MCC
9651M13

박 성 욱, Sung-Uk Park,
Web-based Enterprise Network Traffic Monitoring and Analysis, **웹 기반의 엔터프라이즈 네트워 트래픽 모니터링 및 분석**, **정보통신학과**, 1998, 55P,
Advisor. Won-Ki Hong, Text in Korean.

ABSTRACT

Today's enterprise networks are composed of multiple types of interconnected networks. On top of these enterprise networks, there exist various systems and services supporting a wide variety of applications within an organization. Providing a secure, reliable and efficient operating environment to support the organization's daily activities and its business is the most challenging task faced by operations and management staff today.

In order to provide such an environment, 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 for managing networks. What is needed desperately is a set of simple, uniform, ubiquitous tools for managing networks. Web-based management is a promising approach that can provide such a solution.

This thesis focuses on the use of Web technology for the purposes of enterprise network traffic monitoring, analysis and reporting. In this thesis, we first examine the requirements for enterprise network traffic monitoring, analysis and reporting, and then present a 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 on using such a system for traffic monitoring of two large enterprise networks (POSCO and POSTECH).

목 차

표 목차

수식 목차

그림 목차

1. 서론

1.1 연구 배경

오늘날의 엔터프라이즈 (Enterprise 또는 기업) 네트웍은 여러가지 네트웍 장비와 시스템 그리고 다양한 작업수행을 위한 서비스 프로그램들로 구성되어 있다. 시스템의 증가와 운영되는 프로그램의 다양성으로 인하여 엔터프라이즈 네트웍은 그 규모와 복잡성이 나날이 확대되고 있다. 이러한 가운데 안전하고, 효과적이며, 안정된 컴퓨팅 환경의 제공은 네트웍 및 시스템 관리자들이 당면한 중요한 과제이며, 기업 경쟁력의 핵심인 정보기술 (Information Technology)을 통한 생산성과의 연계성도 갖고 있다.

최근까지도 네트웍 관리자들은 여러가지 종류의 관리 툴들을 (HP OpenView [1], IBM NetView [2], SunNet Manager [3], Cabletron Spectrum [4] 등) 사용하여 그들의 네트웍을 모니터링하고 컨트롤하여 왔다. 그러나 이러한 기존의 툴 들은 가격이 비쌀뿐 아니라 배우고 사용하기도 쉽지가 않았다. 그 툴을 배우고 익히는데 수개월 씩 소요되었고, 적용하는 것도 단지 트래픽 모니터링이나 장애 (fault) 모니터링과 같은 몇 가지 기능에 국한되었다. 심지어는 네트웍 관리자들의 잦은 이직 (changing job)으로 네트웍 환경 및 툴 사용에 대한 재교육 비용이 추가로 발생하게 되어, 엔터프라이즈 네트웍 관리를 위한 사용하기 쉽고 짧은 기간에 운영할 수 있는 관리 툴의 필요성이 더욱 절실해 지고 있다.

한편, World-Wide Web (WWW or Web) [5] 은 지난 몇 년간 인터넷을 통하여 단순하지만 강력하고 다양한 데이타 (text, graphic, image, voice 및 video) 검색기능을 가지고 혁신적인 발전을 해 왔다. Web browser 는 정보에 접근하는 사실상의 표준 절차 (*de facto* standard mechanism)으로 자리 잡았고, Java [6]로 개발된 프로그램은 Web browser 를 통하여 배포되어 수행되는 등 기존의 플랫폼 제한적인 운영형태에 대해 많은 변화를 가져왔다. 이러한 새로운 인터넷 기술들을 엔터프라이즈 네트웍 관리에 적용함으로써 오늘날 당면하고 있는 여러가지 문제점들을 해결할 수 있다. 즉 기존 네트웍 관리 툴의 플랫폼 의존적인 특성으로 구입 및 운영 비

용이 높고, 복잡한 사용형태로 운영자 스스로 지속적인 사용과 기능향상이 어렵고, 콘솔중심의 집중된 운영으로 원거리 지역에서의 관리가 어려운 것을 해결할 수가 있다는 것이다.

Web 기반의 관리는 정보검색에 적합하고 세계적으로 통일된 하나의 사용자 인터페이스 (Web browser)를 제공하고, 이를 통하여 여러가지 인터넷이나 인트라넷 어플리케이션을 수행한다. 대부분의 Web 관련 기술들이 이미 인터넷상에서 검증되어 일반 사용자들에게도 친숙하며, 기존의 여러가지 표준 관리 프레임웍들을 (SNMP [7], CMIP [8], DMI [9] 등) 수용 할 수 있는 가능성이 있다.

최근에 이러한 Web 기술을 적용하여 네트웍을 관리하고자 하는 연구가 활발히 진행 중이다. 기존 네트웍 관리시스템 제작 업체들은 (예를 들어, Cabletron 의 Spectrum [4], HP 의 OpenView [1], IBM 의 TME10 [2], SUN 의 Solstice [3] 등) 그들의 제품에 Web 기술을 접목시킨 새로운 제품들을 개발하고 있고, University of Twente [10] 와 SNMP Research [32] 같은 몇몇 대학과 연구소에서도 자체적으로 Web 을 기반으로 하는 네트웍 관리시스템을 개발하고 있다. Helsinki 의 Telecom Finland 와 같은 곳에서는 기존에 적용하고 있던 시스템인 HP OpenView 를 Web browser 기반의 관리시스템으로 대체하여 운영 중에 있다 [33]. 현재 Web 을 기반으로 한 관리시스템의 두 가지 큰 흐름이 있다. 하나는 SUN 사가 주축이 되어 개발한 Java Management API (JMAPI) [20, 21]를 이용한 관리시스템의 개발추세와 Microsoft, Intel, CISCO, Compaq 및 BMC software 사 등이 주축이 되어 제안한 Web-based Enterprise Management (WBEM) [18, 19]이 있다.

이 논문에서는 이러한 관리 시스템들의 추세와 더불어 엔터프라이즈 네트웍 자원 관리에 필요한 관리시스템의 기능 요건을 정의 하고, 기존의 관리 시스템들이 제공하는 일반적인 트래픽 모니터링 및 리포팅 기능들을 구현하여 실제환경에 오랜 기간 적용하여 봄으로써, Web 기반의 관리시스템이 기존의 플랫폼 의존적인 관리시스템들에 비해 기업 환경에 얼마나 손쉽게 적용가능 한지를 살펴보고, 이의 문제점 및 개선방향에 대해서도 알아 보겠다. 이러한 Web 기반의 관리시스템이 제공하는 기능들의 검증을 통하여 경제적이고도 효율적인 네트웍 자원 관리시스템

에 대한 발전가능성을 알아보고자 한다.

1.2 연구 목적 및 글의 구성

이 논문에서는 Web 기반의 관리시스템이 기존 관리시스템의 단점들을 어떻게 풀어 갈 수 있는지를 살펴보고자 한다. 특별히 엔터프라이즈 네트웍 자원의 효율적인 관리를 위해 필요한 네트웍 트래픽 모니터링, 분석 및 리포팅 기능을 제공하기 위한 시스템의 요건과 구현에 대해 살펴 보겠다. 일반적으로 네트웍 장비와 링크 등의 네트웍 자원에 대한 사용현황을 모니터링함으로써 네트웍 사용현황 및 추세를 파악하고 문제가 되는 곳에 대한 진단과 더불어 향후 네트웍의 확장을 위한 용량계획 (capacity planning)에도 도움을 줄 수가 있다. 이와 같이 네트웍을 모니터링 한다는 것은 기업 내 사용자들에게 안정된 네트웍 대역폭을 제공하기 위한 네트웍 관리의 기초 과정으로써, 모니터링을 통한 데이타의 수집과 수집된 데이타 및 기업 환경에 맞는 적절한 기준을 바탕으로 현상을 분석하고 미래를 예측하며 또한 이러한 결과를 의사결정에 도움이 되는 보고서형태로 제공하는 것은 네트웍 관리시스템이 가져야 할 기본 요소이기 때문이다.

이 논문은 다음과 같이 구성되어 있다. 1 장은 서론으로 연구배경 및 목적을, 2 장에서는 관련 연구로서 일반적인 네트웍 관리시스템의 기능과 표준화 추세, Web 기반의 관리시스템의 사례 등에 대해 알아보고, 3 장에서는 설계 부분으로, 먼저 기능은 강력하면서도 값싸고, 배우기 쉽고, 사용하기 쉬운 네트웍 트래픽 모니터링 시스템의 요구 사항에 대해 알아 보고, 필요한 요구 사항을 바탕으로 Web 기반의 엔터프라이즈 네트웍 트래픽 모니터링을 위한 설계모델의 구조를 제시하고, 엔터프라이즈 네트웍 트래픽 모니터링, 분석 및 리포팅을 위해 인터넷 관리 프로토콜인 SNMP 의 성능 변수 (performance parameter)들을 이용한 트래픽 분석방법 및 지침 (Guideline)을 제시하였다. 4 장에서는 3 장을 토대를 하여 공개된 (public domain) 툴을 바탕으로 부가적인 기능을 추가하여 실제로 기업환경에서 수개월간 이를 적용, 운영한 엔터프라이즈 네트웍 트래픽 모니터링 시스템의 구현에 대해 살펴보고, 5 장에서는 이에 대한 사용 경험을, 6 장에서는 이번 연구를 요약하고 앞으로의 연구 방향을 제시한다.

2. 관련 연구

이 장에서는 Web 기반의 엔터프라이즈 네트웍 모니터링 시스템을 구현하는데 있어서 필요한 일반적인 네트웍 관리시스템의 특징과 기업에 있어서 이러한 관리시스템을 선정하는 절차에 대해 간단히 살펴보고, Web 기술을 이용한 여러가지 관리 시스템에 대해 알아 보겠다.

2.1 네트웍 관리 (Network Management)

네트웍의 구성요소는 장비들과 (router, hub, bridge 등) 그들을 연결하는 케이블 (cable) 및 커넥터 (connector)를 포함하여 이들 장비의 연결을 지원하는 기능들로 구성된다. 여기에 각 장비를 통하여 연결된 시스템들 (Server, PC 등)도 넓은 범위에서는 네트웍의 구성요소로 볼 수 있다. 네트웍 관리의 목표는 한마디로 이들 구성요소 들이 네트웍 상에서 정상적인 동작을 보장하도록 하는 것으로 기술적인 측면에서는 사용자의 요구를 충족하는 최대의 대역폭 (bandwidth)를 제공하는 것이다. 네트웍 관리에 대한 개괄적인 절차를 살펴보면 아래 <그림 1>과 같다.

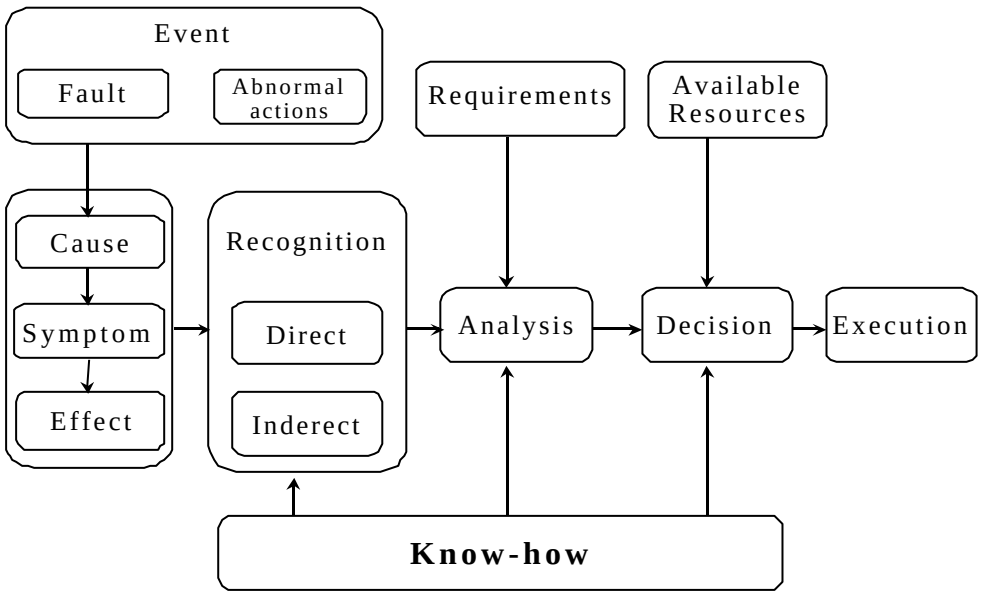

그림 1. 네트웍 관리에 대한 구조적 시각

정상적인 네트웍 작동을 방해하는 사건, 예를 들어 갑자기 response 가 급격히 떨어진다든가 하는 사건이 발생하면, 이는 원인(cause) - 증상(symptom) - 결과(effect)의 형태로 사용자나 네트웍 관리자에게 인지되며, 이를 해결하기 위해 적절한 조치를 간구한다. 이때 관리자의 경험이나 주변의 요구사항이 참고가 되어 부하가 몰리는 라우터의 경로를 조정하거나, 처리능력이 우수한 새로운 시스템을 도입하거나, 트래픽 발생이 많은 batch job 의 실행을 일과 후로 조정하는 등의 가능한 대처방안을 결정한 후 조치를 취한다.

일반적으로, 네트웍 관리를 통해 얻을 수 있는 것은 아래와 같은 것들이 있다.

? 네트웍 및 전산 자원의 효율적인 조절능력
? 장치, 사용자, 응용 프로그램 등의 지속적 증가에 따른 복잡성 조절 능력
? 증가된 자원과 더 나은 서비스를 사용자에게 제공 가능
? 여러 가지 요구에 대한 자원의 배분 및 조절능력
? 사용자 및 어플리케이션의 가용성을 높이기 위해 네트웍 downtime 감소
? 비용을 조절하기 위한 효과적인 자원 활용능력

2.1.1 기능 및 구조

네트웍 관리시스템의 주요 기능을 International Organization for Standardization (ISO) [34]에서는 다음과 같이 다섯 가지 영역으로 분류하고 있다.

? **구성 관리 (Configuration management) :**
시스템의 지속적인 운영을 위해 관리 객체인 네트웍 장비들의 상태정보를 수집하고 분석 및 통제하는 기능이다.

? **장애 관리 (Fault management) :**
시스템의 비정상적인 작동에 대한 감지 (detection), 격리 (isolation) 및 정정(correction)을 위한 정보수집 및 분석, 시험, 장애위치 찾기 기능 등을 수행한다.

? **성능 관리 (Performance management) :**

시스템의 효율적인 운영을 위해 관리 대상인 자원들의 상황을 감시, 평가 및 조정한다.

? **계정 관리 (Accounting management) :**

시스템이 제공하는 서비스의 사용을 측정하여 그에 대한 사용료를 부가하기 위해 관리객체에 대한 비용산출 등의 과금 관리기능을 수행한다.

? **보안 관리 (Security management) :**

시스템 및 관리 객체들에 대한 접근 권한을 설정, 관리하는 기능을 한다. 즉, 네트웍 자원에 대해 인가된 자에게 허가된 권한만을 부여하도록 한다.

네트웍 관리시스템은 manager-agent paradigm **을 기초로 한 네트웍의 모니터링과 제어를 위한 툴들의 집합체이다. 일반적인 관리 구조는 아래의** <그림 2>**와 같이** manager **와** agent **로 구성된다.**

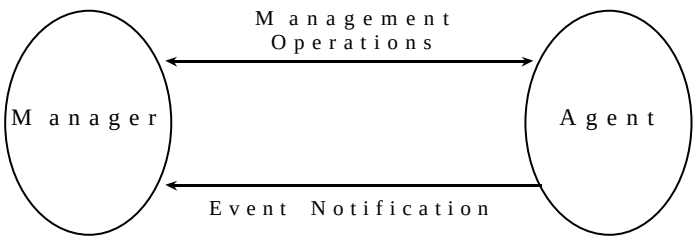

그림 2. **일반적인 관리 구조**

Manager **는 구성, 결함 분석, 모니터링 등을 수행하는 역할을 하며,** agent **는** agent **자신의 환경 관리나 객체들과의 관계를 정립하는 역할을 한다.** <그림 3>**은 이러한 구조를 가진 네트웍 관리 시스템을 보여 준다.**

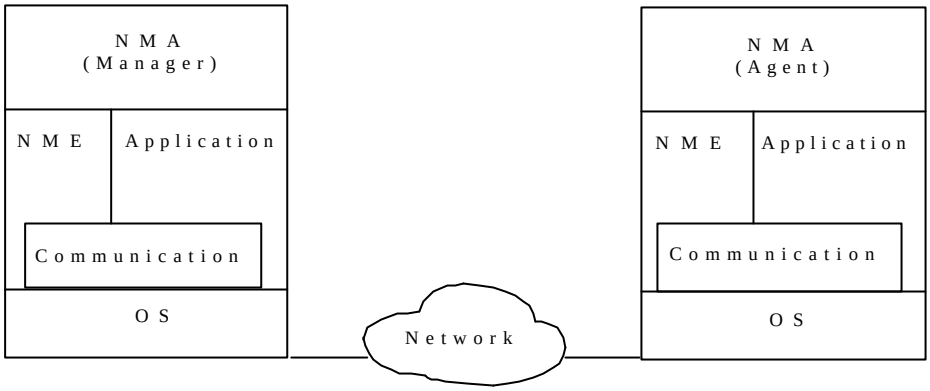

그림 3. 네트웍 관리 시스템 구조

Manager 혹은 Network Control Host 라 불리는 것은 소프트웨어의 집합인 Network Management Application (NMA)를 포함하고 있으며 각 시스템의 Network Management Entity (NME)를 제어하는 역할을 수행하고 네트웍 관리의 효율을 높이기 위해 한 개 이상의 Manager 를 둘 수도 있다. Agent 는 End-User Application 를 지원하며 각 노드의 통신 서비스도 지원하는 모듈을 의미한다. 각 manager 와 agent 의 구성 요소를 보면 공통적으로 NME, Application, Communication Software, OS 등을 지니고 있으며, manager 는 위에서 논한 NMA 를 가지고 있다.

2.1.2 기술 및 표준화 추세

1980 년대 초반부터 네트웍의 사용은 급격히 팽창해 왔다. 네트웍 기술을 사용함으로써 얻는 생산성과 비용의 이득 때문에 많은 회사들은 새로운 네트웍 기술과 제품이 소개되자마자 채택할 정도가 되었다.

기존의 네트웍 기술과 새로운 네트웍 기술이 서로 공존하면서 네트웍은 점차 확장되었으며 이와 같은 네트웍의 확장은 여러 가지 문제점들을 야기시키게 되었다. 즉, 기존의 네트웍을 운용, 유지, 보수하기 위해서는 기존의 네트웍 기술을 소지하고 있는 관리자가 필요하였고 새로운 네트웍을 위해서는 새로운 네트웍 기술을

소지하고 있는 관리자가 필요하였다. 뿐만 아니라 네트웍이 계속해서 확장되고 많은 장치들과 자원들이 네트웍에 연결됨에 따라 사람의 힘으로 그 전체 네트웍을 관리한다는 것은 많은 시간과 비용을 초래하게 만들었다. 특히 여러 회사에서 만들어진 많은 장치들과 자원들을 관리해야 한다는 것은 네트웍 관리를 더 어렵게 만들었다 [35].

따라서 네트웍 관리에 드는 많은 시간과 비용을 줄이고 여러 회사에서 만들어진 많은 장치들과 자원들을 관리할 수 있는 표준화된 네트웍 관리 시스템이 필요하게 되었다. Standard Management Frameworks 는 각 표준기구에 따라 아래와 같은 것들이 있다.

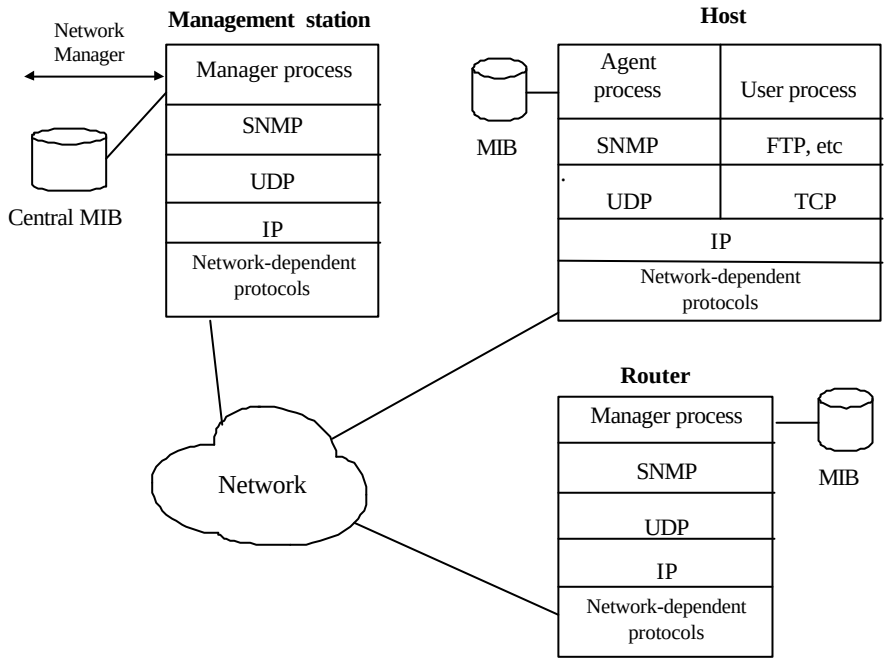

그림 4. SNMP를 이용한 네트웍 관리 구조

2.1.3 Internet Network Management Framework

Internet Management Framework [36]는 Internet Engineering Task Force (IETF)에 의해

표준화 작업이 진행되고 있는 네트웍 관리를 위한 관리 구조, 관리 정보의 구조, 관리 프로토콜 등에 대한 표준 안의 모음이다. 네트웍 장비뿐만 아니라 인터넷에 연결된 모든 장비를 관리하는 산업계 표준 프로토콜이다. 관리 프로토콜의 이름을 따 'SNMP 기반의 네트웍 관리 체제'라고도 일컫는 것으로 주로 컴퓨터 통신망의 관리에 이용되고 있다 [그림 4].

OSI Management Framework 가 네트웍 관리를 위한 궁극적인 모델이라면 Internet Management Framework 는 과도기적이고 임시적인 모델로 구상되었다. 하지만 Internet Management Framework 의 간결성으로 인해 현재로서는 Internet Management Framework 가 널리 이용되고 있는 실정이다. SNMPv1 [42]은 1990 년에 표준으로 채택되어 대부분의 워크스테이션, 브릿지, 라우터, 스위치, 허브 등에 SNMP Agent 가 장착되어 사용되고 있으며 1993 년에 보안 기능과 몇 가지 기능이 첨가된 SNMPv2 [42]의 드래프트 (draft)가 만들어졌다. 현재는 SNMPv3 의 정의가 진행 중이다.

2.1.4 OSI Network Management Framework

OSI Management Framework [34]는 Internet Management Framework 와 마찬가지로 네트웍 관리를 위한 표준 안들의 모음으로 International Organization for Standardization (ISO)과 International Telecommunications Union, Telecommunication Section (ITU-T)가 공동으로 표준화 작업을 진행하고 있다. 관리 서비스와 프로토콜, 관리 정보의 정의뿐만 아니라 시스템 관리 기능에 대한 정의도 포함하고 있다. 관리 프로토콜로는 Common Management Information Protocol (CMIP) [42]을 사용하고 관리 정보는 객체 지향적 방법으로 정의된다.

이와 같은 OSI Management Framework 의 관리구조 및 프로토콜은 Internet Management Framework 보다 많은 기능이 제공되지만 너무 복잡한 구조를 가진다는 단점이 있다. 그래서 일반적인 통신망 보다는 전화망 관리 (Telecommunications Network Management)에 널리 사용되고 있다. 실제로 이 표준에 따라 구현된 시스템으로는 OSIMIS [44], DSET [45], IBM TMN Agent Toolkit [46]등이 있다.

2.1.5 DMI Management Framework

Desktop Management Interface (DMI) [9]는 Desktop Management Task Force (DMTF)에서 제정한 시스템 관리를 위한 표준 안이며, 현재 DMI v2.0 까지 나와 있다. Management Application (MA), Service Provider (SP), Component Instrumentation (CI) 이 세 부분 사이의 인터페이스를 정의하고 있으며, DMI v2.0 에서는 MA 와 SP 사이의 인터페이스에 Remote Procedure Call (RPC)를 사용하여 원거리에서도 관리를 가능하게 하였다. 각 구성요소는 Management Information Format (MIF)이라 불리는 언어에 의해 그 특징들이 기술되고 이것이 SP 에 저장됨으로써 MA 가 그 구성요소를 관리할 수 있게 된다. DMI 는 주로 PC 관리를 위해 사용되며, 이를 지원하는 제품으로는 Intel 의 LanDesk, IBM 의 SystemView, NEC 의 EMM 등이 있다. 또한 DMI 를 이용한 시스템 관리 소프트웨어 개발자들을 위해서 Intel 과 IBM 에서 Software Development Kit (SDK)를 제공하고 있다.

2.1.6 Web-based Management Frameworks

다음으로 Web-Based Enterprise Management (WBEM) [17, 18, 19]이라는 컨소시엄은 Microsoft, Intel, BMC Software, Cisco, Compaq 등의 기업들이 주축이 되어 1996 년 7 월에 결성되었다. WBEM 의 주요 목표는 기본적인 관리 분야, 즉 결함이나 구성, 성능, 보안, 계정관리 등의 관리기능을 제공하는 관리시스템이 만들어 질 수 있는 관리구조를 제공하고 기존의 관리구조에서 사용되고 있는 다양한 종류의 관리 정보들을 일관성 있게 표현할 수 있는 데이터 모델을 제공하는 것이다. 이와 같은 목표를 가지고 WBEM 에서는 웹 기반의 관리 구조인 HyperMedia Management Architecture 와 관리 정보 모델로 HyperMedia Management Schema (HMMS), HMMS 를 억세스하기 위한 프로토콜인 HyperMedia Management Protocol (HMMP), 그리고 관리 정보들을 수집하고 여러 가지 관리 프로토콜 사이에서 프락시의 역할을 하는 HyperMedia Object Manager (HMOM)을 제안하고 있다 [그림 6].

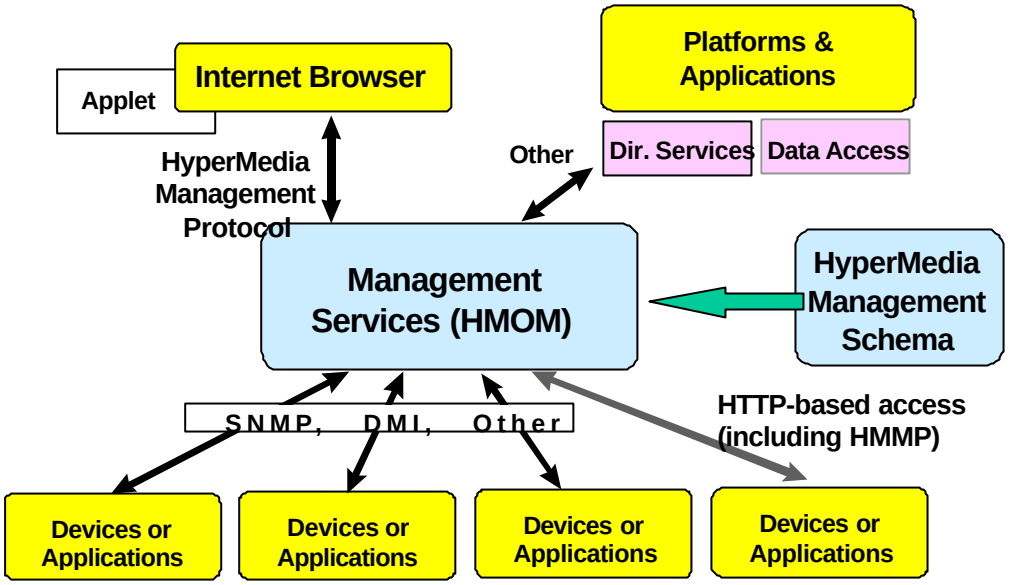

그림 5. 제안된 WBEM 구조

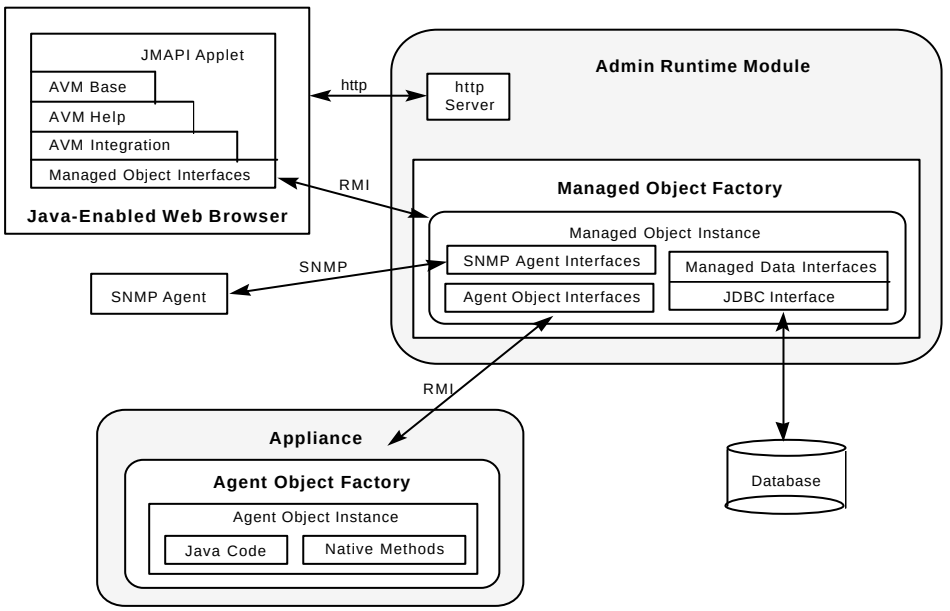

그림 6. JMAPI 구조

그리고 Sun Microsystems 에서는 관리 분야에서 발휘할 수 있는 Java 의 능력을 개발하기 위해 자바의 기본 클래스들을 확장시켜 발표한 것이 Java Management Application Programming Interface (JMAPI) [20, 21]이다. JMAPI 는 네트웍 관리뿐만 아니라 시스템 관리나 서비스 관리에서도 플랫폼과 운영체제, 네트웍 프로토콜에 독립적인 관리 시스템을 개발할 수 있도록 해 준다 [그림 6].

이러한 웹 기반의 네트웍 관리 기술은 획일적이고 일관성 있는 네트웍 관리 사용자 인터페이스를 지원하며 기존의 다양한 네트웍 관리 기술들을 쉽게 포함시킬 수 있는 장점이 있다. 뿐만 아니라 복잡한 대규모의 네트웍 관리를 위해 적합한 분산 네트웍 관리를 가능하게 해 줄 수 있다.

앞에서 살펴본 Internet Network Management Framework 과 OSI Network Management Framework 및 DMI Management Framework 의 특징을 Web 표준 프로토콜인 HyperText Transfer Protocol (HTTP) [24]과 비교해 보면 아래 <표 1>과 같다.

Features	CMIP	SNMP	DMI	HTTP
Information Model	Object-oriented	Limited Object-oriented	Limited Object-oriented	Hypermedia
Language	GDMO	SMI	MIF	HTML
Architecture	Manager-Agent Manager-Manager cascaded	Manager-Agent Manager-Manager	Manager-Agent	Client/Server
Operations	M-gets, M-set, M-action, M-create, M-delete, M-Event-Report	Get, Set, implicit action (side effects) , Trap	Get, Set, implicit action, Add, Delete, Event	Get, Post, Link Trap not supported
Communication Mode	Transaction-oriented Request/Response	Request/ Response	Request/ Response	Asynchronous Request/ Response
Addressing	MIT with OID Scoping / Filtering	MIT with OID at leaves of the tree	component/group /attribute Ids	URL
Management Applications	Five Functional Areas	Not Specified	Not Specified	Pulg-In's Java
Standardization Body	ITU-T, ISO/OSI	IETF	DMTF	IETF

표 1. Standard Management Framework 들의 비교

2.1.3 기존 네트웍 관리시스템의 문제점과 Web 기술의 특징

오늘날의 네트웍은 여러 가지 다양한 자원들의 연결로 이루어져 있고, 이들을 관리하는 툴들은 업체마다 그들 고유의 플랫폼을 근간으로 구성되어 타 제품들과의 통합된 네트웍 관리기능을 제공하기 어려운 실정이다. 최근 들어 HP OpenView, IBM NetView, SunNet Manager 같은 제품들은 general-purpose network management tool 을 제공하고는 있으나 아직은 미약하다. 지금까지의 네트웍 관리 시스템들은 아래와 같은 문제점들이 있다.

? 기존에 널리 확산되어 있는 네트웍, 시스템, 어플리케이션 관리 인프라(infrastructure)의 업체간 비호환으로 소프트웨어 개발자들이 유사한 환경에서 중복적인 어플리케이션의 개발이 많았다.
? 관리 플랫폼상에서 원거리 데이타와 툴에 대한 접근이 어려웠다. 즉, console 중심의 집중 관리 형태로 관리형태에 제한이 있었다.
? 증가하는 수요자의 needs 를 충분히 반영할 만큼 유연한 플랫폼이 아니다. 즉, 시스템에 대한 확장 및 운영에 따른 제약사항이 많았다.

이러한 문제점과 더불어 WWW 의 확산은 복잡한 어플리케이션의 사용에 대한 새로운 길을 제시하였다. 모든 플랫폼에서 Web browser 는 실행되고, 또한 많은 사람들이 이미 Web 을 사용하고 있으므로 Web 기반의 관리시스템은 기존 관리 시스템의 많은 문제점을 쉽게 해결해 줄 수 있는 대안으로 많은 사람들의 관심의 대상이 되었다. Web 기반의 관리란 Web-server 와 browser 기술 (즉, Internet 기술) 사용하여 네트웍 및 시스템을 관리하는 것을 말한다.

Web 기술은 다음과 같은 특징이 있다.

? 하나의 동일한 사용자 인터페이스를 제공한다. 즉, 사용법에 대한 교육 투자가 필요 없다.
? 거의 모든 플랫폼에서 Web browser 가 지원된다. 즉, 소프트웨어의 이식작업(porting)이 필요 없다.

? Web 서버가 중앙저장소 (central repository)역할을 하므로 소프트웨어 분배 및 유지 비용이 감소한다.

? Web page 는 멀티미디어 서비스를 포함한 여러가지 서비스를 통합하는 환경을 제공한다.

조만간 WWW 와 Java 같은 인터넷 기술은 네트웍과 시스템 관리에 있어서 중요한 역할을 할 것이다. 벌써 네트웍 관리시스템업계에서는 그들 제품의 다음 버전은 Web 기반의 관리를 가능하게 하는 쪽으로 나가고 있다 [1, 2, 3, 4, 43]. 비록 현재는 Web 기반의 네트웍 관리에 몇 가지 문제점이 (security, real-time reporting 등) 있지만 하루가 다르게 발전하는 인터넷 기술들을 바라볼 때 곧 해결되리라 보고 Web browser 가 보편적인 사용자에게 제공되듯이 네트웍 관리 작업도 그 개념이 변화되어 마치 사용자가 자기 PC 의 자원을 관리하듯 하는 시대가 도래하리라 생각된다.

2.2 Enterprise Network Management

일반적으로 Enterprise network management 란 엔터프라이즈 전역에 걸쳐 있는 통신 네트웍을 관리하는 것을 말하고, 일명 corporate network management 라고도 한다.

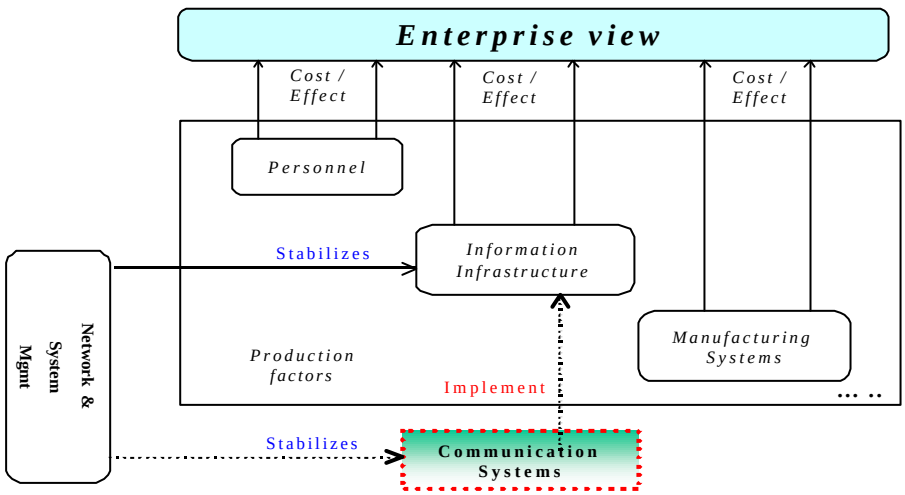

그림 7. Enterprise Network Management system 의 역할

전통적으로 볼 때, 네트웍 관리시스템은 <그림 7>에서처럼 기업의 생산에 직접적인 요인이 아닌, 기업 정보 인프라 (Infrastructure)의 한 부분인 통신 네트웍을 안정화 시키는 보조수단으로 인식되어 왔다. 그러나 최근에 들어 Business Process Reengineering (BPR)등을 통하여 정보기술(Information Technology)의 중요성이 대두되자 네트웍을 기업 경제활동을 지원하는 중요한 요소로서 인식하게 되었고, 네트웍도 기업내의 다른 시스템처럼 자원으로서의 관리 필요성을 인식하게 되었다. 기업의 입장에서는 네트웍관리에 대한 목적이 세계화(globalization), 비용절감, 수입 증대, 효과적인 고객과의 상호연결 지원, 효율적인 정보공유 등을 가능하게 하는 중요한 요소로 자리 잡게 되었다 [38].

2.2.1 기업에서의 네트웍 관리시스템 도입 절차 및 고려사항

오늘날의 관리시스템은 기존에 구축되어 있는 시스템들과의 호환성(interoperability)도 제공하고, 앞으로 나타날 시스템 또한 지금의 시스템들과 호환성을 유지해야 한다. 이러한 시스템이 되기 위해서는 무엇보다도 사용하기 쉬우면서도 표준을 지원하는 시스템이어야 한다. 일반적으로 네트웍 관리시스템을 선정하기 위한 과정을 살펴보면 다음과 같다.

1) Perform device inventory :

해당 네트웍 및 여기에 연결된 장치에 대한 조사를 하고 관리대상을 결정한다.

2) Prioritize the functional areas of network management :

네트웍 관리의 여러 기능들 (fault, security, configuration, performance 관리기능) 중 우선 순위를 결정하여 중요한 관리 기능들을 순서대로 나열한다.

3) Survey network management applications :

결정된 네트웍 관리기능을 지원하는 어플리케이션들을 찾아본다.

4) Choose the network management platform :

앞에서 찾은 어플리케이션 중 기업의 환경에 맞는 관리 플랫폼을 선정한다. 예를 들어, 집중관리, 계층관리, 분산관리 등의 관리 방식에 적합한 플랫폼

을 결정한다.

한편, 일반적으로 네트웍 관리는 90%의 네트웍관련 데이타 수집 작업과, 10%의 분석 및 결정과정으로 이루어진다고 한다. 그러므로 중요한 것은 얼마나 필요한 데이타를 쉽게 수집할 수 있느냐는 것이다. 관리 프로토콜에 대한 선택은 바로 이러한 점에서 네트웍 관리시스템을 선정하는데 있어서 중요한 사항이다. 오늘날의 기업들 관점에서 보는 네트웍 관리프로토콜은 다음의 표준들을 만족해야 한다고 본다.

? 인터넷을 통한 네트웍 모니터링이 가능해야 한다.
? 독점 기술이 아닌 표준화가 된 프로토콜이어야 한다.
? 증명된 기술로 데이타 통신시장의 여러 업체들로부터 받아들여진 프로토콜이어야 한다.
? 최소한의 보안은 지원해야 한다.
? 무엇보다도 구현에 따른 비용이 비싸지 않아야 한다.

이러한 관점에서 볼 때 SNMP 는 적절한 네트웍 관리 프로토콜로 아직 까지는 인정된다고 본다.

2.3 Web-based Management

Web 을 기반으로 하는 네트웍 관리 시스템들의 사례를 살펴 봄으로서 현재의 기술 동향 및 추세를 알아 본다.

2.3.1 Web기반의 분산 멀티미디어 서비스 관리

Object Management Group (OMG)에서 하드웨어 및 소프트웨어 플랫폼에 독립적인 어플리케이션 개발을 위해 표준으로 제공한 Common Object Request Broker Architecture (CORBA) [31]기반의 응용 서비스들의 개발이 분산환경에서 점차 확산되는 추세에 있다. 이에 포항공대 분산처리 환경연구실에서 분산환경에서의 응용서비스 관리에 대한 연구의 일환으로 CORBA 기반의 분산서비스를 위한 관리시스템을 개발하였는데 (CorbaMan [11]) 여기에서도 management application 을 OrbixWeb [39]을 사용하여 Web 기반의 인터페이스를 제공하고 있다 [그림 8].

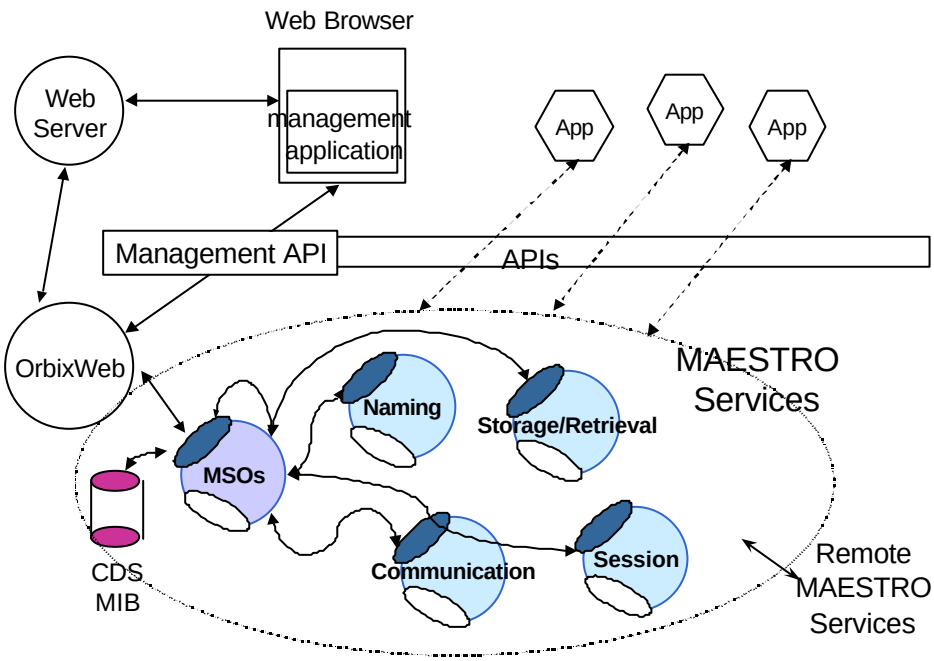

그림 8. CorbaMan 의 구조

MAESTRO [40]는 분산 멀티미디어 응용 프로그램들을 보다 쉽게 개발, 운용, 관리할 수 있도록 다양한 멀티미디어 서비스를 제공하는 시스템이다. MAESTRO 에는 분산 멀티미디어 서비스를 제공하는 CORBA 기반의 분산 서버들이 존재한다. 이 서버들이 제공하는 서비스에는 이름 서비스(Name service), 통신 서비스 (Communication service), 세션 서비스 (Session service), 저장/인출 서비스 (Storage/Retrieval service)가 있다. 이 서비스들을 제공하는 서버들도 역시 CORBA 객체들로 구성된다.

뿐만 아니라, MAESTRO 는 멀티미디어 응용프로그램에서 공통적으로 필요한 멀티미디어 데이타 처리와 멀티미디어 관련 장치의 조작을 위한 API 를 제공함으로써 멀티미디어 응용 프로그램의 개발을 쉽게 하는데 목적을 두고 있다.

<그림 8>은 이러한 분산환경의 응용 서비스인 MAESTRO 를 관리하기 위한 CorbaMan 의 구조를 나타내고 있는데, 그 구성을 살펴 보면, Service Objects (SOs), Management Service Objects (MSOs), Management Information Objects (MIOs), 그리고 Web 기반의 인터페이스를 가진 관리 어플리케이션으로 이루어져 있다.

MAESTRO 는 IONA Orbix 2.0 [41]을 이용하여 개발되었고, CorbaMan 의 여러 CORBA 객체들은 C++ 을 이용해 구현되었으며, 관리 어플리케이션은 Web 기반의 인터페이스를 위해 OrbixWeb 2.0.1 [39]을 이용한 Java 로 구현되었다. OrbixWeb 은 Java 로 구현된 관리 어플리케이션이 Orbix 로 구현된 CORBA 객체들에 대한 호출 메소드 (method)를 제공한다. 그래서 관리 어플리케이션은 OrbixWeb 클라이언트와 Java 애플릿 (applet)의 형식을 동시에 가지고 있다.

2.3.2 Web기반의 ATM Switch 관리

Netherlands 의 University of Twente [10]에서 'Web-based Management (WbM)'라는 논문을 최근에 발표했는데 여기에서 주장하는 것은, Web 기술의 여러가지 특성들 (security, effciency, cost, user friendliness 등)을 이용하여 네트웍 관리시스템으로서의 가능성을 증명하기 위하여 SURFnet4 라는 사용자 네트웍을 Web 기반으로 관리할 수 있는 ATM 스위치관리 시스템을 프로토타입 (prototype)으로 개발하였다. 여기에서는 Web-based Management (이하 WbM)을 두 가지 형태로 정의하고 있는데 System WbM [그림 9]과 Proxy WbM [그림 10]로 구분하였다.

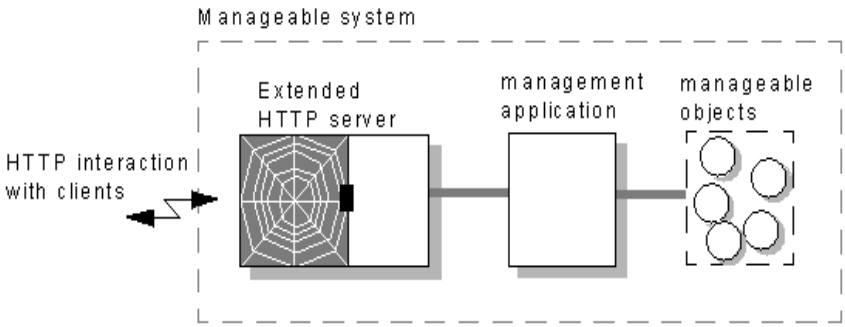

그림 9. System WbM

<그림 9>의 System WbM 은 관리대상인 시스템이나 장비에 HTTP 서버가 포함되어 (embedded)있어서 HTTP 를 통하여 클라이언트에서 직접 해당 장비를 관리할 수 있다. 여기서 클라이언트는 Web browser 이고, HTTP 가 바로 관리 프로토콜이 된다.

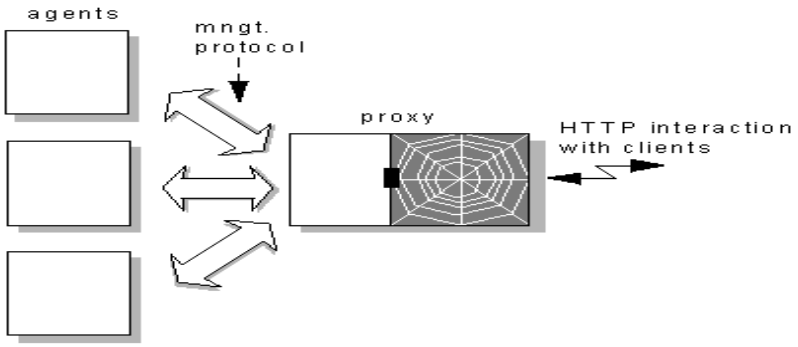

그림 10. Proxied WbM

<그림 10>의 Proxied WbM 은 하나 이상의 관리 프로토콜과 HTTP 간에 중계자(proxy)를 두고, 여기에 HTTP 서버의 기능을 확장하여 HTTP 와 다른 관리 프로토콜간의 데이타 맵핑 (mapping)작업을 하도록 한다. 그러므로 관리 대상이 되는 시스템이나 장비에는 HTTP 서버가 수행될 필요가 없고 단지 기존에 사용하던 SNMP agent 등을 통해 관리 되기만 하면 된다. 이 연구에서 구현한 시스템도 이러한 Proxied WbM 을 근간으로 하고 있다.

또한 기존의 platform 중심의 관리시스템과 Web 기반의 관리시스템의 차이를 비교하고, 문제점과 그 해결 방안을 제시하고 있는데 전체적인 것은 <표 2>와 같다.

	Platform-based Mgmt	Web-based Mgmt (WbM)
Efficiency	SNMPv1 의 요구/응답 시간 제한 - getnext request 만의 retrieval - getbulk request 가 SNMPv2 에서 제공되나 현재 대부분의 agent 가 SNMPv1 사용함	Management application 과 agent 가 local LAN 에 있도록 한다 - proxy 형식의 WbM
		Ineffciency of HTTP - HTTPv1.0 에서는 매 HTTP 요구마다 TCP 연결을 재설정, HTTPv1.1 에서 상시연결 (persistent connection)기능 지원
Scalability	Utilization problem of SNMPv1 agents -동시에 많은 SNMP 요구를 처리못함	Proxy WbM 이용 - 강력한 Web 서버 플랫폼 제공 (서버의 사용율은 증가)
Abstraction Level	One central mgmt station 이 기본이나 RMON agent 이용한 분산관리지원, IETF 의 DISMAN working group 에서 SNMP 이용한 분산관리 연구중	Proxy WbM 이용한 proxy 와 agent 간 분산된 관리기능 수행 - proxy 는 SNMP, client 는 HTTP - 예) client 요구에 대해 proxy 에서 통계 데이타 처리 지원
Expandability	Source recompile 및 모듈 재분배 등의 필요로 일반적으로 어려움	Proxy WbM 에서 script 언어 지원 - 수정 및 분배가 용이
Cost	운영시스템 (하드웨어,소프트웨어) 비용이 높다	Web server, CGI script 지원비용 및 운영비용이 낮다
User friendliness	X-Window (X11/Motiff)기반의 GUI	Web browser 기반의 GUI

표 2. Platform 중심의 관리시스템과 Web 기반의 관리시스템 비교

이 연구에서는 <그림 11>과 같은 Proxy WbM 방식의 프로토타입을 구현했는데 WbM 의 핵심적인 부분인 CGI script 및 collector 를 Tcl [41]을 이용하여 개발했다.

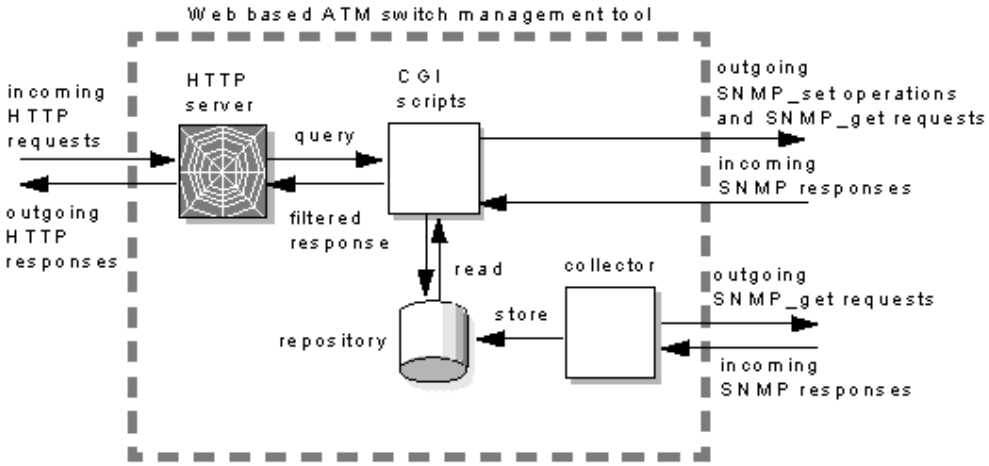

그림 11. Web 기반의 ATM switch 관리시스템 구조

CGI script 를 통하여 해당 ATM 스위치에 대한 트래픽 데이타를 저장소 (repository)에 요일별로 디렉토리 파일형식으로 저장하고, collector 는 매일 한번씩 이러한 요일별 데이타를 읽어서 평균치를 계산하여 다시 저장소에 저장한다.

그러나, 이 시스템은 하나의 프로토타입 (prototype)으로 실제 네트웍에 장기간 사용된 것은 아니다.

3. Web기반의 네트웍 트래픽 모니터링 시스템 설계

이 장에서는 Web 기반의 엔터프라이즈 네트웍 트래픽 모니터링 시스템을 구현하기 위해 필요한 일반적인 요구 사항들을 정리하고, 그에 맞는 아키텍쳐를 설계한다. 또한 구현될 시스템의 실질적인 사용효과를 높이기 위해 네트웍 성능분석을 위한 성능 변수들 (Performance Parameters)을 정의하고, 이들을 이용하여 트래픽 분석을 위한 몇 가지 안내지침 (Guideline)을 제시한다.

3.1. 시스템 요구 사항

일반적인 Web 기반의 네트웍 트래픽 모니터링 시스템이 가져야 할 요구 사항들 중 실제로 기업 환경에서 필요하다고 생각되는 것들을 살펴보면 다음과 같다.

3.1.1 Network Configuration Detection and Discovery

엔터프라이즈 네트웍은 router, bridge, hub, switch 등과 같은 여러가지 네트웍 장비들을 사용하여 다양한 네트웍 구성 형태 (topology)를 이룬다. 네트웍 구성관리 기능은 임의의 엔터프라이즈 내에 있는 네트웍 장비에 대한 자동검색 기능과 해당 네트웍의 backbone 및 subnetwork 의 구성형태를 알 수 있도록 해 준다. 비록 엔터프라이즈 네트웍의 구성형태는 자주 변경되지는 않지만, 변경이 생겼을 경우 이를 발견하고 필요한 정보의 수정을 통하여 지속적인 네트웍 관리기능을 수행할 수 있어야 한다.

3.1.2 Traffic Data Gathering and Logging

오늘날 대부분의 네트웍 장비들은 그 자체에 agent 라는 관리 모듈을 탑재하고 있다. 컴퓨터 네트웍 장비인 경우는 SNMP agent, 전화통신망(telecommunication network) 장비인 경우는 CMIP agent 가 기본적으로 탑재되어 해당 장비에 대한 모니터링을 수행하면서 그들의 management information base (MIB)를 관리한다. MIB 에

는 네트웍을 통하여 해당 네트웍 장비에 발생한 트래픽의 양에 대한 정보와 장비 자체에 대한 정보가 관리된다. 이러한 정보는 관리 프로토콜을 통하여 관련된 MIB 변수를 호출함으로써 얻을 수 있고, 이렇게 해서 얻은 정보는 관리 시스템 내의 임의 영역 (Management Information Repository, MIR)에 저장되어 분석을 위한 데이타로 사용될 수 있어야 한다.

3.1.3 Traffic Data Analysis

MIR 에 저장된 데이타를 읽어서 일정한 기간 동안의 네트웍 트래픽 사용량을 입력과 출력에 대한 최대값과 평균값 등의 통계처리 된 값으로 표현할 수 있고, 이러한 결과들은 다시 MIR 에 저장되어 추후 분석을 위한 자료로 활용되거나 네트웍 사용율 등을 계산할 때 사용될 수 있어야 한다.

3.1.4 Network Fault Detection and Reporting

네트웍 사용에 대한 두 가지 형태의 리포팅이 가능해야 하는데, 네트웍 장치의 오동작 (malfunction) 상태와 급격한 트래픽 증가와 같은 긴급한 문제 발생시에 이를 즉각 관리자에게 통보하여 적절한 조치를 할 수 있게 하는 것과, 주기적으로 (daily, weekly, monthly 등) 네트웍 트래픽 사용에 대한 보고서를 자동으로 생성할 수 있어야 한다. 이러한 기능은 최종관리자 스스로도 네트웍 사용에 대한 파악이 가능하도록 하여, 현재의 네트웍 사용에 대한 분석과 적절한 대응을 통하여 문제 발생을 조기에 막을 수 있도록 한다.

3.1.5 Web-based Management

네트웍에 대한 현재상태 및 분석된 정보를 네트웍 관리를 담당한 부서에서 쉽게 볼 수 있도록 해야 한다. 즉, Web browser 를 사용하여 언제 어디서나 조회할 수 있도록 하기 위해서는 Web 서버가 MIR 에 있는 데이타들을 접근 (access)할 수 있고, 이러한 데이타들은 HTML [23]형식으로 제공되어야 한다. 기존의 콘솔 (console) 기반의 Network Management System (NMS)은 단지 컴퓨터센터와 같이 제

한된 장소에서만 네트웍 관리가 가능했었고, 이 또한 특정한 하드웨어 플랫폼의 콘솔상에서만 가능했었다. Web browser 를 통하여 언제 어디서나 네트웍을 관리할 수 있다는 것이 바로 Web 기반의 관리 시스템이 가지는 주된 장점이다.

3.1.6 Data Protection

여러가지 엔터프라이즈 네트웍에 대한 data 들은 때로 아주 중요한 자료가 될 수 있으므로 이들에 대한 접근을 통제할 수 있어야 한다. 즉 인가된 사용자만 접근이 가능하도록 하는 보안 (security) 대책을 제공해야 한다. 필요한 경우 인가된 사용자도 해당 데이타에 대한 접근기록 (access log)을 관리할 수 도 있다. Web 은 누구나 접근 가능하므로 이러한 보안대책이 먼저 수립된 후 Web 기반의 Enterprise Network Traffic Monitoring System 을 적용한다.

3.2 시스템 구조

대부분의 네트웍 관리시스템은 manager 와 agent 간의 request-reply interaction paradigm 을 가지는데, 본 연구에서 사용될 Web 기반의 네트웍 트래픽 모니터링 시스템도 동일한 paradigm 을 갖는다. <그림 12>는 이러한 Web 기반의 엔터프라이즈 네트웍 트래픽 모니터링 시스템의 설계 구조 (design architecture)로 관리시스템 (manager system), 관리 장비들 (managed devices), 관리정보 저장소 (management information repository, MIR), 그리고 Web 서버와 Web browser 의 크게 4 개의 그룹 모듈로 나뉜다.

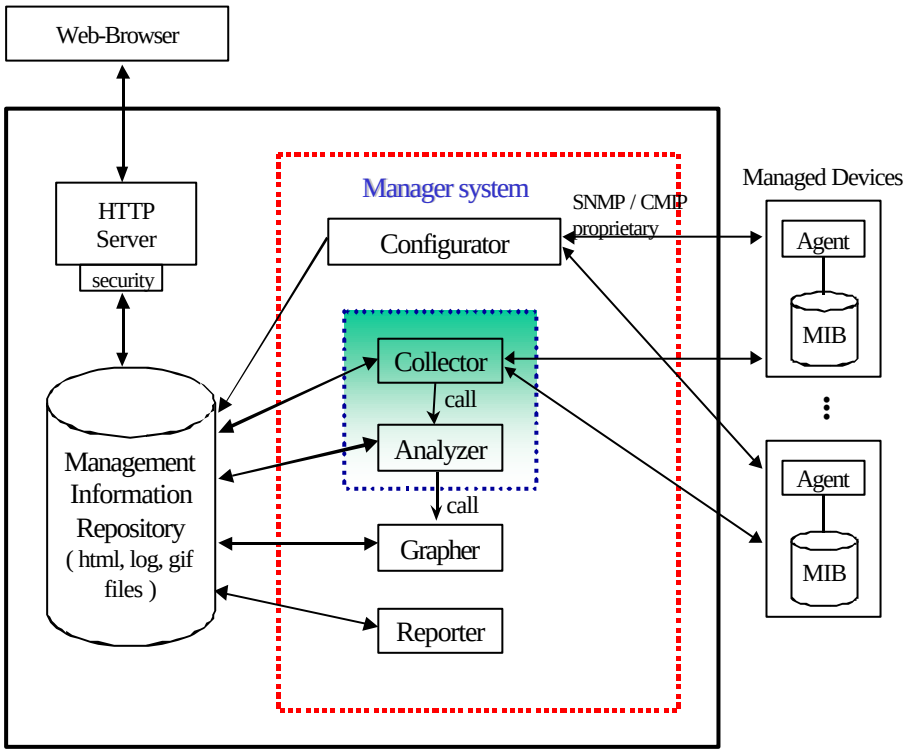

그림 12. Web 기반의 네트웍 트래픽 모니터링 시스템 설계 구조

3.2.1 관리 시스템 (Manager system)

관리 시스템은 전체 시스템의 핵심으로 실질적인 관리 행동들 (management activities)가 수행되는 부분이다. 관리 통신프로토콜들 (SNMP, CMIP 또는 proprietary protocol)을 사용하여 관리 장비 안에 있는 agent 로부터 네트웍 구성 및 트래픽에 대한 데이타를 가져오고, 이것들을 MIR 라는 일종의 데이타베이스에 저장한다. 관리 시스템은 다음의 여러 모듈들로 구성된다:

? **Configurator:** 관리되어야 할 네트웍 장비들을 찾고, 각 장비의 구성 정보에 대한 데이타베이스를 구축한다. 이들 구성 정보에는 장비 자체에 대한 정보와 각 장비의 인터페이스에 대한 정보들로 이루어 진다.

? **Collector:** 'configurator'에 의해 생성된 구성정보를 바탕으로 관리 장비들의 agent 로부터 필요한 트래픽 데이타를 모은다. 각 agent 로부터 데이타를 가져오는 폴링 (polling) 주기는 필요에 따라 조정할 수 있으나 일반적으로는 수분 ~ 수시간까지 다양하다. 정확한 분석을 위해서는 짧은 주기가 좋으나 너무 짧은 폴링주기 (polling interval)은 과다한 관리 트래픽을 발생시켜 오히려 전체 네트웍에 심각한 영향을 줄 수도 있다. 적용하는 네트웍의 규모에 따라 조정하는 것이 좋다. 'Collector'를 통하여 모여진 데이타들은 MIR 에 저장된다.

? **Analyzer:** 'collector'에 저장된 데이타를 주기적으로 읽어서 여러가지 트래픽 분석작업을 한다. 일반적으로 해당 네트웍 인터페이스에 대한 트래픽 사용의 최대값, 최소값, 그리고 평균값을 계산한다. 이 결과는 다시 MIR 에 저장되어 아래의 'grapher'에 의해 사용되거나 또 다른 분석을 위한 데이타로 사용된다.

? **Grapher:** MIR 에 저장된 데이타를 이용하여 다양한 graphical output 들 (즉, GIF 나 JPEG format 의 graph 또는 histogram)을 생성한다. 여기서 생성된 graphical output 들은 Web browser 를 통하여 검색할 수 있도록 HTML 파일에 포함된다.

? **Reporter:** 보통 두 가지 형태의 리포팅 (reporting)을 한다. 첫째는 네트웍에 대한 문제를 발견하여 alarm, e-mail, paging 등의 형태로 관리자에게 알려주는 것과, 둘째는 주기적으로 네트웍 사용에 대한 여러가지 현상분석보고서를 생성하여 상위 관리자에 대한 보고서로 활용될 수 있도록 한다. 대표적인 예로 주요 네트웍 인터페이스별 사용율을 표로서 나타낸 보고서 등이 있다.

3.2.2 관리 장비들 (Managed Devices)

엔터프라이즈 네트웍을 구성하는 네트웍 구성요소로서 switch, router, bridge 그리고 hub 등이 있다. 대부분의 네트웍 장비들은 management agent (즉 SNMP agent, CMIP agent 또는 proprietary agent)를 가지고 있는데 이들은 관리자의 요구에 대한 응답으로 해당 장비에 대한 정보를 관리자에게 전달하기도 하고, 해당 장비 자체에 문제가 생겼거나 미리 정의된 조건 만족 시 대리자 스스로 관리자에게 상태에 대한 정보를 전달하기도 한다.

3.2.3 관리정보 저장소 (Management Information Repository)

관리시스템에 의해 발생되는 모든 데이타들 (network configuration, traffic, analyzed graphic, reporting data 등)이 저장되는 곳으로 Web browser 들의 요구에 의해 Web 서버가 이곳 (관리정보 저장소)을 접근하여 필요한 정보를 보여 준다. 관리정보 저장소는 로컬 파일 시스템 (local file system)위에 구축될 경우는 관리시스템과 Web 서버간에 일종의 공통 저장소 (common repository)로 사용되고, NFS (Network File System)나 AFS (Andrew File System)와 같은 분산파일시스템 위에 구축될 경우 Web 서버와 별개의 저장소 (repository)로 사용될 수도 있다.

3.2.4 Web 서버와 Browser

Web 기반의 관리시스템에서 Web 서버는 관리시스템에 의해 생성된 정보를 Web

browser 에게 제공하는 역할을 한다. Web 서버는 HTTP 1.0 [24] 이상을 지원하고, 불법적인 사용자의 접근을 막고 인가된 사용자만 해당 정보를 볼 수 있도록 하는 보안대책 (security mechanism)도 지원해야 한다. Web browser 는 PC 나 다른 웍스테이션 (workstation)에서 지원되는 어떠한 것도 가능하나 Web browser 를 통하여 관리시스템이 제공하는 최신 정보를 자동적으로 (dynamic 하게) 보기 위해서는 HTML 의 META tag 를 이용한 client-pull [30] 기능이 지원되어야 한다.

3.3. 네트웍 성능 변수들 (Network Performance Parameters)

네트웍 성능 모니터링은 관리 대리자가 제공하는 MIB 들로부터 네트웍 성능에 관련된 정보를 수집한다는 것을 의미한다. 이러한 MIB 정보 중에 성능과 관련된 MIB 변수들을 모아 보면 크게 3 가지 그룹으로 나뉘어 진다. <표 2>는 이러한 네트웍 성능 변수들을 1)대역폭 (bandwidth)에 대한 사용율 (utilization)로 표현되는 성능표시 그룹 (performance indicator class), 2) 충돌 (collision)이나 프로토콜 부하 (protocol overhead)와 같이 효율 (efficiency)과 관련된 성능저하표시 그룹 (performance degradation indicators class), 3) 에러로 인한 문제상태를 나타내는 연결 및 데이타 전송문제 표시 그룹 (connectivity and data transmission problem indicators class)으로 나누고, 이들 각각에 해당하는 상세한 MIB 변수 및 적용 장비에 대한 내용이다 [25].

구 분	성능 변수	네트웍 장비
Performance Indicators	Interface octets in and out Interface unicast frames in and out Interface nonunicast frames in and out CPU utilization Forwarding rate	Hub, Switch, Bridge, Router, Server
Performance Degradation Indicators	Ethernet transmit collisions Ethernet deferred transmissions TCP retransmissions	Hub, Switch, Bridge, Router, Server
Connectivity and Data Transmission Problem Indicators	Interface CRC errors in and out Interface lost carrier Interface disconnect Ethemet excess retries (16 consecutive collisions)	Bridge, Router, Server

표 3. 네트웍 성능측정을 위한 표시자들

네트웍 장비들 (hub, switch, Bridge, router 등)이 가지고 있는 관리 대리자를 통하여 위에서 말한 네트웍 성능 표시자들을 수집할 수 있다. 이러한 정보들은 MIB 변수

나 다른 객체로 표현되어 agent 에 의해 지속적으로 모니터링 된다. <표 4>는 인터넷 표준 관리 프로토콜인 SNMP MIB 에 정의된 네트웍 성능 표시자 관련 변수 들이다 [26].

변수명	설 명
ifInNUcastPkts	상위 layer 프로토콜로 전달된 non-unicast 패킷의 수
ifInUcastPkts	상위 layer 프로토콜로 전달된 subnetwork-unicast 패킷의 수
ifInOctets	인터페이스를 통해 전달된 총 octet 수. Framing character 포함
ifInDiscards	상위 layer 프로토콜로 전달되지 못하고 거부(discard)된 inbound 패킷수. (에러로 인한 경우가 아닌, 버퍼영역을 비우기 위해 거부된 패킷도 포함됨)
ifInErrors	에러로 인하여 상위 layer 프로토콜로 전달되지 못한 inbound 패킷의 수
ifOutErrors	에러로 인하여 전달되지 못한 outbound 패킷의 수
ifOutQLen	Output queue 의 길이 (패킷으로 나타냄)
ifForwDatagrams	해당 entity 에 대해 최종 IP 목적지가 아닌 input datagram 의 수로 최종 목적지를 찾기 위한 forwarding 이 필요함. 해당 entity 가 IP gateway 역할을 하지 않는 경우, 이 값은 해당 entity 를 지나 source-route 된 패킷의 수를 의미함.
ipOutDiscards	문제없이 전달되었지만 거부 (discard)된 output IP datagram 수. (버퍼영역 부족 등이 원인)
tcpRetransSegs	재전송된 총 TCP segment 의 수

표 4. 네트웍 성능에 관련된 SNMP MIB 변수들

3.4. 네트워 트래픽 분석 지침 (Network Traffic Analysis Guidelines)

네트워 트래픽 모니터링의 중요한 목적 중 하나는 효율적이고도 안정된 네트워 환경을 사용자들에게 제공하는 것이다. 이러한 목적을 달성하기 위해서는 지속적인 네트워 모니터링을 통한 트래픽의 분석이 이루어져야 한다. 네트워 트래픽 분석결과는 일정한 기준 위에서 현재 네트워의 상태에 대한 좋고 나쁨을 판단하고, 문제의 소지가 있는 곳에 대한 예방을 위한 자료로 활용된다. 적용되는 기준은 사용자들의 네트워 환경에 따라 그 차이는 다소 나지만 일반적으로 여겨지는 기준을 여기서 제시해 보고자 한다. 여기서는 대부분의 네트워 장비들에 포함된 SNMP agent 를 통해 얻을 수 있는 MIB 변수들을 이용하여 대표적인 네트워 성능 표시자인 네트워 사용율에 대한 계산 방법 및 분석 지침을 제시해 보고자 한다. 참고로, 네트워 장비에 agent 가 있는 경우를 중심으로 트래픽 정보를 수집하고 분석하기 때문에 agent 가 없는 경우는 무시된다. 즉, FDDI 에 agent 가 없는 hub 이 연결되어 있다면 이들에 대해 측정되지 못한 트래픽 오차가 발생할 수 있다.

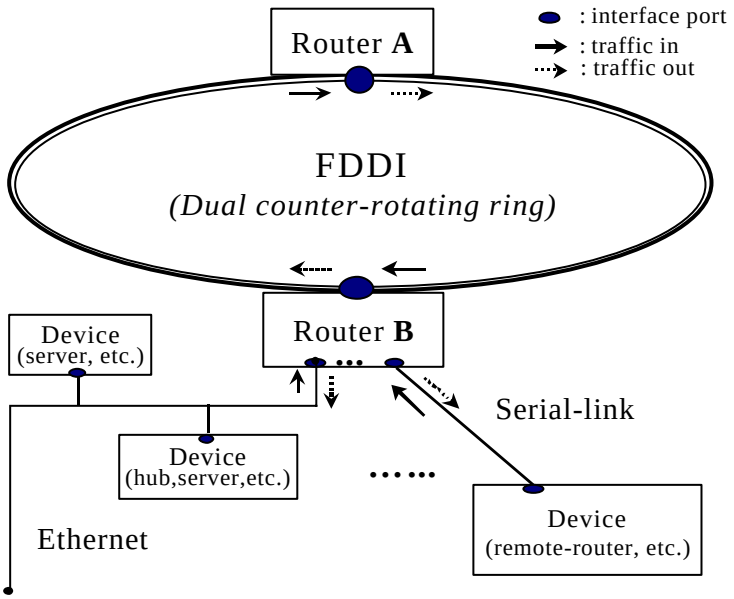

그림 13. 간단한 엔터프라이즈 백본 네트워

<그림 13>은 전형적인 엔터프라이즈 네트웍의 백본 (backbone)일부를 나타낸 것으로, FDDI를 백본으로 라우터들이 연결되어 있고, Ethernet subnet과 WAN serial link가 FDDI 라우터의 인터페이스에 연결되어 있다. 이러한 네트웍의 트래픽을 분석하기 위해서는 네트웍 장비들의 상태에 대한 모니터링 뿐만 아니라 이들 장비에 연결된 인터페이스들을 통하여 출입하는 트래픽에 대한 모니터링도 필요하다. 여기서 트래픽의 방향은 대리자가 있는 장비의 port 중심으로 결정한다. 예를 들어 FDDI 네트웍의 경우 <그림 13>에서 router A 의 output 트래픽은 router B 의 input 트래픽으로 본다.

3.4.1. 네트웍 장비의 사용율에 대한 지침들

일반적으로 네트웍 장비들 (bridge, hub, router 등)의 사용율 (utilization)은 CPU process load로 나타낼 수 있다. High process load는 그만큼 해당 장비의 idle time이 적다는 것을 의미하고, 만약 해당 장비가 이들 process 를 처리할 만큼 충분한 성능이라면 일정한 값 이상 (이론상으로는 100% 가까운)을 넘지 않는 process load는 문제가 되지 않는다. 그러나 장비에 대한 높은 사용율은 해당 장비를 통과하는 패킷들의 거부 (discard), 재전송 (retransmit)등을 반복하여 야기시키고 이것은 결국 전체 시스템의 response time 을 증가 시켜 성능 저하를 가져온다. 즉, 장비가 해당 패킷을 처리할 수는 있으나 시스템 큐 (queue)에 대기해야 한다면 이것은 전체 네트웍 성능을 저하시키는 요인이 된다. 그러므로 해당 장비에 대한 사용율을 측정할 때는 process load 와 queue length 를 같이 비교하여 사용율로 나타내는 것이 좋다.

3.4.2. 네트웍 링크 사용율에 대한 지침들

네트웍 링크 (link) 사용율은 인터페이스 형태에 따라 차이가 있다. 일반적으로 FDDI 와 T1 같은 경우는 80 ~ 90%의 사용율을 나타내더라도 해당 링크의 성능저하는 크지 않으나 Ethernet 인 경우는 40%만 넘으면 심각한 성능저하를 가져온다.

<그림 13>의 전형적인 엔터프라이즈 백본 네트웍을 기준으로 FDDI, Ethernet, serial link 의 경우에 대한 사용율을 구하는 공식을 정의해 보고 트래픽 분석을 위한 지침도 함께 제시해 본다 [29].

? **FDDI 트래픽 분석**

일반적으로 백본용의 FDDI 는 primary ring 과 secondary ring 의 이중으로 된 counter-rotating ring [28]의 구성을 가진다. 대부분의 경우에 primary ring 은 데이타 전송을 위해서 사용되고, secondary ring 은 primary ring 에 문제가 생긴 경우 백업용으로 사용된다.

FDDI 네트웍의 사용율을 구하기 위해서는 FDDI 에 직접 연결된 모든 네트웍 장비들 (위 그림에서는 router A,B)의 인터페이스 port 를 통과하는 입출력 트래픽을 모두 합하여 FDDI 네트웍 용량 (100Mbps)으로 나눈다. 이때, FDDI 는 timed-token protocol [28]을 사용하므로 여기서 다시 2 로 나누어 준다. 이것을 수식으로 나타내면 <수식 1>과 같다.

$$Utilization(\%) = \frac{1}{2} \sum_{devices} (total\ bits\ sent + total\ bits\ received) / bandwidth$$

$$= \frac{1}{2} \sum_{devices} \frac{[(ifInOctets_{x+t} - ifInOctets_x) + (ifOutOctets_{x+t} - ifOutOctets_x)] * 8}{(sysUpTime_{x+t} - sysUpTime_x) * ifSpeed * 100}$$

where x *is previous SNMP polling time and* t *is polling interval in seconds.*
ifSpeed *is the speed of device's port, bandwidth, in bits per seconds.*

수식 1. FDDI 사용율

이렇게 해서 구한 사용율 (utilization)이 90%를 넘을 경우 일반적으로 해당 FDDI 가 부하가 걸린다고 본다. 그러나 peak 사용율이 90%를 넘는 경우가 가끔 있다고 문제가 되는 것이 아니라 이들의 특정 시간대 (보통 일과 시간인 9 시 ~ 18 시) 평

균이 90%를 넘을 경우, 해당 FDDI 에 문제가 있다고 본다고 판단한다. 이런 경우 FDDI 백본이 과부하로 인하여 전체 네트웍 사용자들에게 심각한 response time 저하를 가져오는데 이를 해결하기 위한 대책들 (FDDI ring 분리, router upgrade, router reconfiguration 등)이 시급히 마련되어야 한다.

? **Ethernet 트래픽 분석:**

Ethernet 은 broadcasting 방식의 (즉, CSMA/CD) 데이타 전송을 사용하므로, 사용율을 구하기 위해서는 router 의 Ethernet 인터페이스 port 를 통과하는 입출력 트래픽을 모두 합하여 Ethernet 의 용량 (capacity)인 10Mbps 로 나누어 준다. 이것을 수식으로 나타내면 <수식 2>와 같다.

Utilization(%)

$$? (\ total bits\ sent\ ?\ total bits\ received\ ) / bandwidth$$

$$? \frac{[(ifInOctets_{x?t}\ ?\ ifInOctets_x)\ ?\ (ifOutOctets_{x?t}\ ?\ ifOutOctets_x)]? 8}{(sysUpTime_{x?t}\ ?\ sysUpTime_x)? ifSpeed? 100}$$

where x *is previous SNMP polling time and* t *is polling interval in seconds.*
ifSpeed *is the speed of device's port, bandwidth, in bits per seconds.*

수식 2. Ethernet 링크 사용율

일반적으로 Ethernet 의 사용율이 40%를 넘을 경우, 해당 네트웍은 부하가 있다고 판단한다. 그러나 peak 사용율이 40%를 넘는 것이 문제가 아니라, 업무시간대와 같이 일정시간의 평균 사용율이 40%를 넘는다면 이것은 문제가 된다. 이런 경우는 switching hub 등의 장비를 이용하거나 Ethernet segment 를 분리하거나 불필요한 트래픽 발생요인을 찾아서 문제를 해결하여야 한다.

? **Serial 링크 트래픽 분석**

Serial 링크는 전송방식에 따라 half-duplex와 full-duplex의 2가지 방식으로 나누어진다. 즉, half-duplex인 경우는 앞에서 살펴본 Ethernet의 경우처럼 입출력 트래픽을 모두 합하여서 사용율을 계산하고, full-duplex인 경우는 동시에 트래픽을 동시에 송수신 가능하므로 이들 중 높은 쪽의 트래픽을 기준으로 사용율을 계산한다.

Serial 링크에서는 송수신 전송속도를 서로 다르게 설정할 수도 있으나 일반적으로는 같은 속도로 설정한다. 예를 들면, 56kbps serial 라인은 송신할 때 56kbps, 수신할 때 56kbps로 할 수 있다. 그러나 어느 한쪽도 56kbps 이상을 넘을 수는 없고, 받는 쪽이 부하가 걸릴 경우 일반적으로 보내는 쪽도 송신요구에 대한 지연현상이 나타나서 전체 serial 링크에 영향을 준다. Serial 링크 사용율은 설정된 링크 용량 (link capacity)를 기준으로 계산되므로 입출력 트래픽 중 높은 쪽의 데이타를 설정된 라인 용량으로 나누어서 나타낸다. 이것을 수식으로 나타내면 <수식 3>과 같다.

Utilization(%)

$$? \; Max(\; total bits\, sent,\; total bits\, received\;) / bandwidth$$

$$? \; \frac{Max[(ifInOctets_{x?t} \; ? \; ifInOctets_x),\; (ifOutOctets_{x?t} \; ? \; ifOutOctets_x)] ? 8}{(? \, sysUpTime[x ? t, x]) ? ifSpeed ? 100}$$

where x *is previous SNMP polling time and* t *is polling interval in seconds.*
ifSpeed *is the speed of device's port, bandwidth, in bits per seconds.*

수식 3. Serial 링크 사용율

Serial 링크 사용율 역시 90%를 넘을 경우 부하가 있다고 판단한다. 그러나 역시 peak 사용율이 90%를 넘는 것이 문제가 아니라, 앞에서와 같이 일정시간의 평균 사용율이 90%를 넘는다면 이것은 심각한 문제가 된다. 이런 경우는 해당 라인의 용량을 증설하거나 트래픽 경로를 수정하여 집중되는 트래픽의 분산을 유도하는 것이 필요하다.

? **트래픽 에러율 (error-rate) 분석**

Network **사용율 외에 또 다른 중요한 네트웍 성능 표시자로 트래픽 에러율** (error rate)**이 있다**. **일반적으로** FDDI **와** Ethernet **의 경우는 물리적인 연결매체와 사용 프로토콜의 신뢰성으로 인하여 에러가 거의 발생하지 않으나** serial link **인 경우는 여러가지 원인으로 트래픽 에러가 발생할 수 있다**. **연결 접점이나 라인의 불량**, **노이즈** (noise) **등의 에러가** serial link**인 경우 일반적으로 나타나는 에러이다**.

Input Error Rate (%)

$$? \frac{ifInErrors_{x?t} - ifInErrors_{x}}{totalPktsIn_{x?t} - totalPktsIn_{x}}$$

where,

$totalPktsIn ? ifInUcastPkts ? ifInBroadcastPkts ? ifInMulticastPkts,$

And, x *is previous polling time,* t *is polling interval.*

수식 4. **입력되는 트래픽의 에러율**

<**수식** 4>**는 네트웍 인터페이스에 대한 입력 에러율을 구하는 공식을 나타낸 것으로 해당 인터페이스로 입력되는 모든 트래픽 중에 에러로 분류되는 트래픽에 대한 비율로 나타낸다**. **일반적으로 에러율이** 1%**를 넘을 경우 해당 네트웍에 문제가 발생할 소지가 있다고 본다**. **그러나 간혹 발생하는 경우가 아니고 일정시간 평균적으로** 1% **이상의 에러율을 나타내는 네트웍은 엄밀히 조사해서 그 원인을 살펴볼 필요가 있다**. **이렇게 에러율이 높은 네트웍은 사용율은 높으나 효율** (efficiency)**은 낮아서 사용자들은 성능** (performance)**에 불만을 가지는 경우가 대부분이다**.

3.4.3. 기타 네트웍 분석 지침들

네트웍에 대한 모니터링은 툴(management system)이 하지만 그 결과에 대한 분석을 보고 문제를 찾고 해결하기 위한 방법을 모색하는 것은 네트웍 관리자의 몫이다. 예를 들면, 분석된 결과를 보고, 일과시간 중 네트웍 사용율이 높은 곳은 트래픽의 분산을 유도하기 위해 batch job 의 수행시간을 조정하거나, 한군데로 집중되는 트래픽은 dynamic 하게 여유가 있는 다른 경로로 우회하도록 routing table 을 수정한다던가, 사용 율이 낮은 곳의 라우터를 부하가 많은 곳과 교체한다던가 하는 작업을 통하여 전체 네트웍의 성능을 향상시키는데 기여할 수 있다.

또한 대규모 네트웍을 가진 곳에서는 지역별로 관리자들의 특성에 의해 지역 네트웍 환경이 관리되는 경우도 있다. 이런 경우는 전사적으로 체계적이고 표준화된 관리절차를 수립하여 일관된 관리를 할 필요가 있다. 특히 WAN 을 통하여 연결되는 엔터프라이즈 네트웍에서는 네트웍관리를 위한 시스템의 운영자체도 지역별로 분산시킬 필요가 있기 때문에 분산운영과 통합관리의 이점을 살릴 수 있다. 예를 들어, router 의 이름을 정하는 규칙이라던가, router 의 service level 을 설정한다던가, router 의 기타 환경을 관리할 때에 전사적으로 동일한 규칙을 정의하여 관리함으로써 운영의 편리함과 더불어 문제발생시 신속하게 해결할 수가 있을 것이다.

4. 구현

앞 장에서 정의한 요구사항과 설계구조 및 트래픽 분석방법을 바탕으로 Web 기반의 엔터프라이즈 네트웍 모니터링 시스템을 구현해 보았다. 여기에 사용된 툴은 인터넷의 public domain 에 공개된 Multi-Router Traffic Grapher (MRTG) [27] 로 필요한 몇 가지 기능을 추가하여 본 연구에 사용하였다.

사용된 시스템은 대부분의 UNIX platform 과 Windows NT platform 을 지원하나 본 연구에서는 UNIX platform 을 사용하였다. (주)삼보 마이크로시스템즈에서 SunSparc-20 기종을 개량한 'TGMI SVR-4000' machine 으로 2 개의 75Mhz CPU 와 64MB 의 메인 메모리를 가진 시스템으로 O/S 는 Solaris 2.4 를 탑재하고 있으며, local AFS 서버와 Web 서버역할을 겸하고 있고 Web 서버로는 Netscape Enterprise Server 2.0 을 사용하였다.

4.1. MRTG+ 구성 및 기능

MRTG 가 제공하는 기본 기능에 네트웍 트래픽 map 을 dynamic 하게 보여 주는 GUI 기능과 지정한 threshold 값 이상을 넘는 네트웍들에 대한 online report 생성기능과 라우터와 같은 네트웍 장비의 CPU load 를 모니터링하고 이들 데이타들에 대한 접근 제어를 위한 보안기능을 추가하여 본 연구에서는 MRTG+ 라고 부르는 <그림 14>과 같은 간단한 Web 기반의 엔터프라이즈 네트웍 트래픽 모니터링 시스템을 구현하였다.

이 시스템을 두 군데의 엔터프라이즈 네트웍에 (포스코와 포항공대) 약 6 개월간 적용하여 봄으로서 네트웍 관리시스템의 방향이 인터넷 기술을 이용한 웹을 바탕으로 더욱 더 강력하면서도 사용하기 쉬운 그러나 저렴한 가격으로 구현이 가능하다는 것을 확신하였다. 이에 대한 자세한 내용은 5 장에서 살펴보겠다.

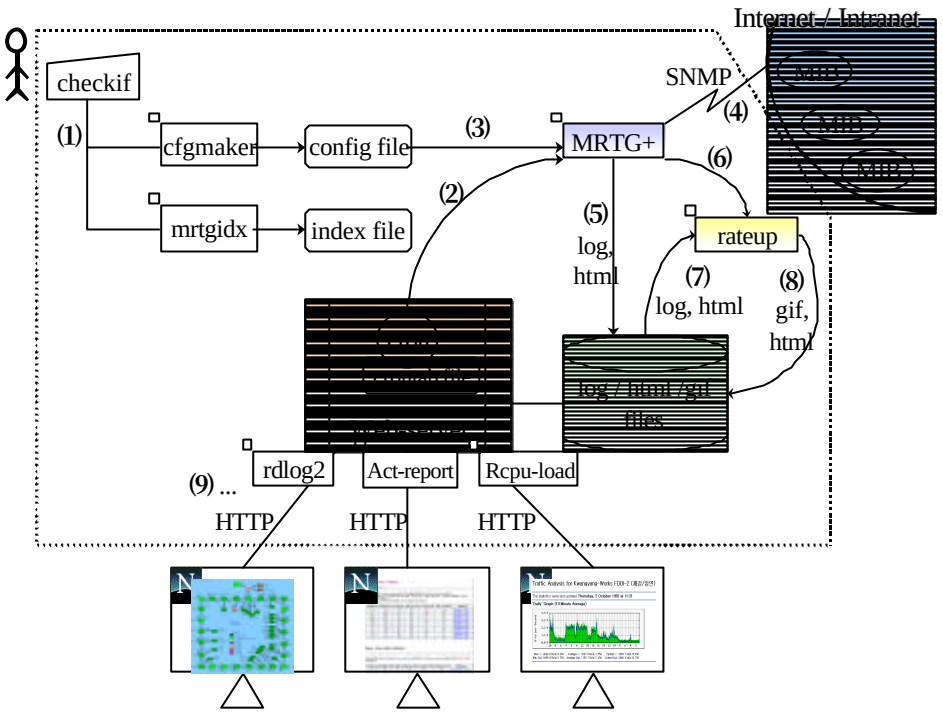

그림 14. Web 기반의 트래픽 모니터링 시스템 구조

구현된 시스템의 구조와 동작과정에 대한 전체적인 것은 <그림 14>에 나타난 것과 같다. 사용자들은 Web browser 를 이용하여 네트웍 구성에 대한 정보 및 네트웍 트래픽에 대한 리포터를 dynamic 하게 볼 수 있는데 이러한 작업이 단순히 Web 서버가 제공하는 HTML 문서를 조회하는 수준에서 이루어 진다. 구성 모듈별 상세한 기능과 동작과정을 아래에서 살펴보겠다.

? **cfgmaker [Perl] :**

설계구조에서의 'Configurator' 역할을 한다. 즉, 네트웍에 연결된 장비들을 찾아 이들의 구성 정보를 구성파일로 만든다. Practical Extraction and Report Language (Perl)로 작성되었다.

? **mrtgidx [Perl] :**

구성파일을 읽어서 HTML 형식의 인덱스 파일을 만든다. 이것은 네트웍 인터페이스에 대한 상세한 트래픽 조회용으로 사용된다.

? **MRTG+ [Perl] :**

주 프로그램으로 트래픽 데이타 추출한 후 MIR 에 저장한다. 설계구조에서의 'Collector' 역할을 수행한다.

? **rateup [C] :**

설계구조에서의 'Analyzer & Grapher' 역할을 한다. MIR 에 저장된 log 데이타를 읽어서 필요한 통계처리 (즉, 뒤에 설명될 log 파일의 consolidation algorithm 작업)를 하고 각 네트웍 인터페이스별 트래픽에 대한 GIF 형식의 그래프 이미지를 생성한다.

? **active-report [Perl] :**

설계구조에서의 'Reporter' 역할을 한다. Log 파일을 읽어서 지정한 threshold 값을 초과하는 네트웍에 대한 트래픽 보고서를 생성한다.

? **rdlog2 [C] :**

그래픽 툴인 xfig 로 작성된 파일을 log 파일과 연결하여 dynamic GUI 네트웍 트래픽 map 을 보여 준다.

? **rcpu-load [Perl] :**

라우터의 CPU load 를 그래프와 레포트로 보여 준다.

? **crontab [Text] :**

위의 각 프로그램들이 cron 에 의해 주기적으로 수행되도록 정의 해주는 script 파일이다.

? **checkif [C] :**

관리 시스템의 초기 네트웍 구성파일 생성을 자동적으로 수행하도록 한다. 전체 라우터에 대한 cfgmaker 작업과 mrtgidx 작업을 수행한다.

4.2. MRTG+ 동작 과정 및 특성

구현된 시스템의 자세한 동작과정을 단계별로 살펴보면 다음과 같다. <그림 14>의 괄호 숫자는 동작수행 순서를 나타낸다.

Step (1): 네트웍 구성에 대한 정보를 최초로 생성하는 작업으로 'checkif' 프로그램에서 전체 네트웍 장비에 대한 정보 (router name, router security 등)를 입력으로 전체 네트웍에 대한 구성파일을 생성한다. 이때 생성되는 것으로 <그림 15>와 같이 전체 네트웍 장비에 대한 인터페이스들의 인덱스파일이 HTML 문서로 생성되어 추후 네트웍 interface 별 상세한 트래픽 조회 시에 사용된다. 밑줄이 그어진 (Hyperlink 된) 인터페이스 port 가 active 한 것이고 그렇지 않은 것은 현재 disable 된 port 들이다.

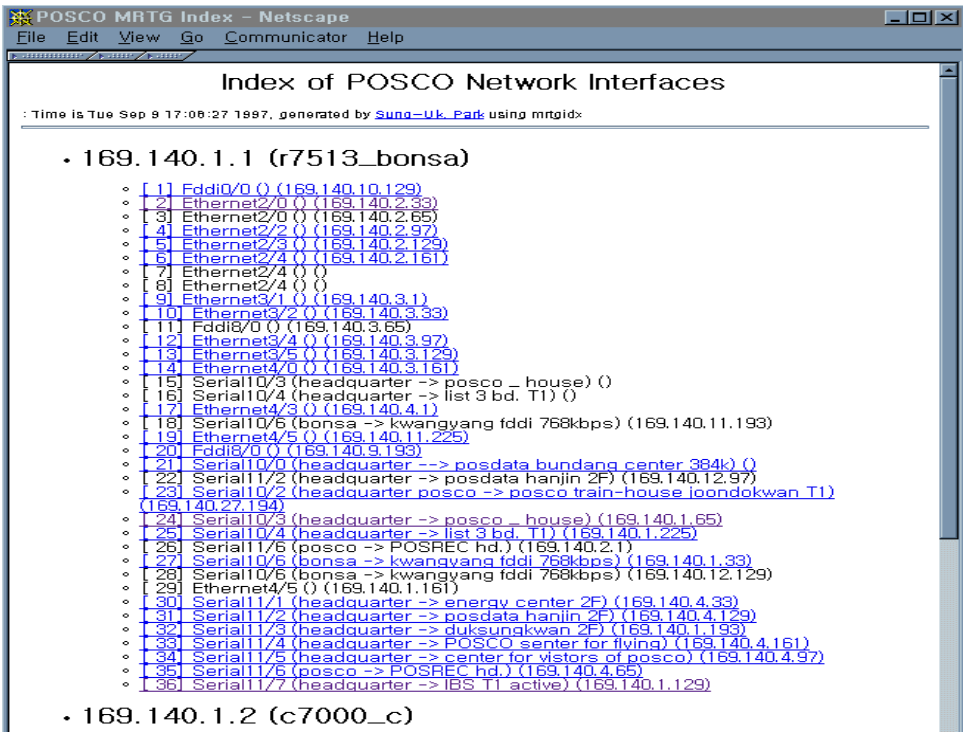

그림 15. Generated Index of Discovered Network Interface

Step (2): 여러가지 관리 기능들이 자동으로 수행되도록 시스템의 crontab 파일에 등록한다. <표 5>는 본 연구에서 사용된 crontab 파일의 일부분으로 MRTG+ 주 프로그램을 매 10 분 간격으로 수행되도록 하였다. 여기에 설정하는 시간 간격은 SNMP polling interval 과 관련이 있으므로 너무 짧은 interval 은 오히려 전체 네트웍에 불필요한 트래픽을 유발시키므로 관리자의 관리기준과 네트웍 환경에 따라 적절하게 조정할 필요가 있다. 즉, 중요한 네트웍과 그렇지 않은 네트웍을 구분하여 polling interval 을 적용한다. <표 5>에서 트래픽 분석을 위한 다른 프로그램들은 시스템의 CPU processing load 와 트래픽 load 를 줄이기 위해 임의로 시간배분을 하였다.

```
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
```

표 5. crontab 파일의 예

Steps (3) ~ (5): Step (1)에서 생성된 구성파일을 MRTG+의 입력으로 사용하여 agent 들로부터 필요한 트래픽 데이타를 SNMP operation 을 통하여 가져와서 MIR 에 특별한 log 파일 형태로 저장한다 [표 6]. 여기서 특징적인 것은 이 log 파일은 통합 알고리즘을 (consolidation algorithm) 사용하여 무한정 커지지 않는다.

	A	B		C	
Line 1	839513143	2405117528		3426790995	
	(1)	(2)	(3)	(4)	(5)
Line 2	839513143	4613	1143	4613	1143
Line 3	839513100	5104	1663	5104	1663
Line 4	839512800	4255	1168	4255	1168
Line 5	839512500	4058	1014	4058	1014

표 6. Log 파일의 예

하나의 인터페이스는 동일한 형태의 log 파일 2 개를 가지는데 하나는 현재 polling time 의 log 파일이고 다른 하나는 직전의 polling time 에 해당하는 log 파일

이다. 모든 log 파일은 <표 6>과 같이 2,533 줄의 text 파일로 이루어져 있고, 그 각각은 다음과 같은 의미를 가진다.

Line 1은 3부분으로 이루어져 있다.
- 'A'는 UNIX의 timestamp 로 1/1/1970을 '0000'으로 나타내고,
- 'B'는 'incoming bytes counter'값을 나타내고,
- 'C'는 'outgoing bytes counter'값을 나타낸다.

Line 2와 나머지는 5부분으로 이루어져 있다.
- (1)은 UNIX 'timestamp',
- (2)는 'average incoming transfer rate in bytes per second since the previous time'
- (3)은 'average outgoing transfer rate in bytes per second since the previous time'
- (4)는 'max-incoming transfer rate in bytes per second since the previous time'
- (5)는 'max-outgoing transfer rate in bytes per second since the previous time'

또한 log 파일 size의 증가를 막기위해 2,533 라인은 4개의 부분으로 나누어져 있다.
- (line 2 ~ 601)은 매 5분마다의 기록이고,
- (line 602 ~ 1201)은 30분마다의 평균으로 5분짜리 6개로 만들어지고,
- (line 1202 ~ 1801)은 120분마다의 평균으로 30분짜리 4개로 만들어지고,
- (line 1802 ~ 2533)은 24시간 평균으로 120분짜리 12개로 만들어진다.

이들 각각은 50시간, 300시간(12.5일), 1200시간(49일), 732일의 데이타를 저장할 수 있고, 화면으로는 4개의 그래프에 400개의 sample을 취해서 daily는 33.3시간의 데이타를, weekly는 8.33일의 데이타를, monthly는 33.33일의 데이타를 yearly는 366일의 데이타를 나타낸다.

포스코와 같은 대규모 네트웍을 가지고 있더라도 1년간의 네트웍 트래픽을 모니터링한 데이타가 차지하는 디스크 영역은 그리 크지가 않다. 예를 들면, 본 연구에서 포스코의 600여 인터페이스에 대해 입출력 트래픽과 에러율 및 기타 여러

가지 보고서 데이타 영역을 고려한 경우를 보더라도 하나의 인터페이스에 대해 아래와 같은 용량계산이 나온다.

- HTML 파일 1개 : 4KB
- GIF 파일 4개 : 12KB
- log 파일 1개 : 50KB
- old 파일 1개 : 50KB

} 총 116 KB

그리고 여기에 365일 동안의 보고서 영역을 고려하면,

- 보고서 파일 365개 : 4KB * 365 = 1.46MB

그러므로 총 소요 영역은

∴ 116KB * 1200 인터페이스 (트래픽 및 에러율) + 1.46MB **141 MB**

이 같은 사실은 본 Web 기반의 엔터프라이즈 네트웍 트래픽 모니터링 시스템을 운영하기 위한 시스템 요구사양이 얼마나 일반적인지를 보여 준다. 또 이를 적절히 이용 할 경우 수년간의 정보도 손쉽게 보관하고, 조회할 수가 있다.

Steps (6) ~ (8): MRTG+ 에 의해 수행된 **rateup** 프로그램이 log 파일들을 읽어서 해당 인터페이스에 대한 GIF 형식의 그래프 이미지를 생성하고, 여러가지 필요한 log 파일 및 HTML 파일 수정작업을 한다. <그림 16>은 Web browser를 통해 볼 수 있는 하나의 HTML 파일이다. 하나의 Web page 에 4 개의 그래프 (daily, weekly, monthly, yearly 그래프)를 볼 수 있다. 각각의 그래프에는 입력과 출력 트래픽에 대한 최대, 평균, 현재 트래픽 값들이 라인 용량에 대한 비율로 나타나므로 네트워 관리자는 한눈에 해당 네트워의 사용 추세를 파악할 수가 있다.

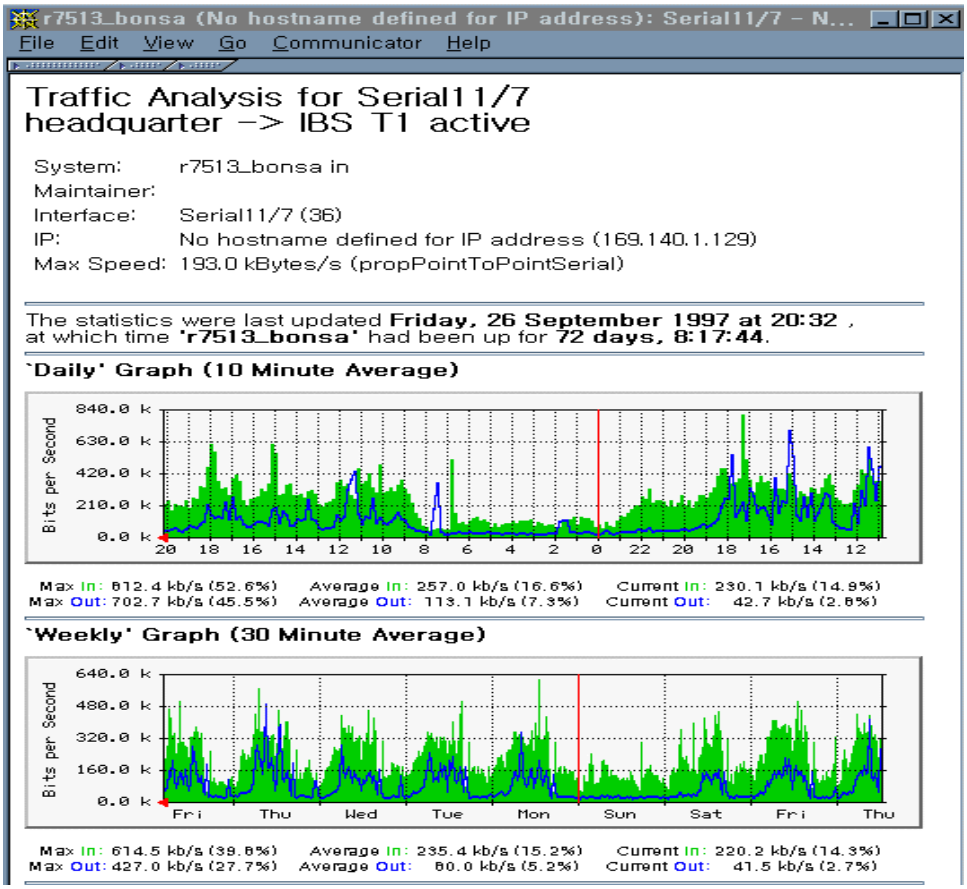

그림 16. 트래픽 모니터링 그래프

Step (9): 관리시스템을 확장한 부분으로 사용자의 편리성을 높이기 위해 GUI **모니터링 기능과** online threshold reporting **기능을 추가하였다** [**그림** 17, 18, 19].

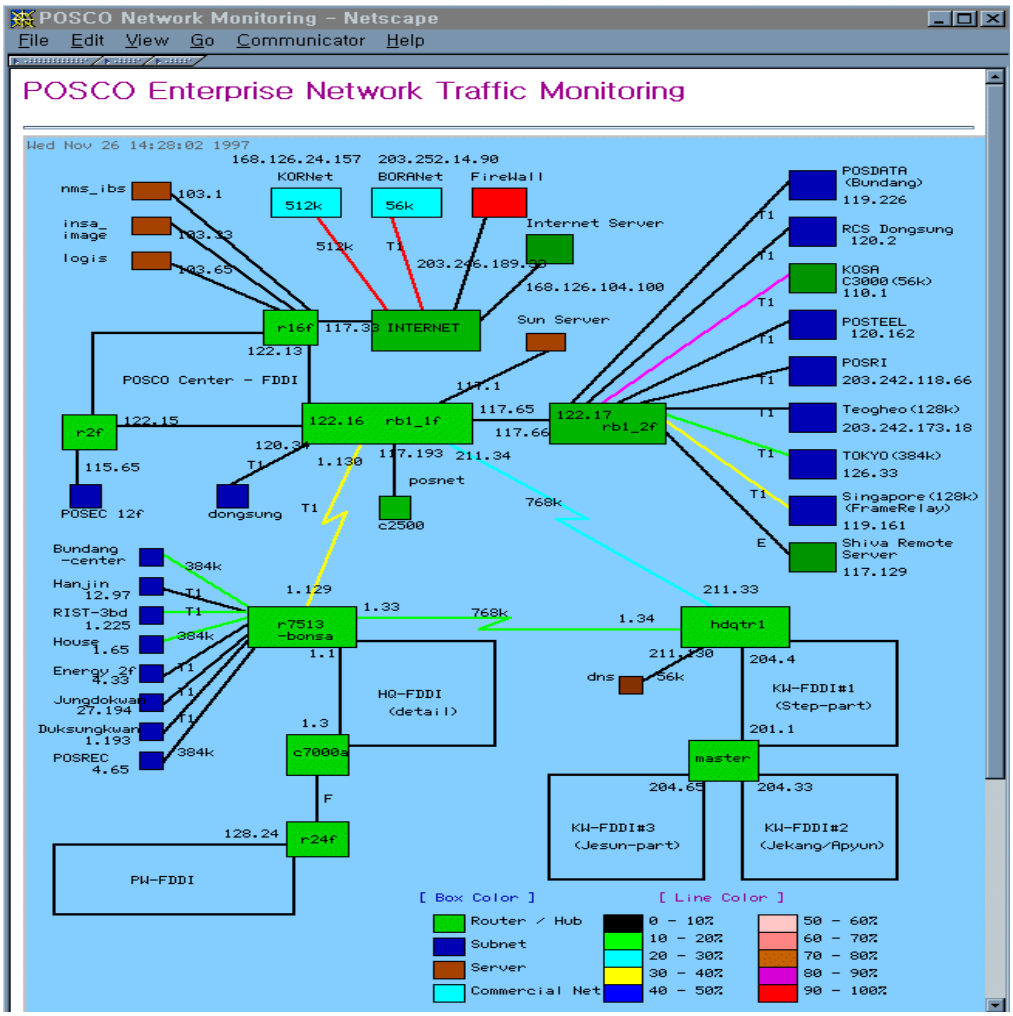

그림 17. POSCO 엔터프라이즈 네트워 구성

<그림 17>은 POSCO Enterprise network 을 한눈에 모니터링할 수 있도록 링크별 사용율에 따라 라인 색깔를 이용하여 표시하였다. 사용율이 높은 곳은 라인 색깔을 보고 쉽게 파악할 수 가 있다. 상세한 트래픽 내용을 보고자 할 경우는 해당 라인을 클릭하면 <그림 16>과 같은 자세한 트래픽 그래프가 나타난다. Dynamic

트래픽 모니터링을 제공하기 위해 HTML 의 META tag (즉 <META HTTP-EQUIV="Refresh" CONTENT=300>)를 이용한 'client-pull [30]' 기능을 사용하여 Web browser 가 매 10 분마다 최신 정보를 Web 서버로 부터 가져오도록 했다. 이렇게 함으로써 네트웍 관리자는 자기 PC 에 위의 Web page 를 띄워 놓기만 해도 포스코 전체의 네트웍 트래픽 현황을 거의 실시간으로 모니터링 할 수 가 있다.

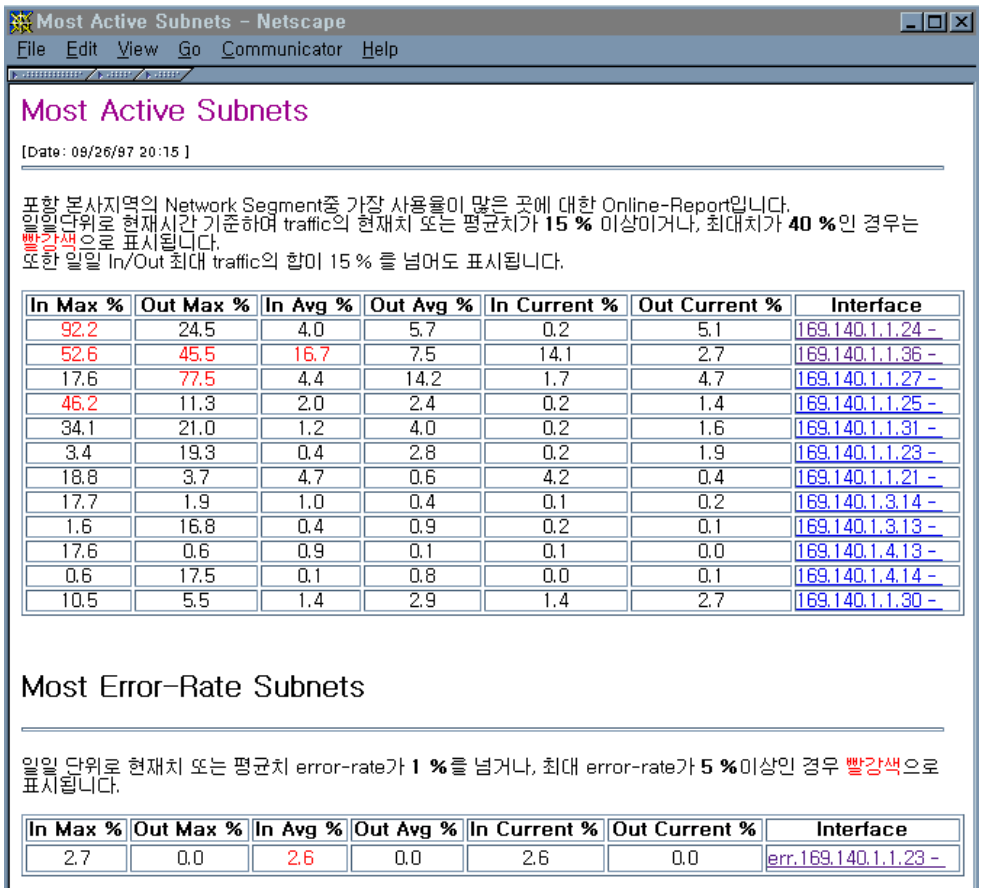

그림 18. Threshold Reports of Active Subnets

<그림 18>은 네트웍 인터페이스에 대한 사용율이나 에러율, 그리고 라우터의 CPU process load 등에 대한 threshold 값을 관리자가 설정한 후 이에 대한 결과를 online 으로 reporting 해 준다. 매 30 분마다 네트웍 segment 별 현재 네트웍 트래픽 상태

를 판단하기 위해 repository 의 log 파일을 검사하여 설정된 threshold 값을 over 하는 segment 에 대해서만 reporting 한다. 또한 이 결과는 날마다 백업되어 daily network report 로도 사용된다.

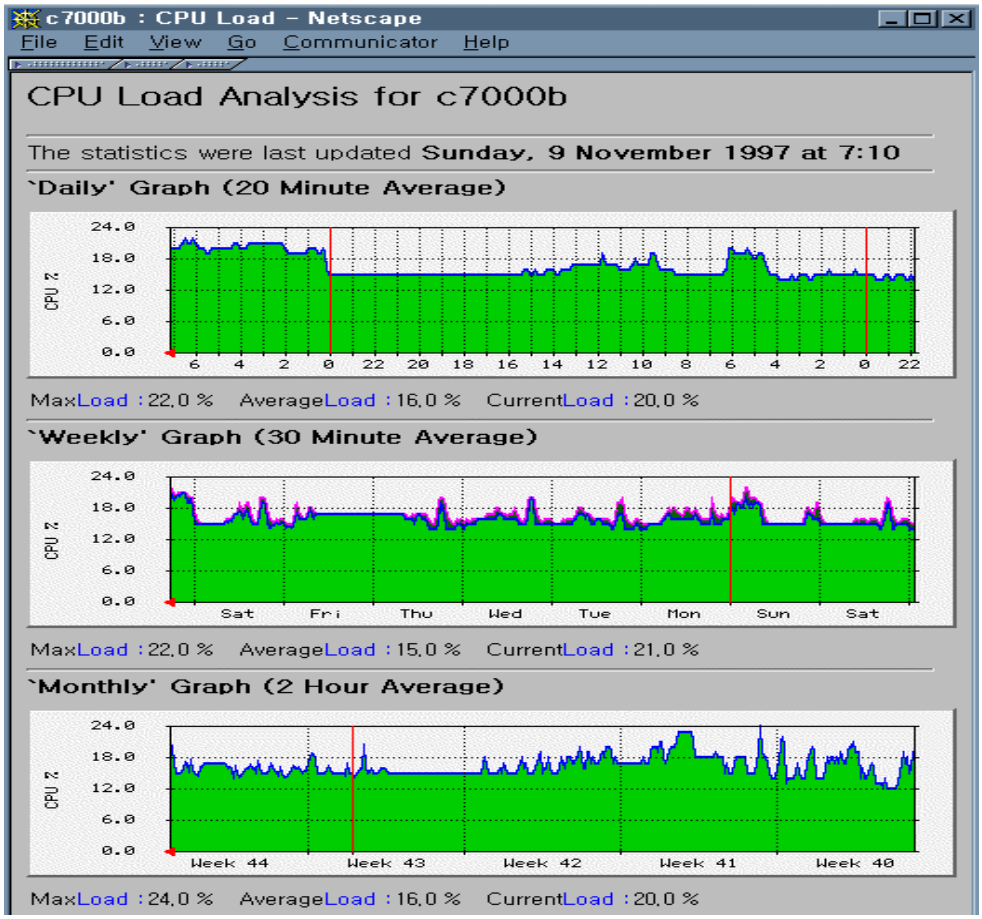

그림 19. Router CPU Load Monitoring

<그림 19>는 CISCO 라우터의 CPU load 에 대한 performance monitoring graph 로 이는 라우터 생산 업체에 따라 agent 가 제공하는 MIB 변수가 다르므로 업체에서 제공하는 MIB 파일을 보고 적용해야 한다. 위의 경우는 CISCO MIB 파일의 'avgBusy5' (즉 cumulative average of the CPU usage percentage over a 5 minute period)를 이용하여 나타내었다.

5. 시스템 적용 경험

여기서는 4 장에서 구현한 Web 기반의 엔터프라이즈 네트웍 트래픽 모니터링 시스템을 Pohang Iron and Steel Company (POSCO) 와 Pohang University of Science and Technology (POSTECH)에 약 6 개월간 적용하여 네트웍 트래픽을 모니터링 해 보고 느낀점에 대해 적어 보겠다.

POSTECH 은 하나의 FDDI backbone 을 중심으로 약 80 개의 Ethernet segment 가 연결되어 있고, 이들을 통하여 약 2 천대의 컴퓨터들이 (mainframe, group server, workstation, PC 등) 연결되어 있다. POSCO 는 서울-포항-광양의 지역 네트웍과 해외사무소 및 각 지역에 분산되어 있는 계열사로 구성된 global enterprise network 을 가진 기업이다. Network 은 크게 mainframe 중심의 non-TCP/IP 계열인 조업용 네트웍과 server 시스템 중심의 TCP/IP 계열인 사무용 네트웍으로 구성되어 있으며, 전체적인 Enterprise Network 은 <그림 17>과 같이 지역별로 FDDI 를 backbone 으로 하고 지역간 인터페이스는 serial link (T1, 768kbps)로 구성되어 있는 전형적인 Enterprise network 형태이다. 전체적으로 70 여개의 router 에 약 700 여개의 인터페이스로 구성되어 있다 [37].

본 논문에서 제시한 Web-based enterprise network traffic monitoring system 을 POSCO 에 약 5 개월간 ('97.6~'97.10) 적용하면서 느낀점은 이제껏 한번도 시행되지 않았던 POSCO 전체의 FDDI 에 대한 사용 율과 POSCO 네트웍의 사용추이를 한 눈에 파악할 수가 있었다는 것이다.

물론 본 논문에서 제시한 Web-based network traffic monitoring system 이 사용되기 이전에도 이미 POSCO 에는 SunNet Manager 나 NetView 같은 몇몇 상업용 NMS tool 들이 도입되어 있었다. 그러나 이러한 툴들은 극히 제한적인 시스템들의 관리에만 적용되거나 지역적인 네트웍 관리에만 적용되고 있었다. 그나마 이렇게 운영되던 NMS system 도 필요한 기능의 추가나 확장 시에는 제대로 활용되지 못하였고, 심지어는 툴 사용교육을 받은 담당자가 해당업무를 떠나게 되면 그나마 운영

되던 시스템도 흐지부지 되었다. 새로운 담당자가 NMS 툴에 대해서 제대로 숙지하기 위해선 또 많은 시간이 필요했다. 기존의 네트웍 관리 툴들은 이러한 사용 및 운영의 어려움으로 인하여 전문적인 네트웍 관리자 외에는 사용이 기피되어 왔고, 계속적인 네트웍 모니터링보다는 필요할 때 한번씩 보는 시스템으로 전락되어 왔다. 그러나 본 논문에서 제공한 Web-based enterprise network traffic monitoring system 을 사용하여 기존의 망 관리 툴이 하지 못했던 기업 전반에 걸친 Network Traffic Monitoring 을 지속적으로 추진하고, Reporting 하면서 전체적인 네트웍 사용현황 추이를 분석할 수 있었다. 더불어 Network capacity planning 시 의사결정 할 수 있는 자료로도 활용되어 네트웍 운영에 대한 적절한 대안을 간구함으로써 불필요한 의구심과 불안을 떨쳐 버릴 수 있었다.

본 시스템의 적용 초기에는 AFS server 가 backup 중일 때 공유 disk 에 대한 lock 이 정의한 polling interval 이상으로 길어져 Web server 시스템에서 한꺼번에 너무 많은 job 이 대기하여 시스템이 down 되는 현상도 있었다. 이를 해결하기 위해 공유 disk 가 아닌 local disk 를 사용하여 server 시스템이 backup 중이더라도 logging disk 에 대한 lock 은 발생하지 않게 함으로써 문제를 해결하였다.

또한 network traffic monitoring 을 위해 발행하는 SNMP traffic 이 얼마나 네트웍에 부하를 주는가에 대해서도 알아 보았다. FDDI 에 연결된 모든 router 의 agent 를 통하여 snmpInPkts 와 snmpOutPkts MIB variable 들의 합을 구한 후 2 로 나누어 보니, POSCO FDDI 같은 경우는 전체 bandwidth 의 약 0.6% 정도를 차지했다. 현재는 10 분의 polling interval 을 가지고 작업했으므로 interval 을 널린다면 이 값은 더 낮아 질것이다.

대부분의 작업이 실제로 네트웍을 관리하는 관리자들과의 협조를 통하여 그들 스스로 적용하여 운영하게 함으로서 사용의 편리성에 대해 서로 공감하게 되었고, 기존에 도입된 망 관리 툴과의 연계 활용성 등도 깊이 고려하여 좀더 효율적인 네트웍 관리에 도움을 주고 있다.

6. 결론 및 향후 과제

본 논문에서는 최근에 가장 보편적인 data access 방법인 Web browser를 이용한 인터넷 기술과 기존의 관리시스템 (SNMP agent)을 이용하여 배우기 쉽고, 사용하기 쉽고, 그러나 기능은 강력한 Web 기반의 엔터프라이즈 네트웍 트래픽 모니터링 시스템에 대해 살펴 보았다.

구현을 위해서는 공개된 툴을 사용하고, 여기에 몇 가지 필요한 기능들 (dynamic traffic map, threshold report, security, router CPU load 및 traffic analysis guideline 등)을 추가로 제공함으로써 상업용 툴에 버금가는 편리성을 추구하였다. 특히 Web 기반의 네트웍 모니터링은 기존 NMS 툴들의 사용하고 배우기 어렵고, 기술 축적이 힘든 점들을 쉽게 극복할 수 있도록 해 주며, Perl 기반의 시스템은 구현이 쉬울 뿐만 아니라 필요한 기능을 사용자 스스로 쉽게 확장할 수 있다는 장점이 있다.

요즘과 같은 Network computing 시대에서 기업의 Network monitoring 은 전산자원의 효율적인 사용을 위한 첫걸음이다. 이를 위해서는 본 논문에서 제시한 시스템과 같은 저 비용 고 효율의 관리 시스템을 사용하여 네트웍 관리자 한 사람에만 의존하던 폐쇄적인 네트웍 관리가 open 화 되어야 한다. 그러나, 아직도 네트웍 관리분야는 기업에 있어서 많은 부분이 감춰져 있으며, 이러한 부분들은 여전히 몇몇 운영자의 know-how 에 의존하고 있는데 Web 을 통한 지속적인 모니터링과 그 분석 결과에 대한 기업 내 공유체제를 통해 점차 해소 되어지길 바란다.

본 논문에서는 현재는 SNMP 에 대한 부분만 적용하였으나, 향후 DMI, CMIP 과 같은 management protocol 에도 확장하기 위한 방안도 연구되어 적용되어야 한다. 그리고, 좀더 다양한 enterprise network 의 사용현황을 분석하기 위해서는 SNMP MIB variable 에 대한 연구도 강화되어야 한다. 특히 RMON 을 이용하여 기존 poll-select 방식의 SNMP monitoring 이 WAN 을 이용한 monitoring 시 야기될 수 있는 문제점을 해결하고, 지금까지는 network monitoring 의 관점에서만 적용하였으나, system 에 대한 monitoring 부문에까지 확대 적용되어야 한다. 이를 위해 system

agent 와 이들이 지원하는 MIB 에 대한 연구도 계속 되어야 하고, 기존에 도입되어 있거나 사용중인 NMS tool 들의 data 를 활용할 수 있는 방법들에 대한 연구도 진행되어야 한다.

참고 문헌

[1] HP, “HP 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. Berbers-Lee, R. Cailliau, J. Groff, and B. Pollermann, “World-Wide Web: The Information Universe”, *Electronic Networking*, Vol. 1, No. 2, Spring 1992.

[6] K. Arnold and J. Gosling, *The Java Programming Language*, Addison-Wesley, 1996.

[7] J. Case, M. Fedor, M. Schoffstall and C. Davin, "The Simple Network Management Protocol (SNMP)", RFC 1157, May 1990.

[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] Master Assignment of Twente University, "Web-based Management", ftp://ftp.cs.utwente.nl/pub/src/snmp/UT-THESIS/Kasteleijn.ps.

[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. Yen, J. S. Kim, J. T. Park and J. W. Beak, “Web-based Intranet Services and Network Management”, *IEEE Communications Magazine*, Vol. 35, No. 10, October 1997, pp. 100-110.

[13] F. Barillaud, Luca Deri, and Metin Feridun, “Network Management using Internet Technologies”, *Proc. of the IEEE/IFIP International Symposium on Integrated Network Management*, San Diego CA, May 1997, pp. 61-70.

[14] M. Maston, “Using the World Wide Web and Java for Network Service Management”, *Proc. of the IEEE/IFIP International Symposium on Integrated Network Management*, San Diego CA, May 1997, pp. 71-84.

[15] M. Peercy, B. Reed and E. Robinson, “Distributed Systems Management on the Web”, *Proc. of the IEEE/IFIP International Symposium on Integrated Network Management*,

San Diego CA, May 1997, pp. 85-95.

[16] Luca Deri, "Surfin' Network Resources across the Web", *Proc. of the IEEE Workshop on Systems Management*, Toronto, Canada. June 1996, pp. 158-167.

[17] WBEM Consortium, WBEM homepage. http://wbem.freerange.com/, July 1996.

[18] WBEM Consortium, "Web-Based Enterprise Management Proposal. HyperMedia Management Protocol Overview", Revision 0.04. July 1996.

[19] WBEM Consortium, "Web-Based Enterprise Management Proposal. HyperMedia Management Schema Overview", Revision 0.04. July 1996.

[20] Sun Microsystems Inc., "Java Management API Architecture", June 1996.

[21] Sun Microsystems Inc., *Java Management Programmer's Guide*, Developer's Release, June 1996.

[22] Sun Microsystems. "Java Remote Method Invocation Specification", Draft, Revision 1.1, November 1996.

[23] D. Ragged, "Hyper Text Markup Language Specification Version 3.0 (HTML)", Internet Draft, April 1995.

[24] T. Berbers-Lee, R. Fielding and H. Nielsen, "hypertext Transfer Protocol - HTTP/1.0", Internet Draft, October 1995.

[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", pp. 61-72, McGraw-Hill, 1995.

[29] A. Leinwand and K.F. Corney, "Network Management: A Practical Perspective", 2nd edition, pp. 207-208, Addison-Wesley, 1996.

[30] Dynamic HTML, "client-pull", http://devlab.cs.dartmouth.edu/cowen/html/les27.html.

[31] OMG. The Common Object Request Broker: Architecture and Specification Revision 2.0. OMG, July 1995. OMG TC Document.

[32] SNMP Research, "The DR-Web Manager", http://www.snmp.com/drwebmgr.html.

[33] Wipro, CyberManage, "A White paper on Web-Based Management", http://cybermanage.wipro.com/cmwhite.html.

[34] OSI, Information Technology - Open Systems Interconnection - Systems Management

Overview, International Organization for Standardization, June 1991.

[35] Manfred R. Siegl, "What is Network Management?", Computing Services, University of Technology, Vienna.

[36] W. Stallings, *SNMP, SNMPv2 and RMON*, 2nd Edition, Addison-Wesley, 1996.

[37] POSTECH, *POSCO 전산자원 효율화 방안*, 정보통신연구소, 1997.

[38] Heinz-Gerd Hegering and S. Abeck, *Integrated Network and System Management*, Addison-Wesley, 1994.

[39] IONA, OrbixWeb, IONA Technologies Ltd., http://www.iona.com/Orbix/OrbixWeb/index.html , December 1996. Release 2.0.

[40] T. H. Yun, J. Y. Kong, and J. W. Hong, "Object-oriented modeling of distributed multimedia services", *Proc. of the IEEE International Conference on Communications*, Montreal, Canada, June 1997. pp. 777-781.

[41] B. Welch, *Practical Programming in Tcl and Tk*, Prentice Hall, 1995.

[42] A. Leinwand and K.F. Corney, "Network Management: A Practical Perspective", 2nd edition, pp. 145-192, Addison-Wesley, 1996.

[43] Bay Networks, "A White Paper on Web-Based Network Management", http://www.baynetworks.com/Products/Papers/webbased.html

[44] OSIMIS, http://www.cs.ucl.ac.uk/research/osimis/

[45] DSET, http://www.dset.com

[46] IBM, TMN agent toolkit, http://issc2.boulder.ibm.com/telmedia/tmnbase.htm