Journal of Network and Systems Management, Vol. 9, No. 1, 2001

Enterprise Network Traffic Monitoring, Analysis, and Reporting Using Web Technology

James W. Hong,[1] Sung-Uk Park,[1] Young-Min Kang,[1] and Jong-Tae Park[2]

Today's enterprise networks are composed of multiple types of interconnected networks. Furthermore, organizations use a variety of systems and applications on these networks. Operations and management staff must provide an efficient, reliable and secure operating environment to support an organization's daily activities. Enterprise networks must be monitored for performance, configuration, security, accounting and fault management. Current management practices typically involve the use of complex, hard-to-learn and hard-to-use tools. What is needed desperately is a set of simple, uniform, ubiquitous tools for managing networks. Web-based management promises to provide such solutions. This paper focuses on the use of Web technology and the Multi-Router Traffic Grapher (MRTG) for the purposes of enterprise network traffic monitoring and reporting. In this paper, we first examine the requirements for enterprise network traffic monitoring, analysis and reporting, and then present the design and implementation of a Web-based network traffic monitoring and reporting system that satisfies those requirements. We also present guidelines we have formulated and used for analyzing enterprise network traffic. We then discuss our experiences in using such a system for traffic monitoring on two large enterprise networks.

KEY WORDS: Network monitoring; traffic analysis; performance management; Web-based management; network planning

1. INTRODUCTION

Enterprise networks are corporate computer networks composed of network devices, systems, and services supporting various corporate applications. Enterprise networks are growing rapidly in size and complexity because more systems and sophisticated applications run on them. The most challenging tasks facing network and system administrative staff today is to provide an efficient, reliable and secure computing environment.

[1] Dept. of Computer Science and Engineering, POSTECH. E-mail: *jwkhong@postech.ac.kr*
[2] School of Electronics and Electrical Engineering, Kyungpook National University. E-mail: *park@ee.knu.ac.kr*

1064-7570/01/0300-0089$19.50/0

Administrators today typically use a management tool (e.g., HP OpenView [1]: IBM/Tivoli NetView [2]: SunNet Manager [3]: Cabletron Spectrum [4], etc.) to monitor and control their networks. However, most of these tools are very costly and difficult to learn and use. Typically, it takes months to learn their features and capabilities. Even then, most administrators use only a fraction of the features provided, performing only basic tasks (such as traffic monitoring and fault monitoring). Further, many network administrators frequently change jobs. Corporations are thus faced with the high cost of retraining new staff how to use their particular systems. Consequently, a management tool that is easy to learn and operate in a short period of time is desperately needed.

The World-Wide Web (WWW or Web) [5] has been revolutionizing the use of the Internet over the past few years with its simple but powerful ways of retrieving various types of data (including text, graphic, image, voice, and video) from worldwide sources. Java [6] has also been changing the way people develop and run network applications on the Internet. Software developed using the Java language is portable across various platforms, and can be distributed and accessed through Web browsers. Applying powerful new Web technologies to the management ("Web-based management") of enterprise networks can provide solutions to the current problems of complexity and decrease maintenance and training costs. At the same time, these new technologies provide solutions to the problems of efficiency, reliability, and security. Web-based management provides a globally uniform user interface (namely the Web browser), which is a well-established user interface for information retrieval, as well as for running various Internet/Intranet applications. Most of the technologies now used by Web-based management have been proven on the Internet and are familiar to ordinary users. Further, Web-based management can easily accommodate most existing standard management frameworks (such as SNMP [7]: CMIP [8]: and DMI [9]) as well as proprietary frameworks.

Our earlier work on applying Web technology to managing ATM customer networks [10]: distributed multimedia systems and services [11]: and Intranet services and network management [12] have already validated simple but powerful features of Web-based management. Others have demonstrated the usefulness and appropriateness of applying Web technology to the management of other network and system resources [13–16]. For the last several years, two industrial groups have been working on standardizing Web-based network management. The outcome of these two working groups are Web-Based Enterprise Management (WBEM) [17–20] and Java Management eXtensions (JMX) [21, 22]. WBEM from Distributed Management Task Force (DMTF) is an initiative based on a set of management and Internet standard technologies developed to unify the management of enterprise computing environments. JMX is based on the Java technology, and so draws on Sun's experience with Java management.

This paper describes how Web-based management can resolve the short-

comings of current management approaches, particularly for traffic monitoring, analysis and reporting of enterprise networks. Section 2 examines the requirements for providing an inexpensive, easy-to-learn and easy-to-use but powerful, unified management approach for network traffic monitoring and reporting. Based on these requirements, Section 3 presents our Web-based enterprise network monitoring and reporting system. Section 4 describes network traffic analysis guidelines in an enterprise network management environment. Section 5 describes an implementation of such a system using widely available technologies with added useful features. Section 6 describes our experiences and results obtained by using the developed system for monitoring the network traffic of two large enterprises. Section 7 summarizes our work on Web-based enterprise network monitoring, analysis and reporting, and discusses potential future work.

2. REQUIREMENTS

In this section we examine some of the most important requirements needed for a Web-based enterprise network monitoring and reporting system.

Enterprise networks typically consist of network devices such as routers, bridges, switches, hubs and so on. These devices are interconnected to form the various topological configurations of enterprise networks. A management system must be able to automatically detect all the network devices in an enterprise network, and discover the topology of the backbone, as well as its subnetworks. Although the topologies of enterprise networks do not change frequently, they do change occasionally. When the topology changes, the management system must detect the change and modify the necessary information for proper monitoring and control.

Most modern network devices are equipped with management agents, typically an SNMP agent [7] for computer network devices and a CMIP agent [8] for telecommunication network devices. These management agents monitor their devices and maintain the corresponding management information base (MIB). These MIBs contain information regarding the amount of traffic the devices generate for the network as well as the amount of traffic they receive from the network. Such traffic data can be collected by the data collection component of the manager system by requesting the appropriate part of a MIB from the agents using a corresponding management communication protocol.

The traffic data collected from the agents by the manager must be stored somewhere in the system. This requires the management system to possess a management information repository. The logged, raw traffic can then be retrieved periodically and used in traffic data analysis, which may involve generating maximum input amount, maximum output amount and average input and output amount over a period of time. These analyzed data can generate graphs or histograms about traffic patterns, or can be used for calculating the utilization

of each subnetwork or the entire enterprise network. The data can also be used for calculating network usage for account management purposes. The analyzed data can be logged back into the management information repository for further analysis later. In order to perform these possible analysis functions, appropriate tools are needed in the management system.

Another requirement for the management system is to generate various reports. There are two aspects to this. One is to report urgent problems that may be detected by the monitoring system. This may include a surge of traffic in one or more parts of the enterprise, which may be reported to the administrative staff who can promptly analyze the problem and take appropriate actions. The other aspect is for the management system to generate periodic reports for the administrative staff to use for generating weekly, monthly and annual reports. It would be ideal if these reports could be generated automatically based on the requirements specified by the administrative staff. These reports can be used in reporting to corporate managers regarding enterprise network usage. They can also be used as supporting data when an enterprise network needs to be upgraded to solve current problems or to prevent disasters in the future.

In order for the network administrative staff to view the current status as well as the past history of the network, the analyzed information must be made easily available. This requires the Web server to access information so it can be made available through the Internet. The information that Web servers provide must be prepared in HTML [23], which may contain graphic images about the traffic patterns as well as numerical data in text. In this way, network administrators can retrieve, view and analyze data from anywhere and at anytime using a Web browser. This is one of the greatest advantages of the Web-based management approach over the traditional platform approach, where administrative staff must use a network management console that is typically located in the computing center of an enterprise.

Various analyzed data about an enterprise network can be important to the enterprise. This data must be protected from unauthorized access. A security mechanism must be placed in the management system to protect such valuable information, preventing illegal users from accessing not only raw data, but also published data, specifically the enterprise's data on the Web.

3. WEB-BASED NETWORK TRAFFIC MONITORING SYSTEM

In this section, we present the design of a Web-based enterprise network traffic monitoring and reporting system. This system has been designed to satisfy all requirements discussed in the previous section. We first present its architecture, and then describe the network traffic related parameters used in our system.

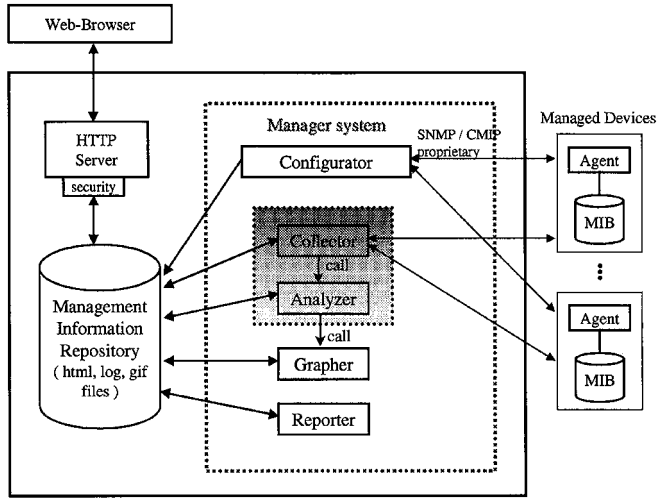

Fig. 1. Design architecture of a Web-based network traffic monitoring system.

3.1. Architecture

Most network management systems use the request-reply interaction paradigm between managers and agents. Our Web-based network management system uses the same paradigm. Figure 1 illustrates the design architecture of a Web-based enterprise network traffic monitoring system.

Onr newly designed monitoring system contains four groups of entities: (1) manager system: (2) managed devices: (3) management information repository: and (4) Web server and browser.

3.1.1. Manager System

The manager system is the core of the entire system, where most of the real management activities take place. It interacts with the agents residing in managed devices using management communication protocols (such as SNMP, CMIP or proprietary) to collect configuration and traffic related data, and store them in the management information repository. The manager system consists of the following functional components:

- *Configurator:* discovers network devices that need to be monitored and creates a database of configuration information on the enterprise network. This configuration information contains not only the devices that must be monitored, but also all the network interfaces in each device.
- *Collector:* uses the configuration information generated by the configurator and collects traffic-related data from the management agents. Polling

frequency can be set and changed for individual devices as well as for all the devices. Typical polling intervals are every few minutes. The collected data is then logged into the management information repository.

- *Analyzer:* periodically retrieves the traffic data logged by the collector and performs various traffic analysis functions that are defined in the system. Simple analysis typically includes calculating maximum, minimum and average utilization of a network segment. Analyzed data may be logged back into the repository for further analysis later or it may be used by the grapher (described later) to generate graphical outputs.
- *Grapher:* generates various graphical outputs (e.g., graphs or histograms in GIF or JPEG format) using the analyzed data. These graphical outputs are inserted into HTML files so they can be retrieved and viewed by Web browsers.
- *Reporter:* typically can perform two types of reporting. One type notifies administrative staff when problems are detected in the network being monitored. The means of notifying the staff may include alarm messages in the browser, electronic mail, electronic paging and so on. The other type generates summary reports in the form of tables.

3.1.2. Managed Devices

Managed devices are the network elements that form an enterprise network. Typical devices are switches, routers, bridges and hubs. Most modern network devices are equipped with management agents (such as an SNMP agent, CMIP agent or proprietary agent) that monitor the devices they reside in, and provide various useful data to managers upon request. They are also capable of reporting when certain predefined problems or conditions in their devices are detected.

3.1.3. Management Information Repository

The configuration, traffic-related data, as well as analyzed data are stored in the management information repository by the manager system. The stored data can be retrieved and processed in a number of ways and then returned to the management information repository. Graphical image files and HTML files generated by the grapher and reporter are also stored here. The Web server, upon request of one or more Web browsers, accesses the repository for appropriate information. Thus, the management information repository acts as a common repository for the manager system and Web server.

3.1.4. Web Server and Browser

In a Web-based management system, the Web server acts as the provider of management information generated by the manager system. Any Web server that supports HTTP 1.0 [24] or higher version can perform such services. A security mechanism is included to prevent illegal access of valuable management information that only the administrative staff is allowed to retrieve. The Web

Table I. Indicators Used in Measuring Network Performance

Class	Specific performance indicators	Network devices
Performance indicators	Interface octets in and out Interface unicast and non-unicast frames in and out CPU utilization Forwarding rates	Hubs, switches, bridges, routers, servers
Performance degradation indicators	Transmission collisions Deferred transmissions TCP retransmissions	Hubs, switches, bridges, routers, servers
Connectivity and data transmission problem indicators	Interface CRC errors in and out Interface lost carriers Interface disconnects Excess retries	Bridges, routers servers

browser can be any browser that is widely available on PC and workstation platforms today.

3.2. Network Performance Parameters

Monitoring the performance of networks entails collecting network-performance-specific information from the MIBs provided by the management agents. Such information can be categorized into three classes: performance indicators, performance degradation indicators, and connectivity and data transmission problem indicators [25]. Specific performance indicators for each class, their descriptions and relevant network devices are given in Table I.

As mentioned earlier, the network performance indicators listed in Table I can be obtained from most management agents instrumented in hubs, switches, bridges, routers and so on. These indicators are defined as MIB variables or objects and are constantly monitored by the agents in network devices. Table II lists relevant performance variables defined in SNMP MIB [26].

4. NETWORK TRAFFIC ANALYSIS GUIDELINES

One of the most important goals of network traffic monitoring is to provide an efficient and reliable network environment to users. To achieve this goal, the network must be constantly monitored, its traffic must be analyzed and results should be automatically reported to network administrators. Network traffic should be analyzed according to specified guidelines. Guidelines may vary from one network to another. In this section we present the analysis guidelines we have formulated and used in our monitoring and reporting system. The analysis is based on the network performance related data we obtained from net-

Table II. Network Performance Related SNMP MIB Variables

Formal name	Description
IfInNUcastPkts	The number of non-unicast packets delivered to a higher-layer protocol.
IfInUcastPkts	The number of unicast packets delivered to a higher-layer protocol.
IfInOctets	The total number of octets received at the interface, including framing characters.
IfInDiscards	The number of inbound packets chosen to be discarded even though no errors were detected to prevent their being deliverable to a higher-layer protocol (e.g., to free up some buffer space).
IfInErrors	The number of inbound packets that contain errors.
IfOutErrors	The number of outbound packets that could not be transmitted due to errors.
IfOutQLen	The length of the output packet queue (in packets).
IfForwDatagrams	The number of input datagrams for which this node was not their final IP destination, thus to be forwarded to their final destinations.
IpOutDiscards	The number of output IP datagrams for which no problem was encountered to prevent their transmission to their destination, but which were discarded (e.g., for lack of buffer space).
TcpRetransSegs	The total number of segments retransmitted—that is, the number of TCP segments transmitted containing one or more previously transmitted octets.

work devices that have SNMP [7, 26] agents (i.e., retrieve appropriate SNMP MIB variables). Fortunately, most, if not all, modern data networking devices are equipped with SNMP agents. Network monitoring also involves monitoring the status of network devices that route packets from one network to another within and between enterprise networks.

Figure 2 illustrates a small portion of a typical enterprise network, consisting of an FDDI backbone network, Ethernet subnets and serial links attached to FDDI routers. In such a network, we need to analyze the traffic going into or coming out of network interfaces of network devices (such as routers, bridges, and gateways) as well as to monitor the status of network devices themselves. In analyzing the network traffic, we need to use a different analysis formula for each type of component network. For example, we need a formula for each FDDI, Ethernet and serial links. We have used and reformulated Leinwand's equations [29, 30].

4.1. Traffic Analysis

An FDDI network is composed of a dual counter-rotating ring [28] with two or more routers attached to it. Thus, it has two rings: a primary and a secondary. The primary ring is used for data transmission. The secondary ring is used as a backup, and is activated only when there is a problem in the primary ring. The traffic flows in only one of the two rings at a time.

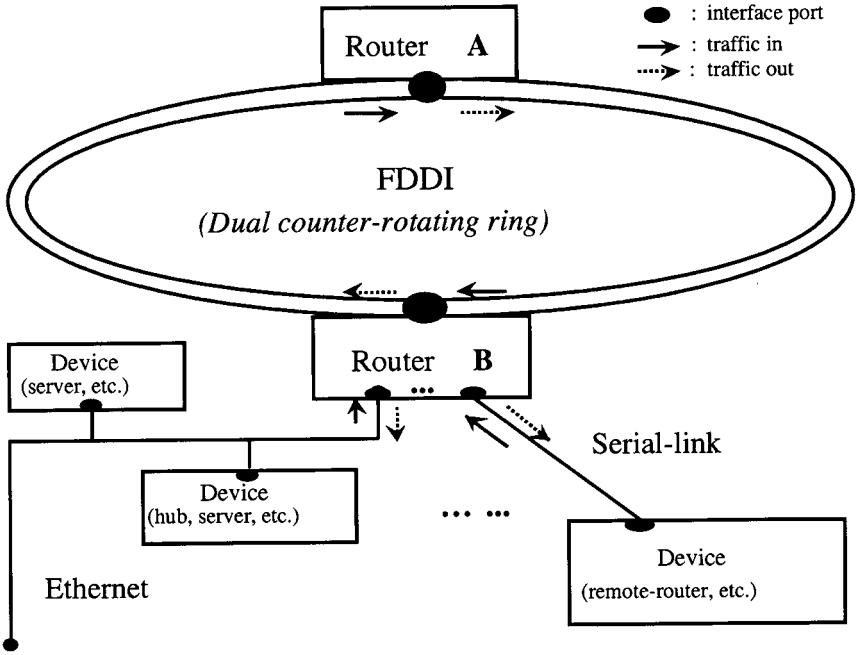

Fig. 2. Typical enterprise backbone network.

When analyzing the utilization of an FDDI network, one must keep track of the input and output traffic of all active FDDI network interfaces attached to FDDI routers. The input traffic represents the number of octets, which originate from the devices and/or networks attached to the router and from other routers in the ring, being transmitted onto the ring. Since FDDI uses the timed-token ring protocol, the output traffic represents the number of octets being received from the ring into a router. In order to determine the utilization of an FDDI network, the input and output traffic of all the routers in the FDDI network must be added, then their sum is divided by two. Since the FDDI network's maximum transmission rate is 100 mega-bits-per-second (Mbps), Formula (1) determines its utilization.

A general rule of thumb is that when the utilization of an FDDI network is over 90%, it is considered to be overloaded. If the utilization goes over 90% occasionally, it is not a serious problem, however, if the average utilization is over 90%, then the FDDI network is highly overloaded and network reconfiguration or upgrade should be considered.

$$\text{FDDI Utilization (\%)}$$

$$\approx \frac{1}{2} \sum_{\text{devices}} [(\text{total bits sent} + \text{total bits received})/\text{bandwidth}]$$

$$= \frac{1}{2} \sum_{\text{devices}} \cdot \left[\frac{[(\text{ifInOctets}_{x+t} - \text{ifInOctets}_x) + (\text{ifOutOctets}_{x+t} - \text{ifOutOctets}_x)] \times 8}{(\text{sysUpTime}_{x+t} - \text{sysUpTime}_x) \times \text{ifSpeed} \times 100} \right] \tag{1}$$

Note that in Formulas (1) to (4), x denotes the previous polling time and t denotes the polling interval in seconds.

The Ethernet uses a broadcasting transmission method (specifically CSMA/CD), where stations attached to it contend for the medium. The maximum transmission rate for an Ethernet is 10 Mbps. When analyzing the utilization of an Ethernet network, one must keep track of the input and output traffic of the Ethernet network interface in a router. The input traffic represents the number of octets which originate from other network interfaces in a router. The output traffic represents the number of octets being generated from the stations attached to the Ethernet network. To determine the utilization of an Ethernet network, the input and output traffic must be added, and then the sum is divided by the maximum transmission speed (i.e., 10 Mbps). The Ethernet traffic analysis formula is given by Formula (2).

A general rule of thumb is that when the utilization of an Ethernet network is over 40%, it is considered to be overloaded. If the utilization goes over 40% occasionally, it is not a serious problem, however, if the average utilization is over 40%, then the Ethernet network is highly overloaded and network reconfiguration or upgrade should be considered.

$$\text{Ethernet Utilization (\%)}$$

$$\approx (\text{total bits sent} + \text{total bits received})/\text{bandwidth}$$

$$= \frac{[(\text{ifInOctets}_{x+1} - \text{ifInOctets}_x) + (\text{ifOutOctets}_{x+t} - \text{ifOutOctets}_x)] \times 8}{(\text{sysUpTime}_{x+t} - \text{sysUpTime}_x) \times \text{ifSpeed} \times 100} \tag{2}$$

In the case of serial link utilization, the mode of transmission (half duplex or full duplex) must be considered. With half-duplex serial links, the sum of

input traffic and output traffic should be used for determining the link utilization. With full-duplex serial links, each link consists of two lines, one for each direction. Although a serial link can be configured with different transmission speeds for each direction, most serial links use the same speed for both directions. For utilization analysis, each directional link must be considered separately for determining the utilization. However, the line that yields higher utilization should be the one that needs to be monitored closely, and any action decision should be based on that line. Formula (3) is used to analyze the utilization of full-duplex serial links for each direction.

A general rule of thumb is that when the utilization of a serial link is over 90%, it is considered to be overloaded. If the utilization goes over 90% occasionally, it is not a serious problem, however, if the average utilization is over 90%, then the link is highly overloaded and network reconfiguration or upgrade should be considered.

$$\begin{aligned}&\text{Serial Link Utilization (\%)}\\ &\approx \text{Max(total bits sent, total bits received)/bandwidth}\\ &= \frac{\text{Max}[(\text{ifInOctets}_{x+t} - \text{ifInOctets}_x), (\text{ifOutOctets}_{x+t} - \text{ifOutOctets}_x)] \times 8}{(\text{sysUpTime}_{x+t} - \text{sysUpTime}_x) \times \text{ifSpeed} \times 100}\end{aligned} \tag{3}$$

4.2. Error Rate Analysis

Another important network performance indicator is the error rate. Ideally, the error rate should be zero. A general rule of thumb is that when the average error rate is greater than 1% of the bandwidth, then the network administrator must closely examine the network interface. The formula for determining the error rate is given by Formula (4).

$$\begin{aligned}&\text{Error Rate (\%)}\\ &= \frac{\text{ifInErrors}_{x+t} - \text{ifInErrors}_x}{\text{totalPktsIn}_{x+t} - \text{totalPktsIn}_x}\\ &\text{where totalPktsIn} = \text{ifInUcastPkts} + \text{ifInBroadcastPkts}\\ &+ \text{ifInMulticastPkts}.\end{aligned} \tag{4}$$

4.3. Network Device CPU Load Analysis

In general, utilization of network devices (e.g., bridges, hubs, routers) can be defined by the CPU load of each device. A high CPU load means that the

device is busy, processing packets going through the device. If the CPU load is over 100%, then the packets could be dropped by the device, thus causing retransmissions. This may cause the network to carry more data than the original data traffic. Accordingly, a device with constant high CPU utilization must be examined carefully and appropriate actions taken. One solution is to upgrade the CPU in the device to overcome the problem. Another measure, which works quite nicely in many cases, is to increase the memory in the device.

5. IMPLEMENTATION

Based on the requirements discussed in Section 2 and the design architecture presented in Section 3, we have implemented a prototype Web-based enterprise network monitoring and reporting system. The system has been built based mostly on freely available tools which we have extended and added useful features to. The most notable freely available tool on the Internet we have used is the Multi-Router Traffic Grapher (MRTG) [27].

5.1. MRTG+

The features we have added to MRTG include a map-sensitive graphical user interface of network maps, active subnet reporting capability, CPU load monitoring capability and a security mechanism. We call the collection of these tools MRTG+. We have used MRTG+ for monitoring two enterprise networks. The experience we have obtained using it over one year will be presented in Section 6.

Figure 3 illustrates the implementation architecture showing the components and operational steps. Users employ Web browsers (such as Netscape or Internet Explorer) to access HTML documents from a Web server for viewing network maps, traffic utilization reports and graphs, which are generated by the manager system and stored in the management information repository. The following is a list of functional modules and their brief descriptions, which compose the monitoring system.

- *cfgmaker:* acts as the 'Configurator' in the design architecture. It discovers network devices in the enterprise network and constructs a network configuration database.
- *mrtgidx:* creates an HTML file with indexed hyperlinks to monitored network interfaces using the network configuration database.
- *checkif:* detects network interfaces in each of the network devices that must be monitored.
- *MRTG+:* acts as the 'Collector & Analyzer' in the design architecture.

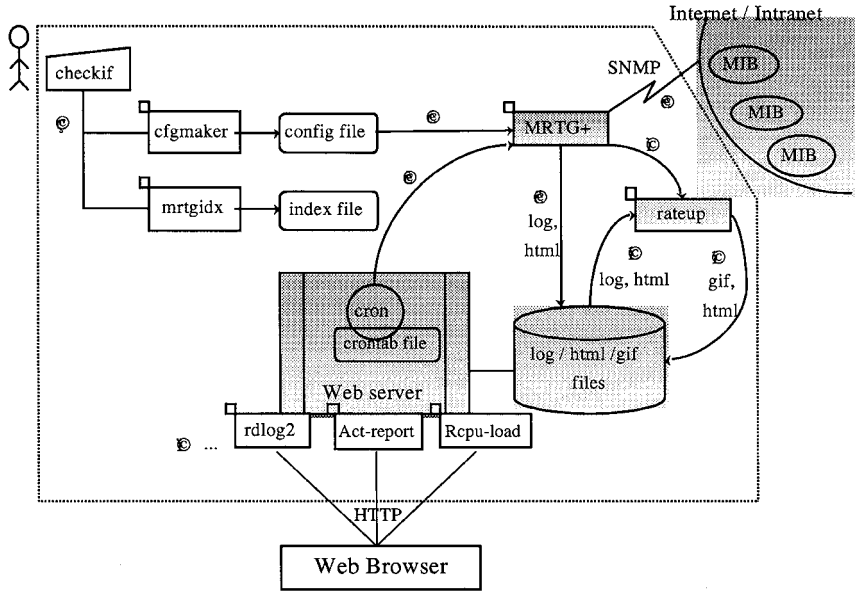

Fig. 3. Implementation architecture of Web-based traffic monitoring and reporting system.

It is the main program that gathers traffic data and stores them into the management information repository.

- *rateup:* acts as the 'Grapher' in the design architecture, retrieving lot files and generating graphs (in GIF format) for the monitored network interfaces.
- *active-report:* acts as the 'Reporter' in the design architecture, retrieving log files and generating reports on interfaces that generated traffic over the specified threshold.
- *rdlog2:* links the network segments in the GUI network map with the monitored data so that link utilization can be viewed directly from GUI maps.
- *rcpu-load:* generates CPU load graphs and threshold reports on network devices.
- *crontab:* is a script file used by the *cron* program for periodic polling of management agents and report generation.

5.2. MRTG+ Operation Steps

Here, we describe how the Web-based monitoring and reporting system works internally. The bracketed numbers in Fig. 3 represent the operational steps.

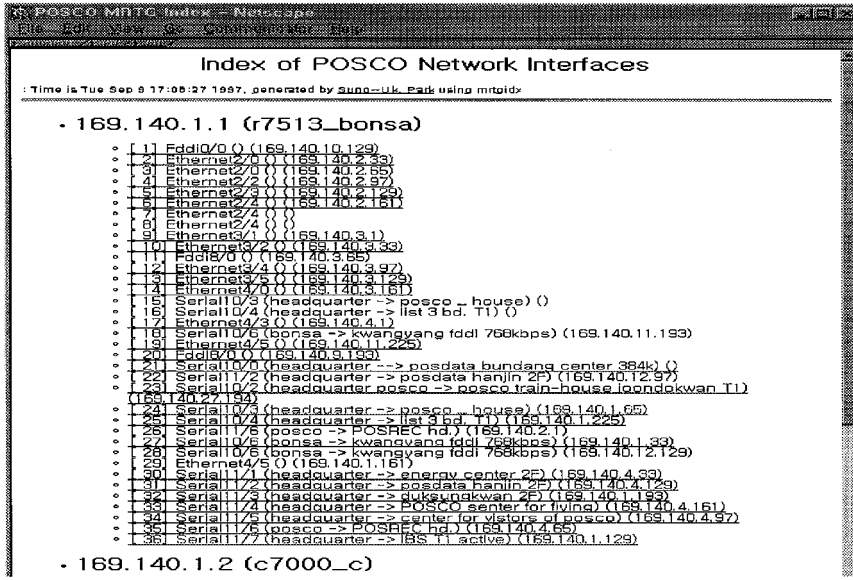

Fig. 4. Generated index of discovered network interfaces.

Step 1. The enterprise network to be monitored is examined and the initial network configuration file is generated. This is done by the *cfgmaker* and *checkif* modules. Then the *mrtgidx* module generates an HTML file containing hyperlinks to network interfaces of network devices to be monitored. Figure 4 shows the generated HTML file displayed in a Web browser. It lists all the network interfaces found in the devices and the active interfaces that have hyperlinks to graphs of their traffic histories. The inactive interfaces do not have hyperlinks since no traffic is generated from them.

Step 2. Based on the triggering functions defined in the system crontab file, various management functions are triggered. A partial content of a sample crontab file used by the Web-based network traffic monitoring system is shown in Fig. 5. For example, MRTG+, the main program in our system, is triggered every 10 minutes and polls the management agents and collects their traffic data. Other programs are triggered periodically to perform traffic analysis, graph generation and report generation. Note that the triggering times have been interlaced in order to evenly distribute the CPU processing and network loads due to the management activities.

Steps 3–5. Using the configuration information generated by *cfgmaker*, MRTG+ generates SNMP requests to the management agents in the enterprise network, receives requested traffic related data, and stores them as log files in

```
0,10,20,30,40,50 * * * */bin/MRTG+ posco.cfg> /dev/null 2>&1
5,25,45 * * * * /bin/mrtg-err posco-err.cfg > /dev/null 2>&1
8,28,48 * * * * /bin/rdlog2 -i posco.fig -o posco.gif
15,35,55 * * * * /bin/active-report > /dev/null 2>&1
58 23 * * * /bin/history-report > /dev/null 2>&1
```

Fig. 5. A sample **crontab** content.

the management information repository. A consolidation algorithm processes the log file so that its size does not increase infinitely.

Steps 6–8. MRTG+ triggers the *rateup* program which retrieves log files, and performs traffic analysis and generates traffic graphs on GIF format. HTML files, used for displaying analyzed traffic information for the monitored network interfaces, are also retrieved from the information repository. Then the numerical data and graphs in the HTML files are updated and returned to the repository. Figure 6 displays an HTML file retrieved from the repository using a Web browser. Although Figure 6 only shows two graphs and their corresponding numerical data for a monitored network interface, four graphs are actually generated in a single Web page: daily, weekly, monthly, and yearly. This permits the administrative staff to observe traffic utilization trends for every network segment by four time durations.

Step 9. *rdlog2* automatically generates a sensitive map for an entire enterprise network, or for portions of it, so that link utilization monitoring, as well as device CPU load monitoring, can be viewed directly by pressing a line or device shown in the GUI map. Figure 7 shows the sensitive network map for the enterprise network we have been monitoring for an enterprise (more on this will be described in Section 6). Links are illustrated using colors. Ten different colors are used to represent link utilization in ten different scales which change automatically whenever the utilization rates change from one scale to another. Thus, the heavily-used links can be easily observed by the administrative staff by looking at the link colors. *Act-report* periodically generates a report for active subnets (whose link utilizations have passed the set threshold) and error-prone subnets (which generate errors over the set threshold) and provides hyperlinks to detailed reports for the detected subnets. Figure 8 displays such a report. *Rcpu-load* generates history graphs for the CPU load of each device, and Fig. 9 displays such graphs for a router.

6. EXPERIENCE

We have deployed our Web-based enterprise network monitoring and reporting system at Pohang Iron and Steel Company (POSCO). We used it over one year to monitor POSCOs enterprise network including all subsidiary com-

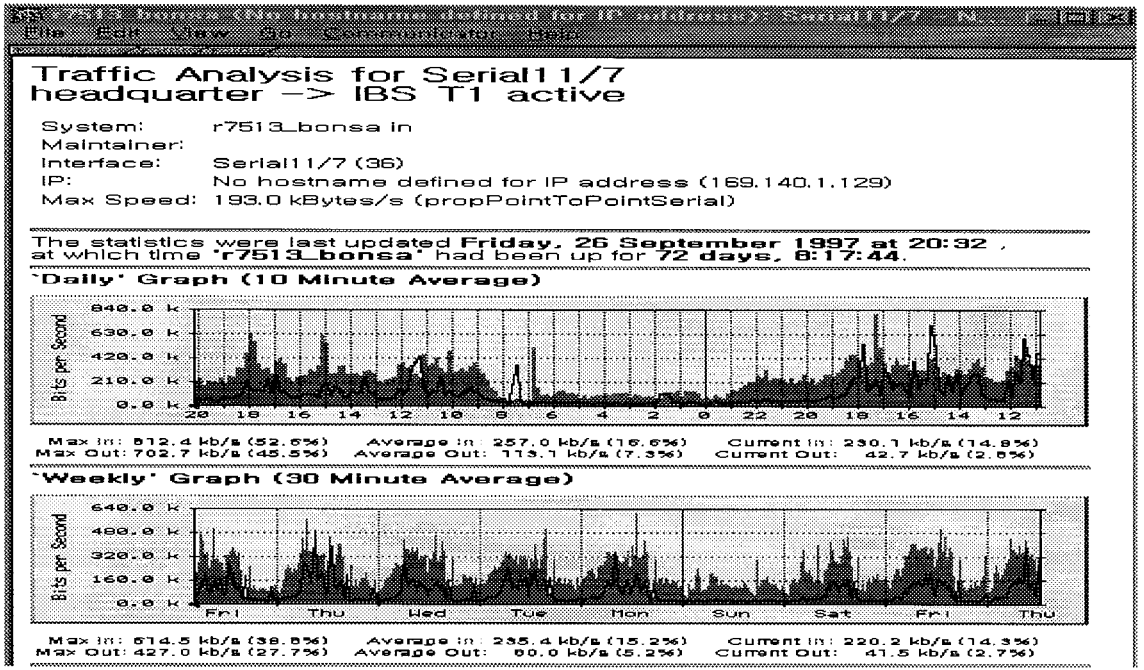

Fig. 6. Traffic monitoring graphs.

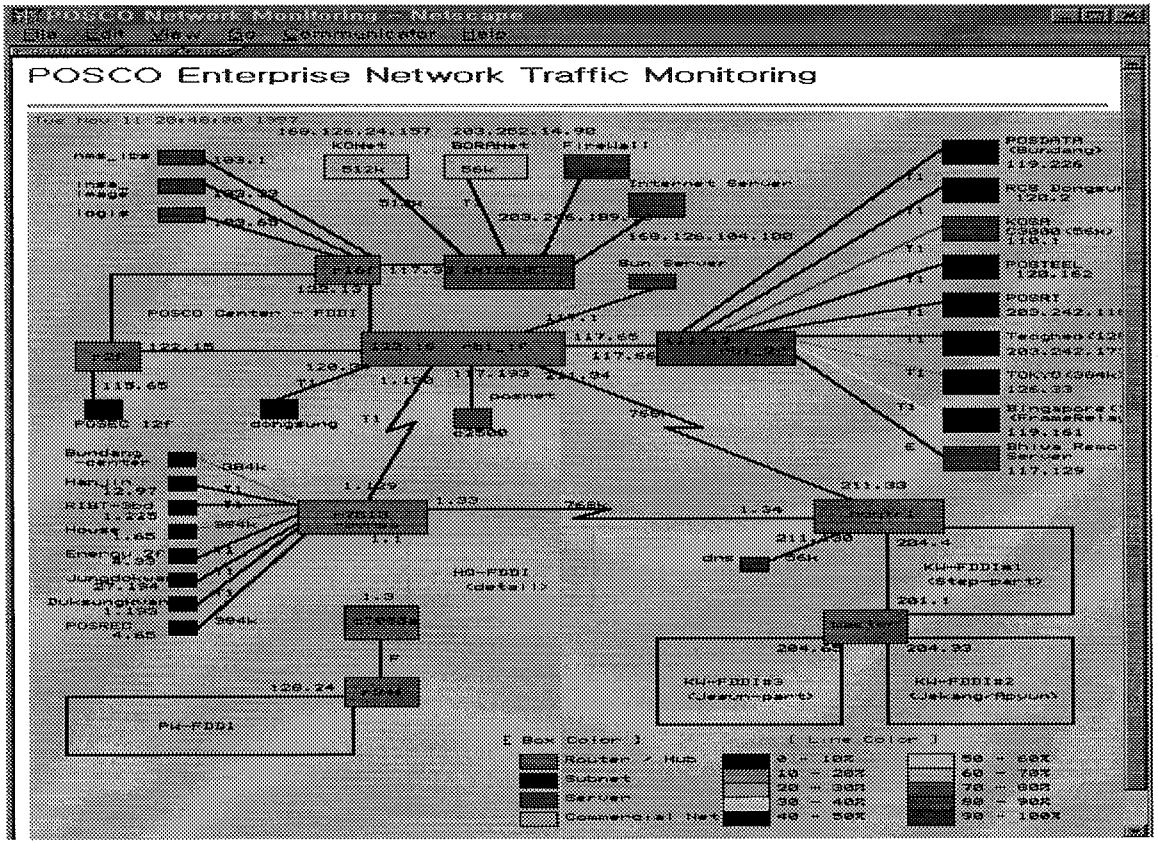

Fig. 7. POSCO enterprise network configuration.

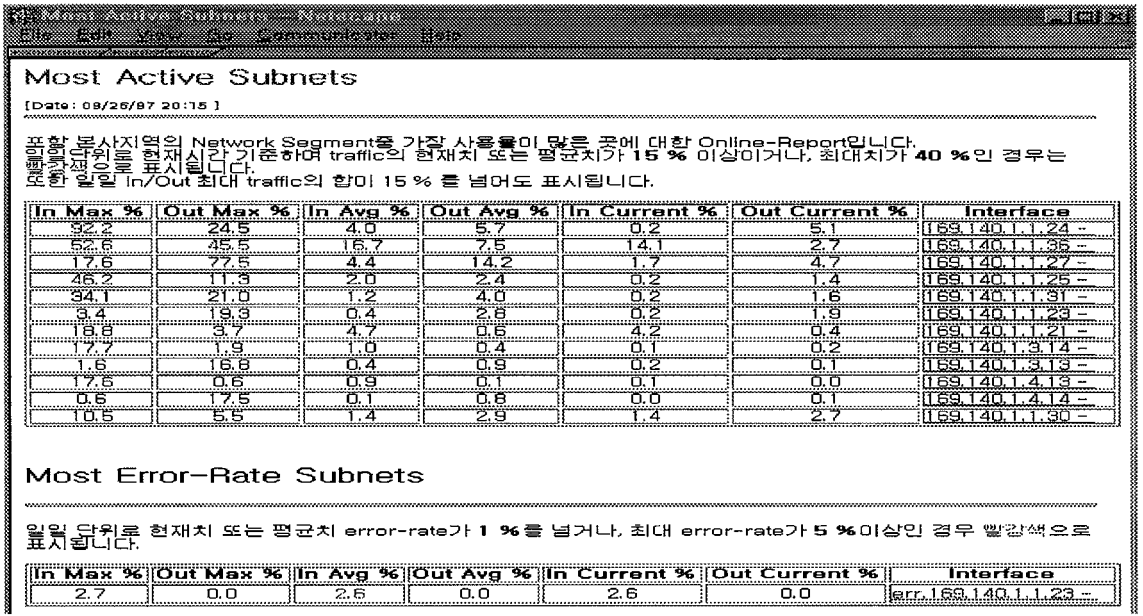

Fig. 8. Threshold reports of active subnets and error-prone subnets.

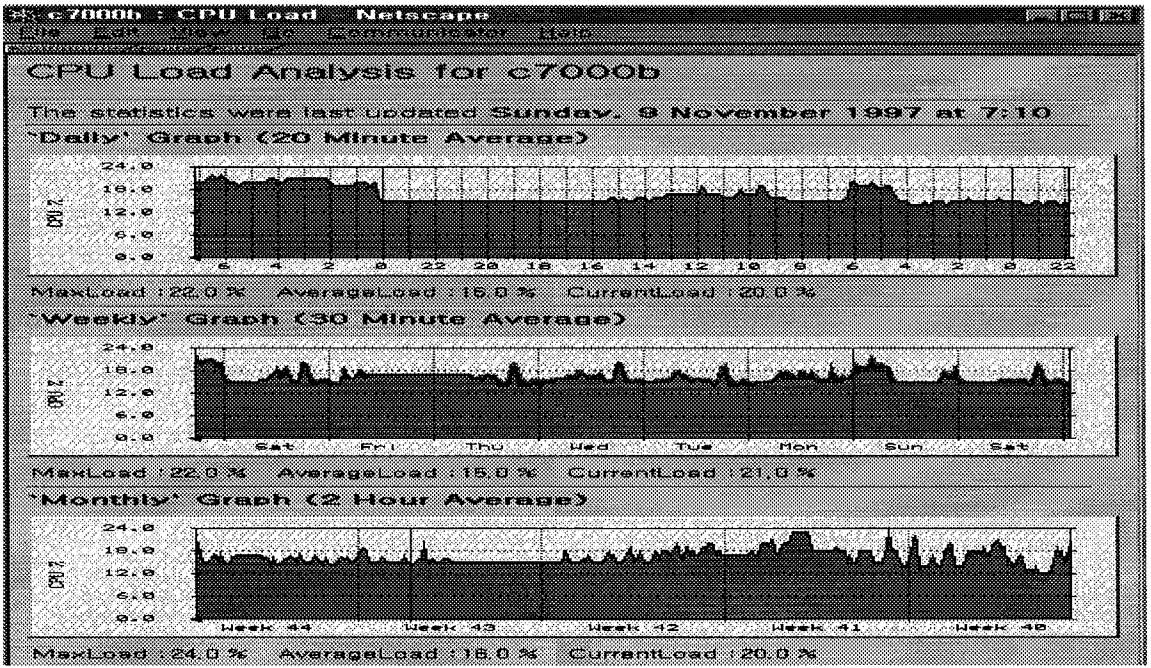

Fig. 9. Router CPU load monitoring.

panies in three different Korean cities and in several foreign countries. Each site has a local area network consisting of a variety of network segments (such as Ethernet, Fast Ethernet, Serial) interconnected by FDDI backbone networks. All sites have been internetworked by wide area networks whose transmission speeds range from 56 kbit/s to T1 links. The POSCO campus network consists of 70 routers and approximately 700 network segments, and the computer systems that are on the network include nine mainframes, hundreds of workgroup servers and approximately 8000 PCs. Figure 7 shows a simplified graphical view of the POSCO campus network.

Before using our Web-baed monitoring and reporting system, POSCO had used some commercial NMS tools such as SunNet Manager and NetView. However, these tools were used for managing only parts of the enterprise network and were not very well utilized by the network administrators. The administrators found these management tools difficult to learn and use because of their many features. Furthermore, they were too complex to fully understand and use. It typically took them several months before they were able to use a tool and even then, only a few critical components were used. Another problem POSCO faced was that these network administrators (after learning the tools) frequently moved to other companies and new administrators had to be trained repeatedly. POSCO was looking for a tool that the staff could learn quickly and easily.

Our monitoring system has been deployed and used for monitoring the entire enterprise network (not just parts of the enterprise network). Shortly after installing and beginning to use our system, the administrators were quickly able to analyze traffic utilization of their entire enterprise network, as well as segment by segment. They were also able to detect a number of segments that were generating high error rates that were caused by hardware problems. Consequently, they were able to correct the problems quickly.

The POSCO network administrators told us that they preferred to use our system over their existing expensive, hard-to-use commercial management tools for the configuration, fault and performance management of their networks. From this reaction, we are convinced that our management solution is moving in the right direction and deserves more research and development.

We have also applied the same monitoring and reporting system in an academic campus network environment (specifically at POSTECH) for a period of 13 months. The POSTECH campus network is not as large or complex as that of POSCO. However, the POSTECH network also consists of FDDI networks and over 80 Ethernet interconnected subnets, connecting over 2,000 computing devices (ranging from mainframes, to group servers, to workstations, to PCs). The result obtained from applying our system for monitoring the POSTECH campus network has also been very effective and it is also a network management system preferred by the POSTECH network administrators over the existing commercial network management tools.

7. CONCLUSIONS

We have examined the requirements needed for providing an inexpensive, easy-to-install, easy-to-learn, and easy-to-use, but secure and powerful enterprise network monitoring and reporting solution. We have chosen Web technology and existing management facilities (such as SNMP agents) to provide such a solution. We have presented the design of a Web-based enterprise network monitoring and reporting system. For implementation, we have utilized freely available tools on the Internet and added useful features such as security, automatic network discovery and configuration, sensitive network map generation, network device CPU load monitoring and active sublet reporting.

Our experience in using the system for monitoring two very large enterprise networks for over one year has been positive. Network administrators in both enterprises preferred our system to their existing expensive, hard-to-learn and hard-to-use traditional commercial management tools.

Our future work includes incorporating facilities to our system to manage telecommunication networks. Since telecommunication network devices are generally instrumented with CMIP agents rather than SNMP agents, it involves adding the CMIP agent access mechanism to the current MRTG+ system. It also involves selecting common performance indicators that can be found on CMIP agents.

We believe our approach is moving in the right direction for making network management easier for the administrators but at the same time providing an inexpensive and powerful solution.

ACKNOWLEDGMENTS

The authors would like to thank the Ministry of Education of Korea for its financial support toward the Electrical and Computer Engineering Division at POSTECH through its BK21 program.

REFERENCES

1. HP, HOP OpenView, *http://www.hp.com/openview/*
2. IBM, TME 10 NetView, *http://www.tivoli.com/products/netview/*
3. Sun Microsystems, Solstice SunNet Manager, *http://www.sun.com/solstice/em-products/network/sunnetmgr.html*
4. Cabletron Systems, Spectrum for Open Systems, *http://www.cabletron.com/spectrum/*
5. T. Berners-Lee, R. Cailliau, J. Groff, and B. Pollermann, World-Wide Web: The Information Universe, *Electronic Networking*, Vol. 1, No. 2, pp. 52–58, Spring 1992.
6. K. Arnold and J. Gosling, *The Java Programming Language*, Addison-Wesley, 1996.
7. W. Stallings, *SNMP, SNMPv2, SNMPv3, and RMON 1 and 2*, Third Edition, Addison-Wesley, 1999.

8. ITU-T, Information Technology, Common Management Information Protocol (CMIP)-Part 1: Specification, Recommendation X.711, 1991.
9. DMTF, DMI 2.0 Specification, *http://www.dmtf.org/tech/specs.html*
10. J. T. Park and J. W. Hong, Web-based customer network management, *Proc. of the First Enterprise Networking Min-conference (ENM'97)*, Montreal, Canada, pp. 160–169, June 1997.
11. J. Y. Kong and J. W. Hong, A CORBA-based management framework for distributed multimedia services and applications, Technical Report PIRL-TR-97-1, POSTECH, Korea, March 1997.
12. J. W. Hong, J. Y. Kong, T. H. Yun, J. S. Kim, J. T. Park and J. W. Beak, Web-based Intranet services and network management, *IEEE Communications Magazine*, Vol. 35, No. 10, pp. 100–110, October 1997.
13. F. Barillaud, Luca Deri, and Metin Feridun, Network management using Internet technologies, *Proc. of the Fifth IEEE/IFIP International Symposium on Integrated Network Management (IM'97)*, San Diego, California, pp. 61–70, May 1997.
14. M. Maston, Using the World Wide Web and Java for network service management, *Proc. of the Fifth IEEE/IFIP International Symposium on Integrated Network Management (IM'97)*, San Diego, California pp. 71–84, May 1997.
15. N. Anerousis, An architecture for building scaleable, Web-based management services, *Journal of Network and Systems Management*, Vol. 7, No. 1, pp. 73–104, March 1999.
16. J. P. Martin-Flatin, L. Bovet, and J. P. Hubaux, JAMAP: a web-based management platform for IP networks, *Proc. of the Ten IFIP/IEEE Int. Workshop on Distributed Systems: Operations & Management (DSOM'99)*, Zurich, Switzerland, October 1999.
17. DMTF, WBEM Initiative. Available at ⟨*http* : *//www.dmtf.org/wbem/*⟩, February 2000.
18. DMTF, Common Information Model (CIM) Specification, Version 2.2. June 1999. Available at: ⟨*http* : *//www.dmtf.org/spec/cim_spec_v*22/⟩.
19. DMTF, Specification for the Representation of CIM in XML. Version 2.0, July 1999. Available at: ⟨*http* : *//www.dmtf.org/download/spec/xmls/CIM_XML _Mapping*20.*htm*⟩.
20. DMTF, Specification for CIM Operations over HTTP. Version 1.0, August 1999. Available at: ⟨*http* : *//www.dmtf.org/download/spec/xmls/CIM_HTTP _Mapping*10.*htm*⟩.
21. Sun Microsystems, Java Management Extensions White Paper, Revision 01, June 1999.
22. Sun Microsystems, Java Dynamic Management Kit, *http://www.sun.com/software/java-dynamic*, 2000.
23. W3C, HTML 4.01 Specification, W3C Recommendation, December 1999.
24. J. Gettys *et al.*, Hypertext Transfer Protocol—HTTP/1.1, IETF RFC 2616, June 1999.
25. John Bloomers, *Practical Planning for Network Growth*, Hewlett-Packard Professional Books, 1996.
26. K. McCloghrie and M. Rose, Management information base for network management of TCP/IP-based Internets: MIB-II, RFC 1213, March 1991.
27. T. Oetiker, MRTG homepage, *http://ee-staff.ethz.ch/ ~oetiker/webtools/mrtg/mrtg.html*
28. B. Kumar, *Broadband Communications*, McGraw-Hill, 1995.
29. A. Leinwand, Accomplishing Performance Management with SNMP, *The Simple Times*, Vol. 1, No. 5, pp. 1–5, Nov/Dec 1992.
30. A. Leinwand, Corrections to Volume 1, Number 5, *The Simple Times*, Vol. 2, No. 5, p. 15, Sep/Oct 1993.

James Won-Ki Hong is an associate professor in the Dept. of Computer Science and Engineering, POSTECH, Pohang, Korea. He received a Ph.D. degree from the University of Waterloo, Canada, in 1991 and an M.S. degree from the University of Western Ontario in 1985. Since joining POSTECH in 1995, he has worked on various research projects on network and systems manage-

ment, with a special interest in Web, Java, and CORBA technologies. His research interests include network and systems management, distributed computing, and multimedia systems. He has served as Technical Chair for IEEE CNOM from 1998 to 2000. He was technical co-chair of NOMS 2000 and APNOMS'99. He is a member of IEEE, KICS, KNOM, and KISS.

Sung-Uk Park received the B.S. degree in Electronics Engineering from Kyungpook National University in 1990 and the M.S. degree in Computer and Communications Engineering from Graduate School for Information Technology, POSTECH, Korea, in 1998. His research interests include enterprise network and systems management and Internet/Intranet applications development. He currently works at IBM Korea.

Young-Min Kang received the B.S. degree in Computer Science from Soong Sil University, Korea in 1997 and the M.S. degree in Computer Science and Engineering from POSTECH, Korea, in 1999. His research interests include network and systems management. He currently works as a software engineer at Ifeelnet Inc. in Korea.

Jong-Tae Park received the B.S. degree from Kyungpook National University, Korea and the M.S. 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 a professor. He is the chair of IEEE ComSoc Technical Committee on Information Infrastructure (TCII). His research interests include telecommunications network management, wireless Internet and service management.