Regular Paper

Rating Oriented Distributed Information System for High Quality Autonomous Service Provision and Utilization

Xiaodong Lu,† Ivan Luque† and Kinji Mori†

The adaptable and efficient infrastructure for popular information provision and utilization is very important in the distributed information service system. In this paper, we introduce a rating oriented distributed information system sustained by push/pull mobile agents to cope with the rapidly changing of providers and users' requirements. Based on this environment, an autonomous information allocation technology is proposed, to put the most popular information near to users. Moreover, when users' demands change, an autonomous information reallocation technology is proposed to achieve load balancing and guarantee that users can always get the information from a certain pathway. We proved the effectiveness of the proposed system through the simulation and comparison with the conventional system. Moreover, advantages through applying autonomous information reallocation within the system is quantitatively shown.

1. Introduction

Advances in communication technologies and the decreasing costs of computers have made distributed information systems an attractive alternative for satisfying the information service needs of global users. In addition, the demand for information services is increasing at an explosive rate. As a result, the needs to cope with rapidly changing user demands, network status and information contents are becoming more stringent and complex[1)].

Users have a specific tendency to utilize information services. The study of the users' access characteristics manifests that popularity of information utilization generally exists in many information services[2)]. Among all kinds of e-commerce applications, this paper focus on online music site. The system provides popular songs on the Internet, and users access the system for download. The characteristics of this system are as follows. First, the majority of users need only small amount of information. For example, imaging system provides top 10 popular songs on the ranking, few users require more than 2 songs at the same time. Second, the majority of users only need popular information. In other words, users' access clarifies the existence rating of songs in the system.

In this paper, we introduce a new system architecture called Rating Oriented Distributed Information System (RODIS), which sustained by push/pull mobile agents. In this system, the service provider allocates its popular information to adjacent nodes recursively, closer to users. The characteristic of RODIS is to balance the cost of information allocation performed by push mobile agents and the cost of the access to the information performed by pull mobile agents.

In a distributed information service system, the most important problem is how to propagate information from the service provider (SP) to nodes in the system. Consequently, when users preferences change, the system that originally was designed to have balanced load may lost this balance, as congestion might arise on some nodes because of the convergence of changed access. Hence, this raises another important problem: how the system copes with the change of users preferences. This paper proposes autonomous allocation and reallocation techniques based on the current situation of the locality to achieve load balancing and eliminate contradiction in rapidly evolving situations.

The remainder of the paper is organized as follows. In the next section, the concept and architecture of RODIS is presented. Autonomous information allocation and reallocation technology are presented in Sections 3 and 4. The simulation results in Section 5 show that the proposed technology improves access time to compare with conventional system. Section 6 summarizes the related work and the last section concludes the paper.

† Tokyo Institute of Technology

2. Rating Oriented Distributed Information System

Under dynamically changing situation, information service systems are characterized by continuous changes of services from service providers promoting their offers and changing users' demands[3]. In such system, service providers require to provide non-stop services with timely update, and at the same time, users require timely access to information services corresponding to their individual needs.

Based on the concept of Autonomous Decentralized System (ADS)[4], Rating Oriented Distributed Information System is proposed to meet both users and providers' heterogeneous requirements. The components in the system coexist and cooperate in an autonomous way to achieve adaptability and timeliness for information service provision and utilization[5].

2.1 System Architecture

The main goal of the RODIS is to guarantee the assurance of autonomous information services provision and utilization. This architecture is based on the rating oriented replication of information services to warrant services availability and responsiveness.

The system architecture is illustrated in **Fig. 1**. Due to the characteristics of the rating of the users' access, popular information is allocated near the users and more replicas are created in the system because of the possible access preference caused by its popularity. As a result, the multi-level distributed information service area is created. Hence the response time of majority users who access popular information is improved. Users with different requirements for information can be satisfied at different levels in the RODIS. Consequently, the cost of service utilization (access time) and provision (update) are balanced by allocating closer to the majority of users the most accessed part of the information services.

The process of information provision by SPs as described above, is called push technology. The process of searching information by users is called pull technology[6),7]. We propose Push Mobile Agents (Push-MAs) for distributing information and Pull Mobile Agents (Pull-MAs) for searching information on the network as shown in Fig. 1.

Mobile agent technologies are utilized in RODIS as the medium for a dynamic interaction between information service provision and utilization. In the RODIS model, each action is given to a mobile agent that bridges the autonomous entities. A reliable mobile agent platform is available on each node, providing an execution and routing environment for mobile agents. In one word, node, Push-MA and Pull-MA are three autonomous subsystems, mainly responsible for information storing, allocation and utilization, respectively.

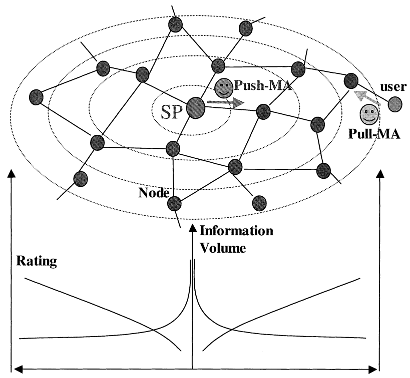

Fig. 1 RODIS architecture.

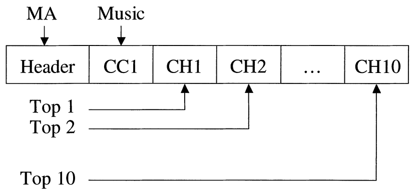

Fig. 2 Message format in the RODIS.

Every subsystem has autonomous controllability over its own operations and autonomous coordinability with the other subsystems to continue its operations even under evolving situations[8]. Information provision and access in the RODIS is based on content code communication to be described in the next subsection.

2.2 Content Code Communication

In highly changing environment, the state of nodes, the stability of connections, the status of SPs are highly unpredictable and maintaining repository of the SP addresses cannot guarantee high-assurance of the system[9]. In the RODIS, information contents are uniquely defined by *Content Code* (CC). The detail of contents and popularity are further specified by its *Characteristic Codes* (CHs). For instance, the CC identified by CC1 standing for Online Music can be further specified by the CHs: from Top 1 to Top 10 as shown in **Fig. 2**.

The SP sends Push-MAs that carry information specified by CC and CHs to allocate the most accessed segments of information from node to node on the network. The node stores a certain amount of less popular information, like Top 10, and fades it from the Push-MA and then sends it to lower nodes (see Fig. 1). Users search information by deploying Pull-MAs with certain CC and CHs, for example, Online Music and the song of Top 3. And then Pull-MAs process at a node or move to upper level, which means receiving the Push-MAs from those nodes stored the less popular information, according to the CHs stored for the requested CC.

3. Autonomous Monitoring and Information Allocation

3.1 Autonomous Monitoring

Each node in the system monitors local situation and status brought by a Push-MA or Pull-MA. When the Push/Pull-MA arrives at a node, the arrival time and coming direction are firstly to be recorded. Next, the request of the agent will be checked. The task of Push-MA is to allocate information, so the stored data on the Push-MA will be analyzed. After recording the version and volume of CC and CHs, it records the leaving time and direction of the departed Push-MA. It is similar with Pull-MA's case except for information required by the Pull-MA is checked in the monitoring process. Besides, total number of accessed Pull-MAs and satisfied Pull-MAs on the node are recorded respectively.

After receiving the Push-MAs, the node determines the routing information to the upper level that stored the less popular information. What's more, through monitoring the Pull-MAs, the change of the rating of stored information is observed. This will contribute to the adaptability of the system under the changed rating situation.

3.2 Users Preferences Measurement

The proper information volume stored on each node is determined by users preferences. How to measure users preferences is discussed as follows.

As shown in **Fig. 3**, at node D, between two Push-MAs update time t_1, *through rate*, $r(t_1, D)$, is defined as the ratio of the number of outgoing Pull-MAs proceeding to upper nodes $N_u(t_1, D)$ compared to the total number of incoming Pull-MAs at node D $N_t(t_1, D)$.

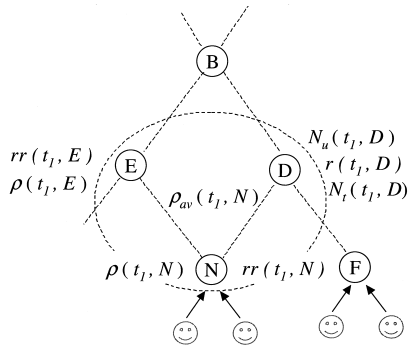

Fig. 3 Measurement of users preference.

$$r(t_1, D) = \frac{N_u(t_1, D)}{N_t(t_1, D)} \tag{1}$$

The information allocation process is regulated by the number of Pull-MAs circulating inside the system. However, each node has no view of the overall structure, we utilize the notion of locality congestion. The locality of a node N is defined as the set of all upper nodes having link with node N, and including node N itself, as shown in Fig. 3.

The Pull-MAs load in the system is therefore perceived per locality of a node. Each node measures the degree of the congestion based on the load of its locality in coordination with Push-MAs (which bring the load information from the upper nodes to lower nodes), and can asynchronously contribute to balance the load in the system by regulating number of Pull-MAs toward upper nodes. Through rate i.e., relative number of Pull-MAs that must proceed to upper nodes in the locality of a node N to achieve load balancing is called *reference through rate* $rr(t_1, N)$. A node can determine reference through rate based on the congestion in the locality. The degree of congestion of a node N depends upon node utilization $\rho(t_1, N)$, the proportion of the length of the waiting queue to the capacity of the node. Each node agent measures utilization $\rho(t_1, N)$, and sends to lower nodes through Push-MAs. The receiving node N determines the average utilization $\rho_{av}(t_1, N)$ of the nodes in the locality. Node N determines reference through rate as follows:

$$rr(t_1, N) = \frac{rr(t_0, N)\rho_{av}(t_1, N)}{\rho(t_1, N)} \tag{2}$$

Consequently reference through rate gives a relative measure of the number of Pull-MAs that are required to proceed to upper nodes to equalize the load on each node in the locality. Therefore, a node is required to allocate information in proper amount to satisfy the reference through rate.

3.3 Autonomous Allocation

In RODIS, users with different requirements can be satisfied at different levels. The information structure consequently permits to preserve the same access time to all unspecified users. Based on the load balancing of locality, the convergence of users access can be avoided and hence load of the system can be balanced.

In order to achieve above mentioned requirement, each node agent monitors the through rate and compares it with reference through rate to autonomously adjust information volume. If the node stored more information at the beginning, when users requests getting more and more, the node will become congested. Then through rate becomes less than the reference through rate, the node agent decreases information amount upon the arrival of Push-MA to bridge the gap between two rates. Consequently, information amount required at time t_2 at node N is proportional to the ratio of through rate to reference through rate at time t_1 of the node. A node agent determines information amount $I(t_2, N)$ at time t_2 at node N as follows:

If $r(t_1, N) < rr(t_1, N)$
then $I(t_2, N)$
$= I(t_1, N)\frac{r(t_2,N)}{rr(t_1,N)}\frac{I(t_2,UpNode)}{I(t_1,UpNode)}$

According to this process, a node agent only needs to know the information amount and utilization of upper nodes in its locality to adjust the information volume of its own to balance the load of locality.

4. Autonomous Information Reallocation and Contradiction Elimination

4.1 Autonomous Reallocation

The system is constructed according to the autonomous information allocation technology. But all these go well only under the condition that the real users preferences are roughly the same as originally predicted and no changes ever since. However, as matter of common sense, users preferences change and that are unpredictable. Dynamic load balancing through adjusting information volume on each node in autonomous information allocation technology partially solves the problem when the number of users who accessing the information changes but there is a limitation exists as it was explained in previous discussion.

As the information system is rating oriented, characteristic of the change of rating should be clarified before looking for a solution. Taking the total number of the users who access the system is constant while the number of users who requires every segment of the information changes is considered as a reasonable assumption. But no matter how quickly the rating changes, it is not difficult to understand that rating changes but gradually. Moreover, the rating of popular information changes faster than unpopular ones. This is obvious because popular information get more access in the same time interval.

Based on the factual understanding of the change of rating, it is reasonable to consider exchanging the information between the nodes in adjacent layers to cope with the new rating. Through exchanging information between neighboring upper-lower layers, popular contents are reallocated in lower-layered nodes, which represents more storage and more powerful process capacity. Less required information then withdraws to free unnecessary resources. Through this, step by step information in changed rating finds its place where it should belong.

As it is clarified that the problem becomes when and who trigger the information exchange process. Each node monitors the total number of accessed Pull-MAs and satisfied Pull-MAs and sends this up as a feedback in the locality after a certain timeout. The upper node analyzes the feedbacks to detect whether the rating changed between the contents stored on the adjacent layers. The node makes a decision on whether to trigger the information exchanging process or not based on the ratio

$$\gamma = \frac{N_{satisfied}}{\sum N_{lower\_satisfied}}$$

If $(\gamma > 1)$, the process of exchange is triggered.

After the node in upper layer decides whether to and what to exchange, it generates a regional exchanging Push-MA bringing the to-be-exchanged data to its lower node(s). The node in lower layer then processes the Push-MA, stores the brought data and generates another Push-MA in return bringing the exchanged data and sends up. Finally, the upper-

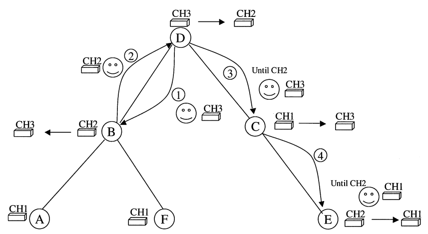

Fig. 4 Contradiction elimination.

layered node receives the returned Push-MA and so that data exchanging transaction is finished.

4.2 Contradiction Elimination

By autonomously performing this process step by step, every CH is reallocated to the new layer according to the changed rating but by local execution. However, users preference changes asymmetrically i.e., during a certain time interval, that all nodes in the same layer observe the same change in rating, hardly occurs in the real system. The autonomous action, which is taken by each node, results in inconsistency of information among nodes in the same layer. Under this condition, the node can do nothing but endure the congestion (see **Fig. 4**) for Pull-MAs may be wrongly navigated after the exchange. In order to guarantee every Pull-MA that can always get the required information on a certain path, there is a need to propose some mechanism to maintain the consistency in information reallocation process.

As shown in Fig. 4, node D receives feedbacks from node B and C that CH2 is accessed 5 times among 20 and CH1 is accessed 3 times. Node D is congested because of 16 accesses during the same time interval. Decision is to change the current situation to balance the load. For node D gets feedbacks on different contents, one among these will be chosen to begin with information exchange. This decision is again made by majority: node D monitors Pull-MAs requesting self-stored content, from every direction, suppose 12 Pull-MAs come from node B and 4 from node C, contents stored on node B will be chosen to exchange. This is because more users requesting CH3 passed by node B, more users get benefit after exchanging for shortened path to reach CH3. The sometimes-effective mechanism, wait for a timeout, is applied here. After the timeout, if the information stored on node B and D doesn't coincide, node D and B then begin the normal process of information exchange as mentioned before. But here, after the process, instead of deleting previously stored information (CH3), node D generates a special Push-MA taking that information to the other side, direction of node C. The Push-MA, taking CH3 to lower layers, alternates information on lower nodes and that on itself. This process will be performed successively until the information stored on some lower node is CH2.

By using this mechanism, users who access node B, D and C get benefits after information exchange for they are distributed in lower nodes where more replicas are allocated and additionally, their path to the requested information is shorten. But the users who were to access CH2 on node E may get unhappy for the opposite reason. And because the information exchange is done without knowing the number of these users, node D may be obliged to alternate the information with node B and C if that number is large enough; previous exchange process add only reallocation cost but advantages in timeliness and reliability of users' information utilization to the system. Moreover, the change of the rating of CH2 observed from node B to D makes this explanation more convincing: CH2 is getting more and more unpopular. Then we can formulate the method as follows:

current node stores CH_{local},
lower nodes store $CH_1, ..., CH_n$
$N_{lower-CH_i} = \sum N_{lower\_satisfied}$
$N_{lower-CH_k} =$
$Min(N_{lower-CH_1}, ..., N_{lower-CH_n})$
Information-exchange(CH_k, CH_{local})

Entirely, the proposed technologies, not only meet the timeliness and reliability of both SP's information allocation and users' information utilization, but also enhance the adaptability of the system.

5. Performance Evaluation

We have developed the Autonomous Information Service System Simulator (AIS3)[10] to prove the validity of the proposed system. The performance of the scheme has been compared with the results obtained in the case of proxy caching system and the effectiveness has been shown in access time improvement.

5.1 Initial Settings

Rating is observed in users preferences that contents have popularity. According to this analysis, and based on the consideration to simplify the situations, the settings of system and

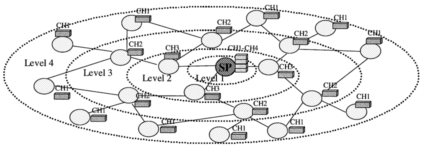

Fig. 5 The RODIS model.

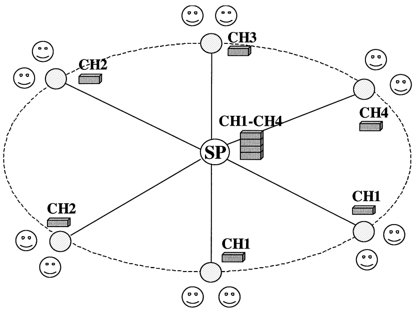

Fig. 6 The proxy caching model.

users preferences are configured as follows:

- SP provides 4 CHs: CH1, CH2, CH3, CH4 in the order of priority from high to low.
- User request but 1 CH in one access.
- Percentages of users who require 1 to 4 CH are 54%, 27%, 14% and 5%, respectively.

The data size of each CH is 5 MBytes and the link bandwidth between two nodes is 10 Mbps.

5.2 Simulation

The first experiment is to show the effectiveness of proposed system compared with proxy caching model. The system is constructed on the logical network. So a grid network spending 4-array connectivity to each node is applied. But the network topology for RODIS is random, i.e., various nodes having different number of connections with other nodes. After information allocation, the real network that the system uses includes 21 nodes and the total volume of information stored in the system is 24 CHs, 20 normal nodes stores 1 CH only and the SP stores all 4 CHs, as shown in **Fig. 5**. The proxy caching model is shown in **Fig. 6** and users' requests are assigned to each proxy server. For the fair comparison, the number of nodes and the total storage cost are same with that of RODIS. It has been proved that a caching proxy has an upper bound of 30–50% in its hit rate[11]. In the comparison the hit rate of 50% for each proxy server is applied.

The model of users preferences are realized in the experiment and users of changing number from 120 to 1,200 per second are sent to the two systems respectively. As results shown in

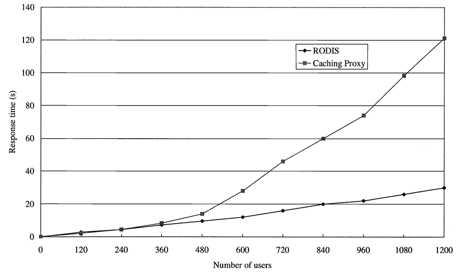

Fig. 7 Comparison of average response time.

Fig. 7, in proposed system, because system is constructed taking users preferences into consideration, users' accesses are distributed into the system. So timeliness in information utilization is achieved thanks to the balanced load in the system. When accessing users are getting more, nodes in proxy caching model get much more congested. Therefore, the average response time increases but exponentially. While in the case of proposed system, because RODIS leads users to the very node where it can actually be satisfied, the average response time increases but not as quickly as the situation in the proxy mode does.

In order to achieve load balancing of locality, additional information of upper nodes is necessary. In the RODIS, this information is brought by the Push-MAs, and the data size is much less than the CHs and network bandwidth. In addition, based on the proposed technology, the load of each node in the system is almost same, and at the same time, the network traffic is also balanced through the system architecture. So with the same refresh rate, the network traffic of RODIS is obviously better than the proxy model.

The second experiment mainly concerns the evaluation of autonomous information reallocation technology. Autonomous monitoring is a built-in function for the node. After configuring the changing users distribution, nodes behavior and refined result according to it will be clarified. To testify the effectiveness of autonomous information reallocation with contradiction elimination mechanism inside, the model of the change of users preferences is realized in the simulation. Users trend changes gradually from CH1, CH2, CH3 and CH4 to CH4, CH3, CH1 and CH2 eventually with the same total number of users, as shown in **Fig. 8**.

The result of comparison between the static system, which means system without adaptation to the change of rating, and the dynamic

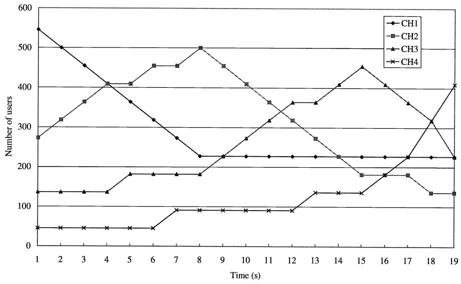

Fig. 8 Users distribution of changing rating.

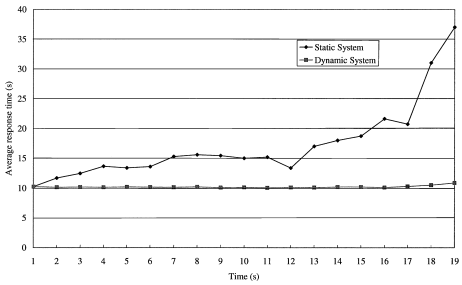

Fig. 9 Comparison of two systems.

system applied with autonomous information reallocation is shown in **Fig. 9**. Referring the access model shown in Fig. 8, a rapidly rising of the number of users who require CH4 is observed from time 15. Due to this the average response time of the static system increases rapidly. This is because users are navigated up to SP, where CH4 is originally allocated. In the case of CH2 and CH1, the response time is decreased but compared with that of CH4, the scale is different. The average response time in the dynamic system vibrates a little but keeps almost the same thanks to the proposed technology. Meanwhile, the cost of the allocation must be mentioned. During the process of information exchanging, which is an atomic transaction, Pull-MAs are made to wait on the lower nodes. This is because Pull-MA may fail to get the information previously on the upper node but newly exchanged to lower nodes.

6. Related Work

Data replication and placement has been a key technology in content delivery networks as well as web caching and web proxy services. The mirroring and server-based caching are main technologies in server-based replication, especially to determine what and where to replicate[12]. In Ref. 2), a demand-oriented server-based caching system is discussed to reduce network traffic. These technologies permit to reduce the processing load at the server, but cannot guarantee the load balance of the users' requests without implementing a centralized solution.

In the user-based replication mechanisms, hierarchical caching[13] and group caching[14] are proposed to reduce the client-side bottleneck. However, proxy caching is limited by the low level of sharing remote documents between users of the same local network. Additionally, user-driven caching cannot be applied to dynamic contents using CGI scripts to fetch customized information from database servers.

Network-based replication technologies have been emerged and became popular in recently. One of them is *Content Delivery Network* (CDN)[15] that allow service providers to distribute rich media content closer to their target customers. The other is *P2P* system that put in common the storage and processing resources of community of users. However, they mostly have the disadvantage to keep some centralized parts for accessing the requested services. For instance, the CDN is managed and monitored via the *Content Distribution Manager* (CDM) and Napster, one *P2P* system, has a central server indexing all the files available within the system and their corresponding storing peer's IP address. The adaptive nature of our approach inspired from ADS can meet providers and users' requirements in rapidly changing environment without any centralized management.

7. Conclusion

Based on the analysis of rating of users access, Rating Oriented Distributed Information System (RODIS) is proposed to satisfy the heterogeneous requirements of users and SPs in rapidly changing environments. Under this environment, the autonomous information allocation technology is proposed to assure the timeliness for SP's information provision and users' information utilization. In addition, to adapt the changes of users preferences, the autonomous information reallocation technology is proposed based on the autonomous monitoring and analysis of rating.

Thus, the RODIS adapts the constantly changing preference and assures that a Pull-MA can always be navigated to some node where it can get the required information. To evaluate the performance of the proposed system, we developed an Autonomous Information Service

System Simulator (AIS3). The simulation results demonstrate the effectiveness of proposed RODIS compared with the conventional system.

References

1) Mori, K., Yamashita, S., Nakanishi, H., Hayashi, K., Ohmachi, K. and Hori, Y.: Service Accelerator (SEA) System for Supplying Demand Oriented Information Services, *Proc. IEEE Conf. on ISADS*, pp.129–136 (1997).
2) Bestavros, A.: Demand-based Document Dissemination to Reduce Traffic and Balance Load in Distributed Information System, *Proc. IEEE Conf. on SPDP*, pp.338–345 (1995).
3) Ahmad, H.F. and Mori, K.: Autonomous Information Service System: Basic Concepts for Evaluation, *IEICE Trans. Fundamentals*, Vol.E83-A, No.11, pp.2228–2235 (2000).
4) Mori, K.: Autonomous Decentralized Systems: Concepts, Data Field Architecture and Future Trends, *Proc. IEEE Conf. on ISADS*, pp.28–34 (1993).
5) Lu, X.D. and Mori, K.: Autonomous Information Provision for High Response Time in Distributed Information System, *Proc. IEEE Conf. on IWADS2002*, pp.22–27 (2002).
6) Cybenko, G. and Brewington, B.: The Foundations of Information Push and Pull, *Mathematics of Information*, Springer-Verlag Press (1998).
7) Mori, K.: Autonomous Fading and Navigation for Information Allocation and Search under Evolving Service System, *Proc. IEEE Conf. on APSITT*, pp.326–330 (1999).
8) Lu, X.D. and Mori, K.: Autonomous Integration and Optimal Allocation of Heterogeneous Information Services for High-Assurance in Distributed Information Service System, *IEICE Trans. Inf. & Syst.*, Vol.E86-D, No.10, pp.2087–2094 (2003).
9) Arfaoui, H. and Mori, K.: Autonomous Navigation Architecture for Load Balancing User Demands in Distributed Information Systems, *IEICE Trans. Commun.*, Vol.E84-B, No.10, pp.1085–1093 (2001).
10) Lu, X.D. and Mori, K.: Autonomous Information Integration and Allocation in Agent-Based Information Service System, *Proc. IEEE Conf. on IAT2003*, pp.290–296 (2003).
11) Williams, S. and Fox, E.W.: Caching proxies: Limitation and potentials, *Proc. 4th Int. World Wide Web Conf.* (1995).
12) Ingham, D.B., Shrivastava, S.K. and Panzieri, F.: Constructing Dependable Web Services, *IEEE Internet Computing*, Vol.4, No.1, pp.25–33 (2000).
13) Rodriguez, P., Ross, K.W. and Biersack, E.W.: Improving the WWW: caching or multicast?, *Computer Networks and ISDN Systems*, Vol.30, No.22-23, pp.2223–2243 (1998).
14) Zhang, L., Floyd, S. and Jacobson, V.: Adaptive Web Caching, *Proc. NLANR Web Cache Workshop* (1997).
15) Liste, M.: Content Delivery Network (CDN) — A reference guide, *Tech. Rep., ThruPoint* (2001).

(Received May 6, 2004)
(Accepted November 1, 2004)
(Online version of this article can be found in the IPSJ Digital Courier, Vol.1, pp.46–53.)

Xiaodong Lu received the B.S. and M.S. degrees in electronic engineering and computer science from Lanzhou University, P. R. China, in 1997 and 2000, respectively. Presently, he is a doctoral student of the Tokyo Institute of Technology. His research interests include the high-assurance information systems and mobile agent.

Ivan Luque received B.S. degree in computer science and engineering from University Autonoma de Barcelona in Spain in 2001. In 2002, he joined the Tokyo Institute of Technology, as a mast student. His research interests are autonomous information systems and mobile agent technology.

Kinji Mori received the B.S., M.S. and Ph.D. degrees in the Electrical Engineering from Waseda University, Japan in 1969, 1971 and 1974, respectively. From 1974 to 1997 he was in System Development Lab., Hitachi, Ltd. In 1997 he joined Tokyo Institute of Technology, Tokyo, Japan as a professor. His research interests include the Autonomous Decentralized Systems (ADS), distributed computing and the mobile agent. He is a Fellow of IEEE and a member of IEICE, IPSJ and SICE, Japan.