Design and Implementation of Web-based Internet/Intranet Application Service Management System

Eun-Ju Ha, Su-Hyung Jo and Jong-Tae Park
School of Electronic and Electrical Engineering
Kyungpook National University
ejha@ee.kyungpook.ac.kr

James Won-Ki Hong
Dept. of Computer Science and Engineering
POSTECH
jwkhong@postech.ac.kr

Abstract

There exist various Internet/Intranet application services such as e-mail, ftp, news, telnet, archie, gopher and WWW, and new application services are continuously being developed. These services must be managed in order to provide secure, reliable and efficient services to users. This paper presents a framework for Web-based Internet/Intranet application service management system. As the proposed management system uses the Web technology, it can provide platform independence, mobility and friendly user interface. Using CORBA technology, the architecture can offer security, portability and scalability. It can also support multiple network management protocols and easy integration of new Internet/Intranet service management applications and the existing management applications. The Java technology is used to provide the interface of Web-based Internet/Intranet service management. Finally, we describe a prototype implementation and evaluate its performance.

[Keyword : Internet, Intranet, application and service management, WWW, Web-based management, CORBA, Java]

1. Introduction

There exist various Internet applications such as e-mail, ftp (file transfer protocol), news, telnet, archie, gopher, WWW (World Wide Web) [1] and so on. Internet/Intranet application services have been applied to improve the productivity of many organizations. The multimedia Internet/ Intranet application services of Mbone [2], Internet Phone, whiteboard [3] and Real Audio are being used widely by ordinary users. The electronic commerce (EC) [4], one of the new Internet application services, is spreading rapidly throughout the world as well. The need for a management system that can efficiently manage various new application services and multimedia services on top of existing services has been increasing.

Users ask for more reliable and effective services on the Internet/Intranet. The efficient management of Internet/Intranet application service is needed to satisfy these requirements. But, the management of these services is not trivial, because Internet/Intranet application services have diverse characteristics and they are sensitive to

many network infrastructures.

Web-based network management system is based on Web technologies, and is being developed by WBEM (Web Based Enterprise Management) [5] and JMAPI (Java Management API) [6]. These Web-based network management [7] approaches compared with traditional network management [8, 9] have many advantages such as platform independence, mobility and friendly user interface.

We have applied the WWW technology for the management of Internet/Intranet application services. In particular, we have applied the CORBA technology [10] to provide scalable, distributive, and extensible management of legacy system as well as new Internet/Intranet application services.

This paper is organized as follows. In Section 2, we describe the related work. In Section 3, we present the architecture of a Web-based Internet/ Intranet application service management system using CORBA and Java technology. In Section 4, we describe a prototype implementation. Finally, we summarize our paper in Section 6.

2. Related Work

In this section, we describe the previous research related to Web-based management, there are two main approaches; Web-based Enterprise Management (WBEM) and Java Management API (JMAPI) [11]. And we summarize MIBs related to Internet/Intranet application service management.

2.1 Web Based Enterprise Management (WBEM)

Web-based Enterprise Management (WBEM) proposal has been developed and announced by Web-based Enterprise Management consortium, a standards effort chartered by BMC Software, Cisco, Compaq, Intel, and Microsoft. The goal of WBEM is to develop industry standards that will allow administrators to use any Web browser to manage disparate networks, systems, and applications. WBEM describes an architecture, a protocol, a management schema, and an object manager. Though WBEM is designed to address the failings of current management applications, it embraces existing management standards and protocols, allowing the integration of the distributed management services provided by different management platforms. The proposed standards integrate network and system management through the use of the Web technology without affecting the existing network infrastructure.

The WBEM has been designed to:

- provide an architecture that allows management solutions to be constructed covering the traditional areas of configuration management, fault management, accounting, performance, security, operations management, and planning.
- build upon current and emerging Internet standards in the areas of transport, security, and configuration protocols.
- provide a data model that allows uniform modeling and management of system, network, and application elements.
- address the needs of a large distributed set of management elements by providing a scaleable solution.

The proposal presents management information in HTML and other Internet data formats and use the HyperText Transfer Protocol (HTTP). The main components of WBEM are as follows.

- HyperMedia Management Schema (HMMS) - the definition of an extensible, implementation independent common data description/schema, allowing data from a variety of sources to be described, instantiated and accessed regardless of the original source of the data.
- HyperMedia Management Protocol (HMMP) - the definition of a standard protocol over which management data may be published and accessed, allowing management solutions to be platform independent and physically distributed across the enterprise.
- HyperMedia Object Manager (HMOM) - a generic definition for management applications that aggregates management data and uses one or more protocols to present a uniform representation to the browser using HTML.

2.2 Java Management API (JMAPI)

Java computing environment provides the basic capabilities to deliver solutions across a number of operating systems and network protocols. To fully exploit the power of the Java computing environment for solving management problems Sun has developed extensions to the Java base classes that specifically address management problems. These classes are called Java Management API (JMAPI) [12]. JMAPI is a rich set of extensible objects and methods for the development of seamless system, network, and service management solutions for heterogeneous networks. This core set of application programming interfaces can be used across a diverse array of computing environments involving numerous operating systems, architectures, and network protocols, enabling the development of low maintenance, heterogeneous software from a single source.

JMAPI is focused on solving distributed system management problems. Thus, it is crucial that the architecture scales to a number of different environments. It is accomplished in two ways. First, the component that allows a machine to be managed is small. Agent objects for management operations are securely downloaded and executed. This minimizes the distribution and versioning problem for management operations and easily allows for modification and extension of the management operations. Second, the Java Virtual Machine resides on the key platforms to be managed.

The architecture consists of *Browser User Interface*, *Admin Runtime Module*, and *Appliances*. The Browser User Interface is the mechanism from which an administrator issues management operations. These operations may be invoked either interactively through a Web browser or standalone application. The Admin Runtime Module is the mechanism that provides active instantiated management objects to applications. It includes the Agent Object Interfaces, Notification Interfaces, and Managed Data Interfaces. Within the JDBC compliant Managed Data Interfaces, security and data access provisions exist to ensure data security. Appliances are simply the networked devices to be managed. The strategy is to push management close to managed devices with dynamic downloading of agents. JMAPI provides two new approaches, direct download of Java code and Java Remote Method Invocation (RMI) based managed object framework, and Java interfaces for SNMP protocol stack.

2.3 Web-based Network Management System

Web-based network management system is based on the previously developed Web technologies, and is being developed by WBEM (Web Based Enterprise Management) and JMAPI (Java Management API). There are many technological differences between Web-based network management and legacy TMN or SNMP based network management. Figure 1 shows advantages that can be achieved by network management using Web technology.

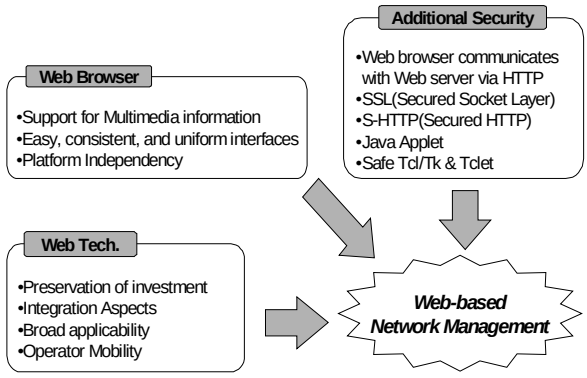

Figure 1. Advantage of Web-based Network Management System

2.4 Internet/Intranet Application Service MIB

Web servers are located at various sites all over the world, providing hypermedia information to the users. The performance of Web server is degraded as the number of users accessing the server increases. The management of Web server has done by using proprietary techniques such as storing management information into a log file. This technique becomes a difficult task when a single organization is responsible for the management of a number of servers. Since many organizations currently use the Internet-standard

SNMP to manage their network devices, it is desirable to manage these Web servers by SNMP. Defining a standard MIB for these servers allows a single management application to manage a number of servers from a variety of vendors.

There are several MIBs related to the management of Internet/Intranet application services, proposed by IETF. These include Host Resources MIB [13], Network Services Monitoring MIB [14], Application MIB (applmib), HTTP MIB and WWW Service MIB [15], each of MIB is described in more detail below.

- **Host Resources MIB**

The Host Resources MIB defines a uniform set of objects useful for the management of host computers independent of the operating system, network services, or any software application. These attributes allow monitoring of the health of the machine and perform capacity-planning calculations. For Web servers, this information is useful in monitoring disk, CPU and other associated hardware.

- **MADMAN MIB**

The Mail And Directory MANagement (MADMAN) MIBs defines a framework for the management of networked applications. This definition is from a service perspective and is independent of the implementation of the application in terms of processes or installed files. Three MIBs have been defined. The first is the Network Services Monitoring (NSM) MIB that defines the generic attributes for management of network applications. The remaining two MIBs are extensions of the first for the management of Message Transfer Agents (MTA) and Directory Service Agents (DSA). These MIBs are not relevant to management of Web servers, but rather demonstrate types of applications that have been instrumented by extending the NSM MIB.

- **Application MIB**

The Application MIB working group is chartered to define a set of attributes for management of Generic applications. This is being done in two phases. The first phase is the system Application MIB (sysApplMIB), intended to require no application instrumentation. Attributes have been defined to represent installed, running, and previously run applications and their components. This generic framework will then be extended with additional attributes that will likely require application instrumentation to support.

- **WWW Service MIB**

The WWW Service MIB defines an experimental portion of the Management Information Base for use with network management protocols in the Internet Community. In particular it describes a set of objects for managing World Wide Web (WWW) services. This MIB extends the application management framework defined by the System Application Management MIB (SYSAPPL MIB) and the Application Management MIB (APPLICATION MIB). The protocol statistics defined in the WWW service MIB are based on an abstract document transfer protocol (DTP). This memo also defines a mapping of the abstract DTP to HTTP and FTP. Additional mappings may be defined in the future in order to use this MIB with other document transfer protocols. It is anticipated that such future mappings will be defined in separate RFCs.

Table 1 summarizes various MIBs for Internet service management. We implemented WWW service MIB of these MIBs in our Web-based Internet/Intranet application service management system.

Table 1. Summarization of Internet Service Management MIBs

	Standard	Definition
Host Resources MIB	RFC 1514 Sept. 1993	Objects for the management of host computers independent of the OS or network services
Network Services Monitoring MIB	RFC 1565 Jan. 1994	Attributes common to the monitoring of any network service application
Mail Monitoring MIB	RFC 1566 Jan. 1994	For monitoring Message Transfer Agents which are responsible delivering email messages
Application MIB	Internet Draft Oct. 1996	Managed objects for fault, configuration and performance management of distributed

		applications from a system perspective
WWW Service MIB	Internet Draft Mar. 1998	Objects for managing World Wide Web (WWW) services

3. Design of Web-based Internet/ Intranet Application Service Management System

3.1 Web-based Management System Architecture for Internet/Intranet Application Service

The Web-based Internet/Intranet application service management system consists of a Web-based managing system and a managed system. The Web-based managing system is composed of Gateway, Web server, manager, management application service and Trap handler. Manager communicates with agent that collects or changes information and remotely controls the resources. Gateway plays the role of message conversion between managers and Web browser. Web server is needed for bringing in Java applet, which will be operated on the Web browser. Java applet allows the network manager to operate Internet/Intranet management services through a Web browser. Java applet communicates with Gateway through CORBA ORB. As CORBA ORB handles all requests from Web browser to Gateway, it provides transparency of information exchange and makes an easy integration of new Internet/Intranet service management applications with the existing management applications, providing effective extensibility of management applications.

The details of gateway and Web server are as follows.

- Web Server: Web server supports the initial connection for the service that Web browser wants. When the initial connection is established, Web server provides manager with Java applet program and HTML text that support network management application service.
- Message Handler: This module deals with request and response messages between Java applet and managers.
- Service Handler: This module offers management application service related to the result of message handler.

Figure 2 shows the proposed architecture of Web-based management system. The following section explains the detailed components.

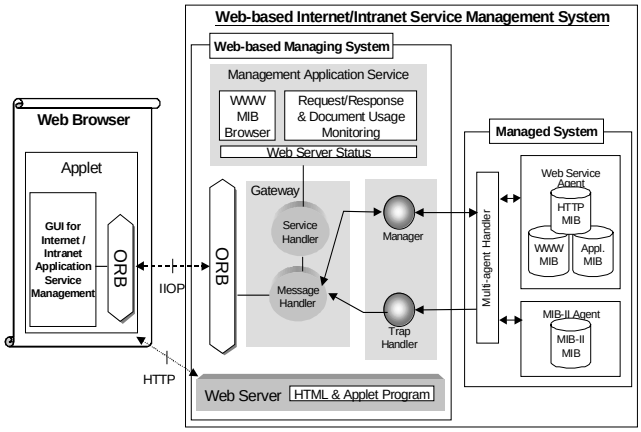

Figure 2. Architecture of Management System

3.2 Design of Web-based Managing System

3.2.1 Design of Manager System

Figure 3 presents the architecture of manager system. Manager systems are composed of manager and Trap Handler. Manager supports management function and collects the managed information from agent. Trap Handler forwards the event report from agent to Message Handler in Gateway.

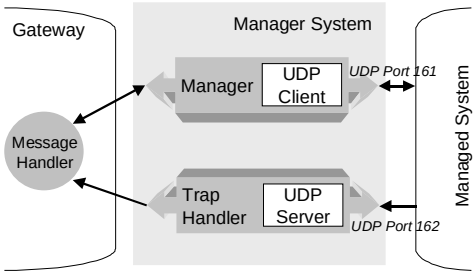

Figure 3. Architecture of Manager System

3.2.2 Design of Gateway

Figure 4 shows the architecture of Gateway. Gateway is composed of Message Hander and Service Handler. Gateway communicates with Java applet in Web Browser through CORBA ORB. Message Handler coordinates the requests from Java applet to various Managers and Trap Handler. It parses the request and forwards it to the appropriate destination. Service Handler controls and maintains the Internet/Intranet application service defined by management systems.

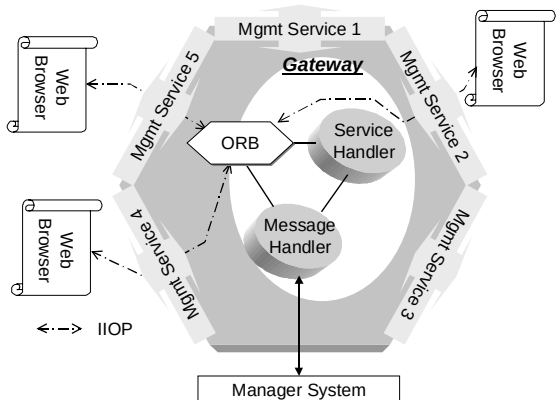

Figure 4. Architecture of Gateway

The procedure for providing Internet application service management is as follows.

① First, Web Browser tries to connect the Web Server in Management System.
② After connection, Web server sends HTML document to Web browser and Java applet is loaded as byte code.
③ Java virtual machine executes loaded applet in Web Browser. Java applet operates management function.
④ Java applet then communicates with the Message Handler in the Gateway.
⑤ Agent performs management using operation primitives, get or set.
⑥ Service Handler communicates with management application service module and provides network management service.

Figure 5 shows the message flows and connection procedures among Web Browser, Managing System and Managed System with regard to Get, Set and Trap operations.

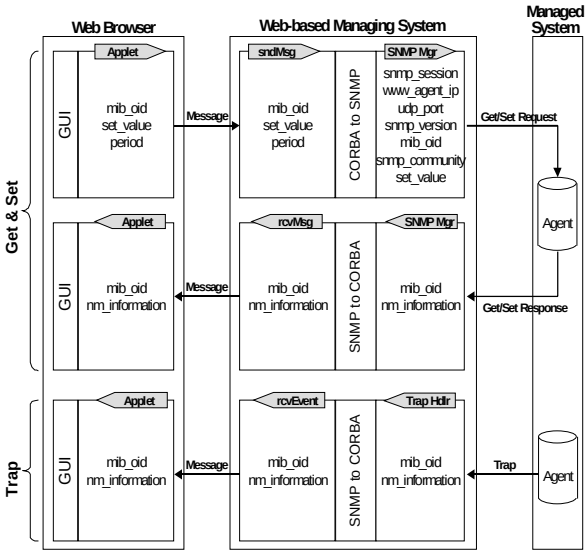

Figure 5. Management Message Conversion

Management messages are shown below.

◆ The management information in Applet

```
## MIB OID
mib_oid = wwwServiceOperStatus
## set wwwServiceOperStatus to running
set_value = running
## The period of Monitoring service
period = NULL
## After Set operation, the value is
## changed to running
nm_information = running
```

◆ The management information in Web-based managing system

```
## SNMP Session is SNMP interface install
snmp_session = 1
## default UDP Port number
udp_port = 161
## SNMP version, community
snmp_version = SNMPv2
snmp_community = public
## IP address of the WWW agent
www_agent_ip = 155.230.12.51
## MIB OID
mib_oid = wwwServiceTable
## set wwwServiceOperStatus to running
```

```
set_value = running
## After Set request, the value is
## changed to running
nm_information = running
```

This is an example of collecting and controlling the information of operational status of a WWW Service (wwwServiceOperStatus). The message of 'running' indicates that the service is operational and available. Manager may have down, halted, congested or restarting as value of wwwServiceOperStatus.

3.2.3 Design of Internet/Intranet Application Service Management Functions

Internet/Intranet Application Service management system provides manager with Internet/Intranet service management based on collected result of management information. Figure 6 shows the structure that Web browser gets the management service through network management service interface.

The management functions are as follows

- MIB Browser: User accesses MIB by using browser interface. MIB browser shows the management information about Internet/ Intranet service management.
- MIB Tree: MIB tree shows the structure of managed object in a tree format.
- Document Usage Monitoring: This module shows the WWW document statistics with graph.
- Request and Response Monitoring: This module shows the request and response of Internet/Intranet services

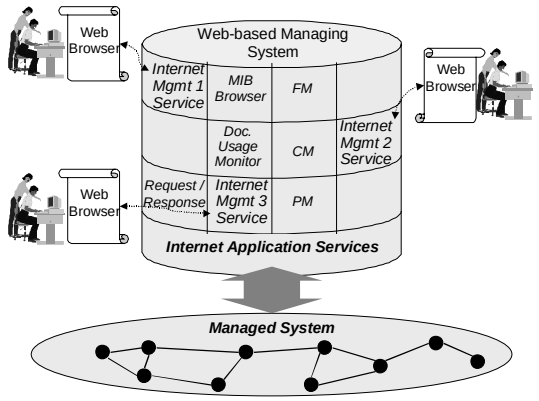

Figure 6. Web-based Internet/Intranet Application Service

3.3 Design of Managed System

Figure 7 shows the architecture of managed system proposed in this paper. This managed system is composed of Multi-agent Handler and two agents, which are Web Service Agent and MIB-II Agent. Each agent has management information about related real resource. We use UNIX system call and log file to build management information.

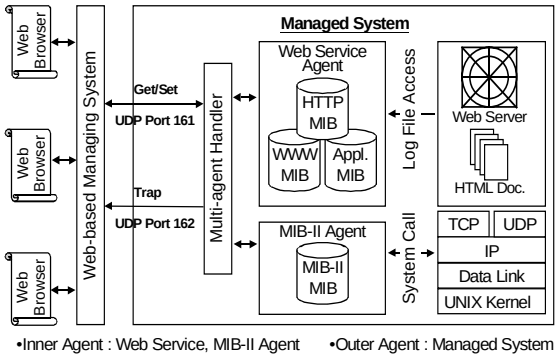

Figure 7. Architecture of Managed System

Managers must have uniform management interface when they access agents. Multi-agent Handler has interface information of agents, such as UDP port number, OID, community and so on. Also, Multi-agent Handler forwards Trap message received from agent to Trap Handler in Web-based Managing System. Diverse agents can be integrated into the Managed System through Multi-

agent Handler.

4. Prototype Implementation

We implemented Web-based managing system using JDK1.1 [16] and Orbix 2.3c [17], and agent is implemented by using Tcl/Tk8.0 and Scotty package 2.1.5 [18].

Figure 8 shows an example of the implemented system using Web browser.

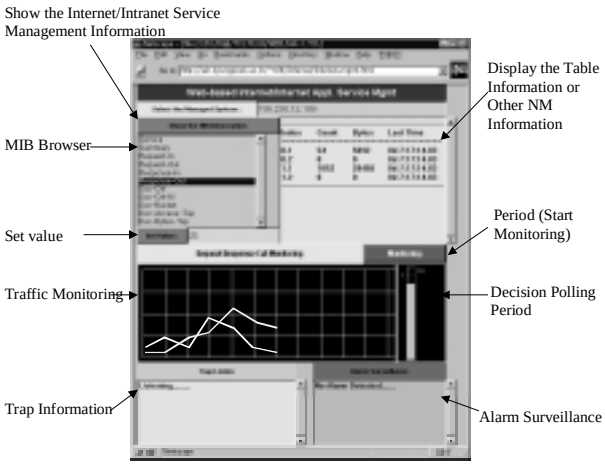

Figure 8. Implementation of Management System for Web Server

It consists of five parts that are explained below.

- Showing the WWW management Information: After selecting an item in the MIB browser, text box shows the NM information to the operator.
- MIB Browser: MIB Browser shows the information of WWW MIB.
- Set value: The value of resources can be changed using this button.
- Request and Response Monitoring: This module shows the request and response of Internet services
- Trap: This module shows events that are notified by trap handler.

Figure 9 shows MIB Tree and MIB Browser.

Figure 9. Implementation of MIB Tree and MIB Browser

MIB Tree shows the structure of WWW MIB. Oid is 'wwwServiceOperStatus', Host is 'ain.kyungpook.ac.kr', Version is 'SNMPv2c' and Community is 'public'. Click the Description button, then MIB Browser shows the information of wwwServiceOperStatus.

6. Conclusion

In this paper, we proposed the architecture of Web-based Internet/Intranet application service management system. We described a prototype implementation. As the proposed Web-based management architecture uses Web technology, it provides mobility, independent platform and useful user interface. Using CORBA technology, the architecture can offer security, portability and scalability. It can also support multiple network management protocols and makes integration of new Internet/Intranet service management applications with the existing management applications.

Reference

[1] T. Berners-Lee, R. Cailliau, J. Groff and B. Pollermann, "World - Wide Web: The Information Universe", Electronic Networking, Vol. 1, No. 2, Spring 1992.

[2] Kumar, Vinay, *Mbone: Interactive Multimedia On The Internet*, Macmillan, November 1995.
[3] Jeff Pulver, *The Internet Telephone Toolkit*, John Wiley & Sons, September 1996.
[4] Phyllis K. Sokol, *From EDI to Electronic Commerce*, McGraw-Hill, 1995.
[5] WBEM Consortium, WBEM homepage. *http://wbem.freerange.com/, July 1996*.
[6] Sun Microsystems, "Java Remote Method Invocation Specification," Prebeta Draft, Revision 1.1, November 1996.
[7] Jong-Tae Park, James Won-Ki Hong, Kyung-Chan Sohn, and Jong-Wook Baek, "Web-based Customer Network Management," Proc. 1997 1st IEEE Enterprise Networking MINI-Conference, Montreal, Quebec, Canada, June 1997.
[8] J. Case, M. Fedor, M. Schoffstall, and C. Davin, "The Simple Network Management Protocol (SNMP)," RFC 1157, May 1990.
[9] William Stallings, *SNMP, SNMPv2, and RMON*, Practical Network Management, 2nd Edition, Addison-Wesley, 1996.
[10] Object Management Group, "The Common Object Request Broker: Architecture and Specification," Revision 2.0, July 1995.
[11] J. W. Hong, J. Y. Kong, T.H. Yun, J. S. Kim, J. T. Park and J. W. Baek , "Web-based Intranet Services and Network Management," IEEE Communication Magazine, Oct. 1997.
[12] Sun Microsystems Inc., "Java Management API Architecture", June 1996.
[13] C. Krupczak and S. Waldbusser, "Applicability of Host Resources MIB to Application Management," Group 3 Report, Empire Technologies, Inc., International Network Services, October 1995.
[14] S. Kille and N. Freed, "Network Services Monitoring MIB," RFC 1565, ISODE Consortium, Innosoft, January 1994.
[15] H. Hazewinkel, C. Kalbbfleisch, and J. Schowoder, "Definitons of Managed Objects for WWW Servers," Internet Draft, March 1, 1998.
[16] Java homepage. *http://java.sun.com*
[17] IONA Technologies Homepage. *http://www.iona.com*
[18] Scotty Homepage. *http://wwwsnmp.cs.utwente.nl/~schoenw/scotty*
[19] Brent Welch, "Practical Programming in Tcl and Tk," draft, January 13, 1995. *http://www.sunlabs.com/~bwelch/book/index.html*.
[20] K. McCloghrie and M. Rose, "Management Information Base for network management of TCP/IP-based internets: MIB-II," RFC 1213, March 1991.
[21] T. Berners-Lee, R. Fielding and H. Frystyk, "Hypertext Transfer Protocol," RFC 1945, May 1996.
[22] H. Hazewinkel, E. van Hengstum and A. Pras: Result of the CEO Project – WWW management. CTIT Technical Report NO. 96 – 18, University of Twente, 1996.
[23] Joe Diamand, "Web-based System and Network Management," Tutorial 8 of IFIP/IEEE International Symposium on Integrated Network Management, San Diego, CA, USA, May 1997.

Biography

Eun-Ju Ha received the BSc and MSc degrees in Electronic Engineering from Kyungpook National University, Korea., in 1993 and 1995, respectively. She is currently working towards the Ph.D. degree at Kyungpook National University.

Su-Hyung Jo received the BSc degree in Electronic Engineering from Kyungpook National University, Korea., in 1997. She is currently working towards the M.S. degree at Kyungpook National University.

Jong-Tae Park received the BSc degree from Kyungpook National University, Korea and the MSc degree from Seoul National University, Korea, respectively. He received the Ph.D. degree in Computer Science and Engineering from the University of Michigan, in 1987. From 1987 to 1988, he was at AT&T Bell Labs, working on network management and service provisioning. Since 1989, he has been working at the School of Electronic and Electrical Engineering at Kyungpook National University, Korea, where he is now an associate professor. He is a chairman of the Committee of Network Operations and Management of Korean Information Communication Society. His research interests include telecommunication management network, distributed database systems, network and resource management in ATM networks, multimedia

communication, and personal communication services.

James Won-Ki Hong is an assistant professor in the Dept. of Computer Science and Engineering, POSTECH, Pohang, Korea. He has been with POSTECH since May 1995. Prior to joining POSTECH, he was a research professor in the Dept. of Computer Science, University of Western Ontario, London, Canada. Dr. Hong received the BSc and MSc degrees from the University of Western Ontario in 1983 and 1985, respectively, and the Ph.D. degree from the University of Waterloo, Waterloo, Canada in 1991. His research interests include network and systems management, distributed computing and multimedia systems. He is a member of IEEE and ACM.