

네트워크 자원관리 시스템의 설계 및 구현

김연성*, 한판암
경남대학교 정보통신공학부

e-mail:{zetto*, pahan}@zeus.kyungnam.ac.kr

Design and Implementation of Network Resource Management System

Youn-Sung Kim, Pan-Am Han
Division of Information & Communication, Kyungnam University

요 약

인터넷 인구의 증가로 인하여 보다 강력한 컴퓨팅 파워를 위해 네트워크로 연결되어 모든 것을 공유하고자 하는 노력이 진행되었다. 이를 관리하는 기존의 시스템들은 시스템관리나 네트워크 장비, 회선에 대한 관리 등이 따로 수행되고 있는 실정이다. 이에 따른 복잡한 네트워크 자원들을 관리하기 위해서 플랫폼 독립적 통신을 위한 JAVA RMI와 TCP/IP, SNMP를 이용한 네트워크 자원관리 시스템을 설계 및 구현하였다.

1. 서론

최근 몇 년간의 지속적인 인터넷 인구의 증가로 인하여 보다 강력한 컴퓨팅 파워를 위하여 네트워크로 연결될 수 있는 모든 것을 공유하고자 하는 노력이 진행되었다. 이에 따른 복잡한 네트워킹 환경 속에서 효율적인 네트워크의 자원을 관리하기 위한 필요성이 대두 되었고 여러 종류의 관리 방법론이 제기된다. 현재 여러 종류의 네트워크 관리 시스템이 존재하고 있다. 하지만 이들은 시스템 관리나 네트워크 장비, 회선에 대한 관리 등이 따로 수행되고 있는 실정이다.

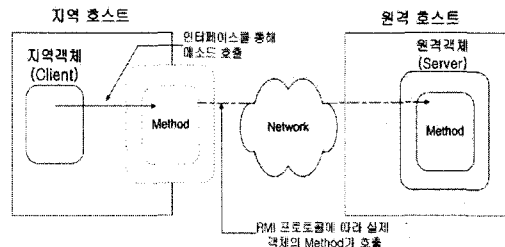
이상과 같은 문제점을 해결하기 위하여 본 논문에서는 여러 자원들을 통합적으로 관리하기 위한 NRMS(Network Resource Management System)를 설계 및 구현했다[2][3][9].

2. 관련연구

2.1 RMI

RMI(Remote Method Invocation)는 여러

JVM(Java Virtual Machine)에 상주하는 객체간의 통신을 가능하게 한다. 그리고 애플리케이션이 원격으로 위치하고 있는 객체의 메소드를 호출할 수 있게 해줄 뿐만아니라 공유자원을 이용할 수 있다. 또한 시스템에 대한 부하 처리를 할 수 있게 해준다. 한편 필요시 새로운 클래스의 동적 적재(loading)을 허용하면서, 이전까지 외부 가상머신이 다루지 않았던 객체조차도 메소드로 전달할 수 있다. [그림 1]은 Java의 원격 메소드 호출을 나타내고 있다 [4][5].



[그림 1] Java의 원격 메소드 호출

2.2 SNMP

SNMP(Simple Network Managment Protocol)표준은 전산망 구성요소에 관한 관리 정보가 논리적으로 원격지 사용자들에 의해 조사되거나 변경될 수 있게 지원하는 간단한 프로토콜을 정의한다. 특히, 관리 정보 베이스(MIB)와 함께 관리정보의 구조(SMI)를 기술하는 관련 표준들과 같이 본 표준은 TCP/IP를 기반으로 하는 연결망들(internets)을 관리하기 위해 간단하고 운영 가능한 구조와 시스템을 제공한다.

SNMP의 구조적인 모델은 전산망 관리 스테이션들과 전산망 요소들로 이루어진다. 전산망 관리 스테이션들은 전산망 요소들을 감시하고 통제하는 관리 응용들을 수행한다. 전산망 요소들은 호스트, 게이트웨이, 단말기 서버 등과 같은 장비이다. 이 장비에는 전산망 관리 스테이션들로부터 요청된 전산망 관리 기능들을 수행하는데 책임이 있는 관리 수행자가 있다.

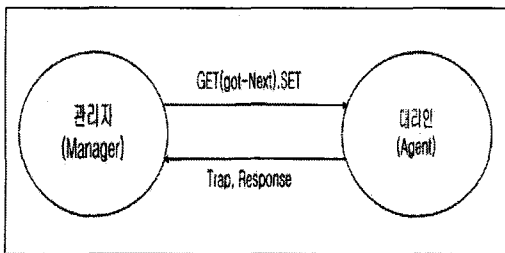
SNMP는 기본적으로 3가지 작동으로 네트워크를 관리한다[7].

*GET : 관리자가 대리인에 있는 객체의 값을 가져온다.

*SET : 관리자가 대리인에 있는 객체의 값을 변경한다.

*TRAP : 대리인의 특정 상황 발생을 관리자에게 알린다.

[그림 2]는 SNMP는 관리자의 요청에 대리인이 응답하는 것이 기본이고 대리인은 특수한 상황이 발생한 경우에 관리자에게 발생을 알리고 필요한 정보를 함께 보낸다.



[그림 2] SNMP 기본통신

2.3 MIB

MIB(Management Information Base)는 SNMP에서 관리하는 정보의 데이터베이스와 같은 것으로 (관리 항목의 정의 파일 및 표 등이 있는 것), 어떤

항목에 대하여 요청하면 어떤 결과가 되돌아올지를 각각 정해 놓고 있다. MIB에는 다음의 세 종류가 있다[8].

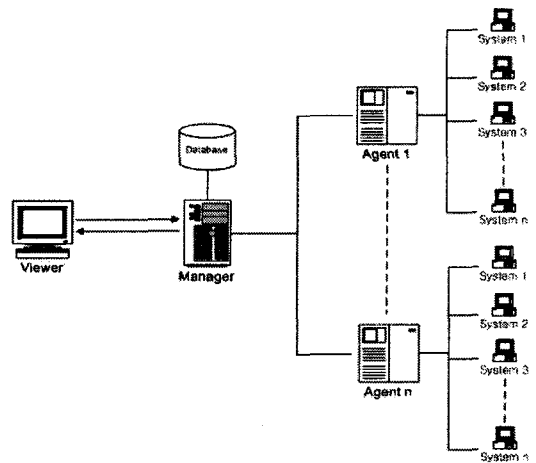
1) MIB-1 : MIB-1은 관리 정보 베이스로 원래는 MIB라고 불렀으나 MIB의 확장판인 MIB-2가 발표됨에 따라 MIB-2와 구별하기 위해서 MIB-1이라 불리게 되었다. MIB -1은 네트워크 관리에 필요한 최소한의 관리대상을 정의하고 있는데 그 Object는 114개이다.

2) MIB-2 : MIB-2는 MIB-1의 확장판으로 MIB-1의 모든 Object들을 포함하여 총 171개의 Object를 포함하고 있다. 현재 시중에 제공되고 있는 대부분의 제품은 MIB-2를 지원하고 있다.

3) 확장 MIB : MIB-1, MIB-2에서는 규정되어 있지 않으나, Vendor가 가지고 있는 독자적 기능을 SNMP에서 관리할 수 있도록 정의한 관리 항목이다.

3. 시스템 설계 및 구현

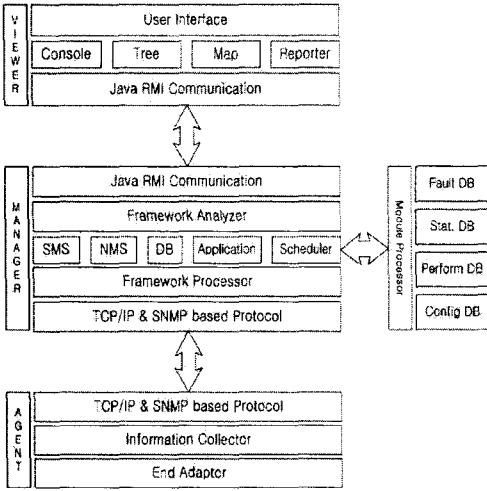
본 연구에서 구현하려는 네트워크 자원관리를 위한 시스템의 구성은 [그림 3]과 같다.



[그림 3] NRMS 의 구조도

시스템은 사용자별 접근 제어 및 플랫폼 독립적 관리를 할수 있는 JAVA RMI 기반의 Viewer, 각 기능별 통합 관리와 운영 및 전체 이벤트의 통합과 통보를 할수 있는 Manager, 그리고 직접적인 시스템별 기초자료를 수집하는 Agent로 구성되어 있다. 이들은 TCP/IP 망을 이용한 정보수집 및 관리를 한다.

3.1 NRMS 모듈구조

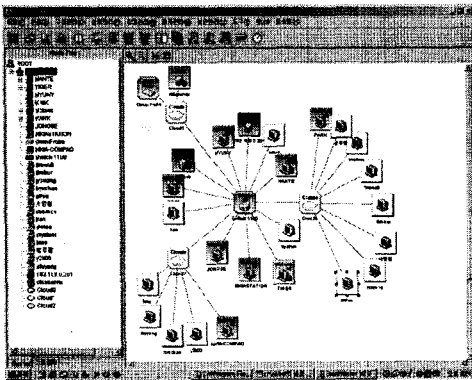


[그림 4] NRMS의 모듈 구성도

본 연구에서 구현한 NRMS는 Viewer와 Manager간의 통신은 JAVA RMI를 이용한 플랫폼 독립적인 환경에서 모니터링 및 관리 할 수 있다. 또한 Manager와 Agent간의 통신은 산업표준인 SNMP와 TCP/IP 프로토콜을 병행함으로써 성능과 유연성을 향상시킬 수 있다. [그림 4]는 NRMS의 모듈 구성이다.

3.2 Viewer

[그림 5]는 Viewer의 GUI 환경으로, Manager에서 생성되는 각종 정보들을 그래픽을 이용한 분산된 네트워크 자원들을 콘솔 및 맵을 이용한 장애 발생 시 직관적 판단이 가능하다.



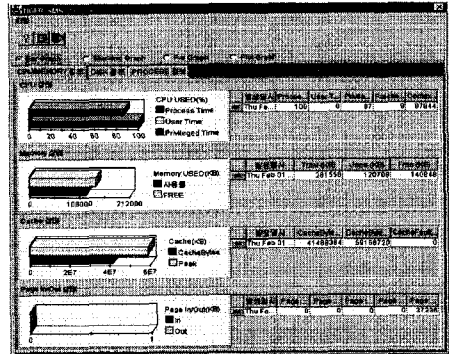
[그림 5] 노드들의 구성 화면

3.3 Manager

관리자가 Viewer의 GUI를 통해서 Manager의 Framework Processor에게 정보를 요청하게 되면, Framework Processor는 SNMP를 통하여 MIB를 참조하여 Agent에게 요청한다. Agent에서 수집한 정보를 Framework Processor에서 전처리를 수행한 다음에 각 해당 모듈에 전달하게 된다. 각 모듈에서 해당 정보를 가공한 후 모듈 프로세서가 각 데이터베이스에 수집된 정보를 기록한다. 그리고 Framework Analyzer가 모듈에서 가공된 정보를 Viewer에게 전달하게된다. 이처럼 Manager에서는 시스템, 네트워크, 데이터베이스, 응용프로그램 통합 관리, 하나의 공통 기반 구조와 정보공유에 따른 공동정책 위주의 관리를 하게 된다.

· SMS

SMS(Server Management System)는 시스템 상태(CPU, Memory, Disk, Process) 및 자원 사용량, 성능 정보를 실시간 관리를 한다. [그림 6]은 CPU와 메모리의 사용률을 Viewer의 GUI를 통해서 그래프 형식으로 보여주고 있다.

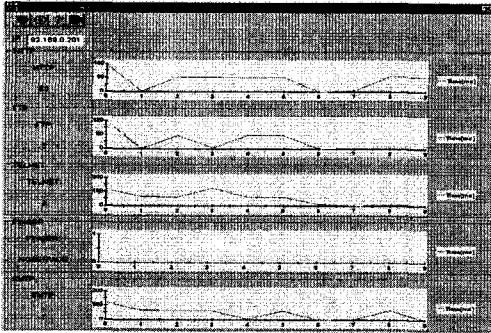


[그림 6] 자원사용정보 화면

· NMS

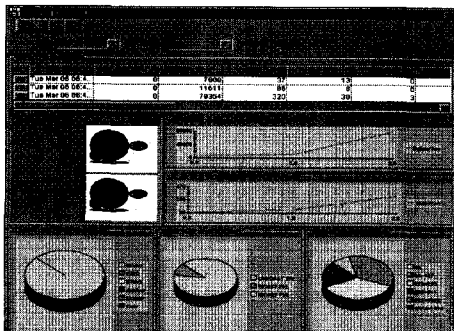
NMS(Network Management System)는 전반에 걸친 정보를 수집 관리하는데 그 목적이 있다. 전체 네트워크의 Utilization을 분석, 특정 노드간 트래픽 정보 제공, 시스템 상태 및 자원 사용량, 성능 정보를 실시간 관리를 한다. SNMP Agent에 정보를 의뢰함으로써 Device들을 감시 제어한다. SNMP Agent는 NMS의 요구에 응답하고 네트워크상의 관리 대상 장비에 존재하는 소프트웨어이다. SNMP Agent에는 장비에 관한 정보인 MIB(Routing Table

Counter, status indication 등)가 있다. 이들은 SNMP Agent에 대한 NMS의 Poll과 Query에 대한 응답으로 NMS에 보내지고 이들은 다시 데이터베이스에 저장된다.



[그림 7] 노드의 서비스 응답시간 화면

[그림 7]은 선택한 노드를 대상으로 HTTP, FTP, TELNET, FINGER, SMTP의 다섯 종류의 프로토콜에 대한 정보를 나타낸다. 사용자는 트리 선택 혹은 직접 입력으로 노드의 IP를 선택한 후 시작 버튼을 눌러 주기적으로 정보를 가져온다. 정지 버튼으로 서비스를 멈출 수 있으며 닫기 버튼을 눌러 화면을 닫을 수 있다. [그림 8]은 RMON(remote network monitoring) Statistics과 같은 네트워크 트래픽을 모니터링하고 있다.



[그림 8] 네트워크 트래픽 모니터링 화면

3.4 Agent

Manager를 통한 Viewer의 요청에 응답을 하기 위해서 MIB를 이용하게 된다. Information Collector가 각 노드들의 정보를 End Agent를 통하여 실시간으로 수집하여 Manager에 전달 함으로써 항상 최근의 MIB 정보를 유지하게 된다.

4. 결론 및 향후 과제

본 논문에서는 JAVA RMI와 TCP/IP, SNMP를 이용하여 독립적인 네트워크 자원의 관리가 가능한 네트워크 자원 관리 시스템(NRMS)을 설계 및 구현하였다. 플랫폼 독립적인 JAVA의 RMI를 이용하여 Viewer의 타 플랫폼으로 이식성을 높였고, 직관적인 GUI를 이용하여 전문적인 지식을 갖추지 못한 신입 관리자라 하더라도 손쉽게 네트워크 자원을 관리할 수 있는 시스템이다. 향후 연구 과제로는 Viewer와 Manager의 통신에서 SOAP(Simple Object Access Protocol)을 이용한 브라우저에서 접근시 안정적인 환경이 되도록 추가 구현해서 좀 더 실용적인 네트워크 자원관리 시스템의 구축을 실현하는 것이다.

참고문헌

- [1] 박진홍, 장국렬, 박순영, "지능형 원격 감시시스템의 개발에 관한 연구", 목포대학교, 1998.
- [2] 장훈 역(1998), 시스템 관리의 핵심, 한빛미디어.
[원전: AEleen Frish (1996). Essential System Administration. O'Reilly.]
- [3] 한국네트워크연구조합(1999), 학내 전산망 구축 지침서, 진한도서.
- [4] "Inside Secret Java 2 Developer's Handbook", Philip Heller & Simon Roberts, pp. 648-693
- [5] David Flanagan, "Java in a Nutshell, A Desktop Quick Reference for Java Programmers", O'Reilly & Associates, Inc. 1996.
- [6] Alfons Kemper, Guido Moerkotte, Object-oriented Database Management : Applications in Engineering and Computer Science, Prentice Hall, 1997.
- [7] RFC1157, "Simple Network Management Protocol(SNMP)", May 1990.
- [8] RFC1213, "Management Information Base(MIB-II)", March 1991.
- [9] Kohei Ohta, Nei Kato, Glenn Mansfield, Yoshiaki, Configuring a Network Management System for Efficient Operation, Nemoto International Journal of Network Management Vol.6, Issue:2, pp. 108-118. March/April, 1996.