks it will be possible to completely decentralize price, implementation and usage while still preserving the optimal resource allocation characteristics of pricing. Figure 5 depicts the degree of decentralization and user control with and without active networks that can be achieved via price implementation.

The pricing scheme also provides the differential data handling by using priority classes. A simulation of the pricing approach conclusively shows that significant benefits can be derived by both the users and the Intranet compared to a free access or a flat pricing scheme. The active network technology, in isolation, has the potential to offer a high degree of user control over their network service needs. However, it also threatens create resource allocation problems if the user requests are not properly regulated. Static regulation of network resources using the polling systems will not be able to handle the network management tasks, especially in case of Intranet management where certain organizational constraints on the resources have to be enforced. Using dynamic pricing to regulate the resource allocation can provide an efficient and scalable way of managing network resources.

	Without Active Networks	With Active Networks
Decentralized	Application Level	Packet Level
Centralized	Node Level	Packet Level with Node level

Figure 5. Control level with and without active networks

References

1. Gupta A, Stahl DO, Whinston AB. A stochastic equilibrium model of Internet pricing. *Journal of Economic Dynamics and Control* 1997; **21**: 697–722.
2. Shenker S. Service models and pricing policies for an integrated services Internet. In *Public Access to the Internet*, Kahin B, Keller J (eds). Prentice-Hall: Englewood Cliffs, NJ, 1993.
3. Wang Q, Peha J, Sirbu M. The design of an optimal pricing scheme for ATM integrated-services networks. In *Internet Economics*, McKnight L, Bailey J (eds). MIT Press: Cambridge, MA, 1997; 353–376.
4. MacKie-Mason J, Varian H. Pricing the Internet. In *Public Access to the Internet*, Kahin B, Keller J (eds). Prentice-Hall: Englewood Cliffs, NJ, 1995.
5. Shenker S, Clark D, Estrin D, Herzog S. Pricing in computer networks: Reshaping the research agenda. *Journal of Telecommunications Policy* 1996; **20**(3): 183–201.
6. Deerin S, Hinden R. Internet protocol, version 6 (IPv6) specification, *Technical Report*, IETF. Available from: http://www.globecom.net/ietf/rfc/rfc1883.shtml, 1995.

7. Kohl J, Neuman BC. The Kerberos Network Authentication Service (V5), *Internet RFC* 1510. Available from: http://www.cis.ohio-state.edu/htbin/rfc/rfc1510.html, 1993.
8. Tennenhouse D, Wetherall D. Towards an active network architecture. In *Multimedia Computing and Networking (MMCN 96)*, San Jose, CA, 1996.
9. Chellappa R, Barua A, Whinston AB. Intranets: Looking beyond internal corporate web servers. In *Readings in Electronic Commerce*, Kalakota R, Whinston AB (eds). Addison-Wesley: Reading, MA, 1997; 311–321.
10. Chellappa R. Cloud computing—emerging paradigm for computing. In *INFORMS 1997*, Dallas, TX, 1997.
11. Gupta A, Stahl DO, Whinston AB. Priority pricing of integrated services networks. In *Internet Economics*, McKnight L, Bailey J (eds). MIT Press: Cambridge, MA, 1997; 323–352.
12. Gupta A, Parmeshwaran M, Stahl DO, Whinston AB. Streamlining the digital economy: how to avoid a tragedy of the commons. *IEEE Internet Computing*, forthcoming.
13. Gupta A, Jukic B, Stahl DO, Whinston AB. Designing an incentive compatible mechanism for internet traffic pricing. under review in *Informs Journal of Computing* 1997. ■