

계층적 관리 구조를 갖는 정보 자원 관리 시스템의 설계 및 구현

이 원 혁[†] · 안 성 진^{††} · 정 진 욱^{†††}

요 약

본 논문에서는 PC나 라우터, 서버 장비 등 모든 정보자원 시스템을 통합 관리하기 위한 시스템을 제안한다. 이 시스템은 PC 및 라우터의 회선에 대한 성능 분석 항목을 정의하고 분석하며, 서버 장비에 대한 자산 관리를 수행한다. 관리자는 분석된 데이터를 바탕으로 현재 시스템이나 회선에 대한 상태를 파악하기 위한 자료로 활용할 수 있다. 시스템의 구조는 차후 관리 도메인의 증설이 쉽고, 효율적으로 관리하기 위해 3계층으로 구분하여 관리하고자 한다.

Design and Implementation of Information Resource Management System with Hierarchical Management Architecture

Won Hyuk Lee[†] · Seong Jin Ahn^{††} · Jin Wook Chung^{†††}

ABSTRACT

In this paper, we propose the integrated management system for computing resource such as personal computers, the interface of router, and server equipment. IRMS defines and analyzes the performance factors for system such as pc and router, also it manages assets for server equipment. Manager can make use of data for understanding conditions whether system is normal or not. We suggest a hierarchical integrated management architecture for resolving problems of scalability and managing efficiently in large domain networks.

키워드 : 통합 관리(Integrated Management), 성능 분석항목(performance factor), 자산 관리(assets management), 시스템(system), 네트워크(network), 계층적 관리(hierarchical management), 관리도메인(managed domain)

1. 서 론

90년대 중반이후 인터넷 사용 인구의 폭발적인 증가로 인해 급격하게 네트워크의 증설이 이루어 졌고, 개인의 생활이나 업무에서 컴퓨터와 밀접한 관계를 갖게 되었다. 대부분의 회사는 자체 네트워크를 구축하여 운영하고 있고, 개인 사용자들도 회선을 제공, 대여해 주는 회사를 통하여 인터넷에 쉽게 접근하고 있는 실정이다[1].

이에 따라 복잡한 네트워크와 정보자원 시스템을 관리하기 위한 필요성이 대두되고 있다. 정보자원 시스템이란 연산을 수행하고 통신을 수행할 수 있는 컴퓨터나 라우터와 같은 Computing Resource를 의미하는 것이다. 현재 네트워크의 구성은 여러 가지 종류의 정보자원 시스템으로 구성되어 있다. 예를 들면, 개인이 사용하는 일반 컴퓨터와 네

트워크간의 연결을 제공하는 라우터, 그리고 서버 장비나 허브, 스위치와 같은 네트워크 장비들이 존재한다[7].

네트워크의 규모가 증대되고 피관리 자원들이 증가함에 따라 이를 효율적으로 관리하기 위한 방법론이 제기된다. 현재 여러 종류의 네트워크 관리 시스템들이 존재하고 있지만 이는 특정 네트워크 장비만을 관리하며, 시스템 관리나 네트워크 장비, 회선에 대한 관리등이 서로 따로 수행되고 있는 실정이다[4-7, 12-15].

이에 본 논문에서는 일반 PC나 서버 장비와 같은 시스템은 물론 라우터 등의 네트워크 장비도 통합 관리하기 위한 IRMS(Information Resource Management System)에 대한 모델을 제시하고 이를 구현하고자 한다[12, 13].

본 논문의 목표는 PC 기반의 시스템 및 네트워크를 관리하는 통합 관리 시스템 개발을 통해 PC와 통신 장비에 대한 관리를 통합적으로 관리하여 네트워크 상의 모든 Computing Resource들을 관리하고자 하는 데 있다. 이와 더불어 인터넷 통신 장비인 라우터, 허브 등에 관련된 성능분석

† 준 회원 : 성균관대학교 대학원 전기전자 및 컴퓨터공학부
 †† 중신회원 : 성균관대학교 컴퓨터교육과 교수
 ††† 중신회원 : 성균관대학교 전기전자 및 컴퓨터공학부 교수
 논문접수 : 2001년 12월 11일, 심사완료 : 2002년 5월 10일

항목들을 정의하고 이를 기반으로 데이터를 측정하고 실측 데이터의 의미를 분석하여 통합적인 시스템 및 네트워크 관리를 목적으로 한다.

또한 기존의 관리 시스템의 구조를 살펴보면 중앙 집중적인 시스템이 많이 사용되고 있는데, 이는 관리 네트워크의 규모가 커지게 되면 부하가 많이 발생하여 효율성이 떨어지고 확장성에서 많은 제약을 가지게 된다. 따라서 이러한 제약점을 보완하고 관리 시스템의 효율성을 높이기 위하여 계층적인 관리 구조로 설계를 하고자 한다[4, 5, 7].

아울러 라우터와 같은 장비의 관리는 기존에 많이 사용되는 관리 프로토콜인 SNMP(Simple Network Management Protocol)를 이용한다. 라우터 관리에 있어서 앞으로 정의하게 될 여러가지 성능 분석 항목(Factor)들에 대하여 필요한 MIB항목들을 자체 정의하고 이에 대한 값을 폴링하고 분석하기 위하여 SNMP를 활용하게 될 것이다[2, 3].

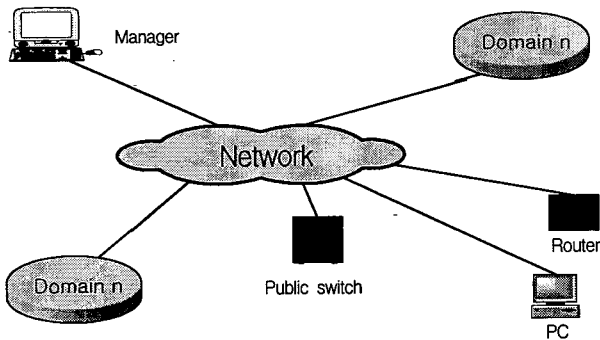
본 논문은 크게 6개의 절로 구성되어 있다. 2절에서는 기존에 연구되었던 네트워크 관리 구조의 형태에 대하여 정리하였고, 3절에서는 IRMS의 전체 구조에 대한 설계와 더불어 각 계층별 구조를 정의하였다. 4절에서는 자원 관리대행자의 내부 흐름도와 관리자의 상태 천이도, 통신 메시지 형식을 정의하였다. 또한 5절 실험 및 고찰에서는 실험 환경과 테스트 시나리오를 구성하여 본 논문의 설계에 대한 구현을 수행하였다. 마지막으로 6절에서 결론을 기술하였다.

2. 네트워크 관리 구조의 형태

본 시스템에 대한 정의를 하기 전에 기존에 많이 쓰이는 네트워크 관리 구조의 세 가지 형태에 대하여 살펴보도록 하겠다. 현재 구성되고 있는 관리 시스템은 이 세 가지 유형의 구조를 지니게 된다[4].

2.1 집중형 구조(Centralized Architecture)

집중형 구조는 하나의 관리자가 모든 관리 네트워크를 관리하고 책임지는 일반적인 형태이다. 관리자는 모든 에이전트와의 통신을 처리하고 이에 대한 제어 및 정보에 대한 처리를 수행한다. 하나의 관리자가 요구되므로 구현에 있어서

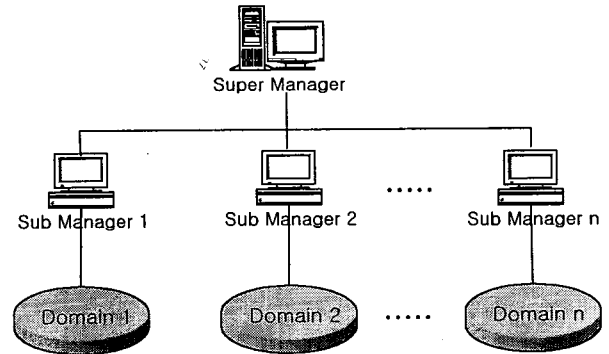


(그림 1) 집중형 구조

간단하고 여러 가지 정보에 대한 정의가 간단한 편이다. 또한 하나의 관리자가 모든 에이전트에 접근할 수 있으므로 관리의 편리성을 얻을 수 있다. 그러나 관리 네트워크의 증가는 관리자에 과부하를 주게 되고 결국 효율성도 떨어지는 단점이 있다. 다음 (그림 1)은 집중형 구조도를 나타낸 것이다.

2.2 계층형 구조 (Hierarchical Architecture)

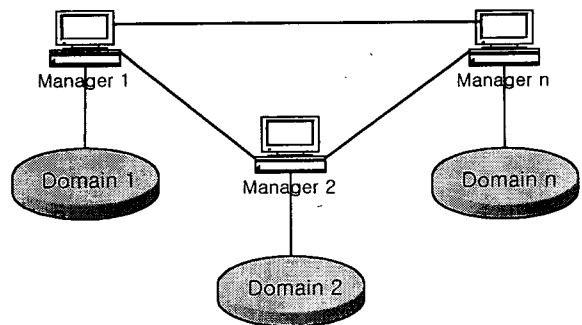
계층형 구조는 하나의 주 관리자(Super Manager)와 여러 개의 보조 관리자(Sub Manager)로 구성된다. 보조 관리자는 집중형 구조의 매니저와 같이 자신이 관리하는 도메인에서는 모든 관리 책임을 갖게 된다. 또한 주 관리자는 각 도메인마다 설치되어 있는 보조 관리자만을 관리함으로써 전체적인 네트워크 영역을 관리할 수 있게 된다. 관리 네트워크의 추가 시에는 하나의 보조 관리자를 추가함으로써 전체적인 관리가 손쉽게 이루어진다. 따라서 계층형 구조는 관리 네트워크의 확장에 대하여 유연성을 가지며 손쉽게 관리 영역을 넓힐 수 있다. (그림 2)는 계층형 구조를 나타낸 그림이다.



(그림 2) 계층형 구조

2.3 분산형 구조 (Distributed Architecture)

분산형 구조는 위에서 살펴본 집중형 구조와 계층형 구조를 혼합하여 구성한 형태이다. 집형 구조의 관리자나 계층형 구조의 주 관리자와 보조 관리자의 개념을 넘어서서



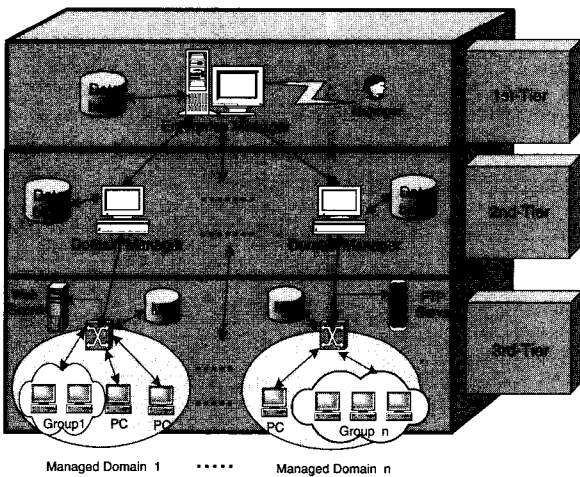
(그림 3) 분산형 구조

다중 관리자(Multi-peer Manager)를 사용한다. (그림 3)에서와 같이 하나의 관리자는 다른 관리자의 리더(Leader) 역할을 수행한다. 이 구조는 분산 네트워크에서 잘 활용될 수 있으며 이전의 구조와는 달리 영역 관리자간의 통신이 이루어진다.

3. IRMS(Information Resource Management System)의 구조

3.1 전체 구조

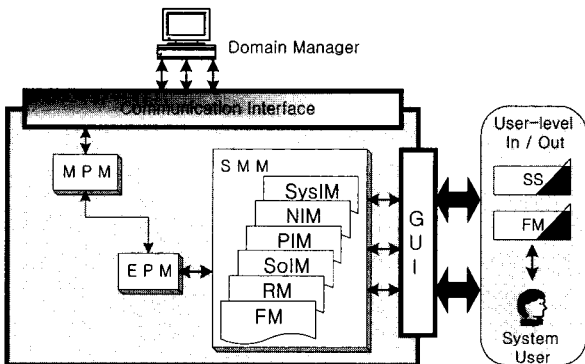
IRMS는 최상위층에 전사적 관리자(Enterprise Manager)를 포함한 1st-Tier, 중간계층에 영역 관리자(Domain Manager)를 포함한 2nd-Tier, 그리고 최하위층에 자원 관리대행자(Resource Agent)를 포함하는 3rd-Tier로 구성된다. 각 관리 영역(Domain)마다 하나의 영역 관리자가 설치되고 관리하며, 이 영역 관리자는 상위 계층의 전사적 관리자에 의해서 관리된다. (그림 4)는 시스템의 전체 구조를 나타낸 그림이다[6].



(그림 4) IRMS의 전체 구조도

3.2 계층별 내부 구조

3.2.1 관리 대상 계층 (3rd-Tier)



(그림 5) 3rd-Tier 내부 구조도

관리 영역 내의 모든 관리 호스트들에 해당하는 계층으로, 본 계층에 속한 피관리 호스트들은 자원 관리대행자를 탑재하여 관리 시스템의 관리를 받는다. 본 계층의 자원 관리대행자의 내부 구조도는 (그림 5)와 같다.

본 계층에서 사용되는 모듈은 다음과 같다.

1) 통신 인터페이스 (Communication Interface)

통신 인터페이스는 상위 계층이나 하위 계층간의 통신, 관리자와의 통신을 위한 인터페이스이다. 각 통신 구조는 다음과 같다.

- User Level Interface Communication : 시스템 사용자가 자원 관리대행자의 GUI를 통해서 제공받는 통신 모듈로서 User-level에서는 상위 계층의 관리자와의 대화창으로서 통신 인터페이스가 제공된다.
- Agent-to-DM Communication : 영역 관리자와 자원 관리대행자간의 통신으로서 자원 관리대행자의 내부적 사건(Interior Event) 혹은 외부적 사건(Exterior Event)에 의해서 요청(Request)과 응답(Response)이 이루어 지는 통신 관계이다. 주로 시스템 사용자에게 의한 장애관리나 서버 설정을 수행하기 위한 통신이다.

2) 사용자 인터페이스(User-level Interface)

시스템 사용자가 직접 시스템에 접근할 수 있는 인터페이스이며 서버 설정이나 시스템이 장애에 관련된 설정을 수행할 수 있다.

3) 메시지 분석 모듈(Message Parsing Module)

상위 계층의 통신 인터페이스를 통해 들어오는 메시지를 받아들이는 모듈이다. 이 모듈을 통해서 입력된 메시지가 어느 정보인지를 판단하여 EPM으로 전달하여 적절한 이벤트를 호출하고 이에 대한 처리를 가능하게 한다.

4) 이벤트 처리 모듈(Event Processing Module)

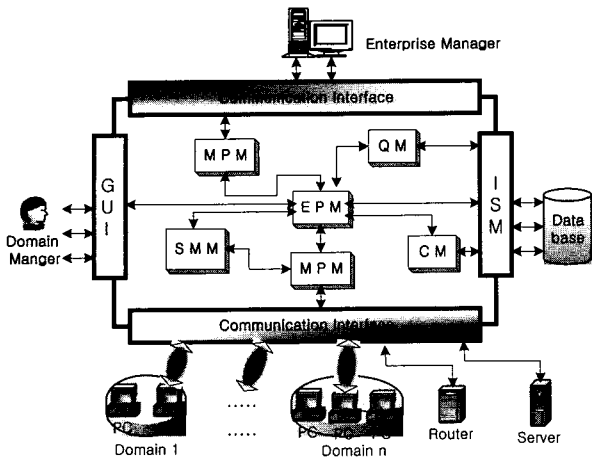
메시지 분석 모듈을 통해서 분석된 메시지가 들어오면 이에 맞는 이벤트 핸들러를 동작시킨다. 이를 통해 시스템의 동작이 원하는 대로 수행되게 되고 이벤트 처리 모듈에는 상위 계층의 시스템으로부터 들어오는 메시지에 대해 호출되는 내부적 이벤트 호출 모듈과 사용자에게 의해 GUI로 입력되는 외부적 이벤트 호출 모듈이 존재한다.

5) 시스템 관리 모듈(System Management Module)

상위 관리 시스템에 의해서 관리되는 모듈로서 정의의 내용은 다음에 설명될 것이다.

3.2.2 서버 관리자 계층(2nd-Tier)

전체 시스템의 계층상 중간에 해당하는 계층으로 상위계층의 전사적 관리자에 의해서 요청되는 이벤트를 처리하거나 하위의 관리 영역을 직접 관리하기 위한 기능을 수행한다. 서버 관리자 계층의 내부 구조도는 다음 (그림 6)과 같다.



(그림 6) 2nd-Tier 내부 구조도

1) 환경 설정 모듈(Configuration Module : CM)

시스템이 관리하고자 하는 관리 대상에 대한 초기화를 수행하며 관리 정책 및 동작 위한 설정을 수행하는 모듈이다. 환경 설정에 대한 정보는 초기에 입력을 통해 저장되며 주기적으로 폴링(Polling)을 하여 그 결과를 갱신(Update)한다.

- 관리 대상 설정 : 관리자에 의해 관리를 받을 피관리 객체를 설정한다. 관리 대상에는 PC 및 Router가 포함되며 자산 관리를 위한 여러 가지 네트워크 장비 및 서버 장비도 포함된다.
- 관리 도메인 설정 : 관리자가 원하는 범위를 시스템을 관리하기 위해 영역을 설정한다. 이 정보는 관리자가 GUI(Graphical User Interface)를 통해 입력하며 정보 저장 모듈(ISM)을 이용하여 데이터 베이스에 저장된다.

2) 정보 검색 모듈(Query Module : QM)

관리 대상으로 등록된 시스템들에 대한 정보를 검색하기 위한 모듈이다. 정보 검색 모듈은 다음과 같이 각각의 시스템의 종류대로 구별하여 검색이 가능하다.

- PC별 정보 검색 : 피관리 시스템 중에서 PC들을 개별, 혹은 그룹별로 검색하기 위한 모듈이다. 이 기능을 통해서 피관리 시스템들의 통계적인 정보의 비교를 통해서 현재 관리되는 시스템들의 현황을 분석하고, 관리 정책의 자료로 사용할 수 있다.

3) 통신 인터페이스(Communication Interface : CI)

통신 인터페이스는 상위 계층이나 하위 계층간의 통신, 관리자와의 통신을 위한 인터페이스이다. 각 통신 구조는 다음과 같다.

- User Level Interface Communication : 관리자가 영역 관리자의 GUI를 통해서 제공받는 통신 모듈로서 User-level에서는 각 계층간의 실무 관리자간의 대화창으로서 통신 인터페이스가 제공된다.

- DM-to-Agent Communication : 영역 관리자와 자원 관리대행자간의 통신으로서 영역 관리자의 내부적 사건 혹은 외부적 사건에 의해서 요청과 응답이 이루어 지는 통신 관계이다.

- DM-to-EM Communication : 영역 관리자(DM)와 전사적 관리자(EM)간의 통신으로서 EM이 DM을 관리하기 위한 통신관계를 구현한 인터페이스이다.

4) 메시지 분석 모듈(Message Parsing Module : MPM)

상위 계층, 혹은 하위 계층의 통신 인터페이스를 통해 들어오는 메시지를 받아들이는 모듈이다. 이 모듈을 통해서 입력된 메시지가 어느 정보인지를 판단하여 EPM으로 전달하여 적절한 이벤트를 호출하고 이에 대한 처리를 가능하게 한다.

5) 이벤트 처리 모듈(Event Processing Module : EPM)

메시지 분석 모듈을 통해서 분석된 메시지가 들어오면 이에 맞는 이벤트 핸들러를 동작시킨다. 이를 통해 시스템의 동작이 원하는 대로 수행되게 되고 이벤트 처리 모듈에는 다음과 같은 두 종류의 모듈이 있다.

- 내부적 이벤트 호출 모듈(Interior Event Invoke Module) : 상위 계층이나 하위 계층의 시스템으로부터 들어오는 메시지에 대해 호출되는 이벤트 호출 모듈이다. 상하위 계층으로의 Request/Response 메시지에 대한 호출이며 내부적으로 처리되는 모듈이다.
- 외부적 이벤트 호출 모듈(Exterior Event Invoke Module) : 실제 관리자에 의해서 요구되는 동작에 대한 이벤트 호출 모듈이다. 관리자는 시스템에서 제공하는 GUI를 통하여 원하는 동작을 수행하며 시스템의 외부에서 들어오는 이벤트에 대한 호출 모듈이다.

6) 정보 저장 모듈 (Information Saving Module : ISM)

환경 설정 모듈이나 정보 검색 모듈, 혹은 이벤트 처리 모듈의 요청에 의해서 데이터 베이스에 접근하여 각 모듈의 요청에 응답하는 모듈이다. 데이터 베이스에 대한 일관성있는 접근을 위해 하나의 모듈을 생성하였으며 이로 인해 데이터 무결성이 보장된다.

7) 시스템 관리 모듈(System Management Module : SMM)

- 시스템 정보 관리(System Information Management : SysIM)

관리 시스템의 시스템 정보를 수집하고 관리정보를 서버로 전송한다. 이때 수집되는 시스템 정보항목은 <표 1>과 같다.

- 네트워크 정보 관리 (Network Information Management : NIM)

관리 시스템의 네트워크 정보 수집 및 설정을 제공하며, 관리 범주 내의 호스트 중 IP 충돌을 방지하는 기능을 제공한다.

<표 1> 시스템 정보관리 세부기능

종 류	수집항목	설 명
System	SysProcessor SysWhatBios SysVideo SysPrinter SysMemory	Processor의 종류 Bios 종류 및 버전 Video Card의 종류 프린터의 종류 메모리량
OS	OsHostName OsComName OsWinVer OsFlat OsWinDir	호스트의 이름 컴퓨터의 이름 윈도우 버전 OS의 플랫폼 종류 윈도우 설치 디렉토리
Hard disk	HardNum HardLabel HardFileSys HardTotal HardUsage	하드디스크의 갯수 하드디스크의 레이블 하드디스크의 파일시스템 총 하드디스크량 하드디스크의 사용량

● 프로세스 정보 관리(Process Information Management : PIM)

프로세스 관리 기능은 에이전트가 수행중인 프로세스 정보를 관리자 시스템에서 실시간으로 파악하고, 불필요한 프로세스가 수행중인 경우 이를 원격에서 중지할 수 있는 기능으로 프로세스 정보를 주기적으로 갱신되어 관리자 시스템에 보내진다.

● 소프트웨어 정보 관리(Software Information Management : SoIM)

소프트웨어 관리기능은 해당 시스템 내에 설치되어 있는 소프트웨어 정보를 수집 통보하며, 이를 바탕으로 관리자 시스템에서 설치가 필요한 소프트웨어를 원격 배포할 수 있도록 하는 기능이다.

● 원격 관리(Remote Management : RM)

원격관리기능은 크게 실시간 모니터링과 원격 프로그램 실행으로 구성된다. 먼저 실시간 모니터링은 관리자 시스템의 화면을 원격지에서 실시간으로 모니터링하여 볼 수 있으며, 이를 비트맵 파일로 저장할 수 있다. 원격 프로그램 실행은 관리자 시스템에서 에이전트 프로그램을 실행시키는 기능이다.

● 장애 관리(Fault Management : FM)

장애 관리기능은 관리 도메인 내에서 발생하는 장애 상황들을 관리자 시스템에서나 관리 PC에서 자체적으로 수행하기 위한 기능으로 관리자는 그룹단위나 개별 PC단위로 시스템 구동에 필요한 백업 항목과 백업 주기를 설정한다.

● 그룹 정보 관리(Group Information Management : GIM)

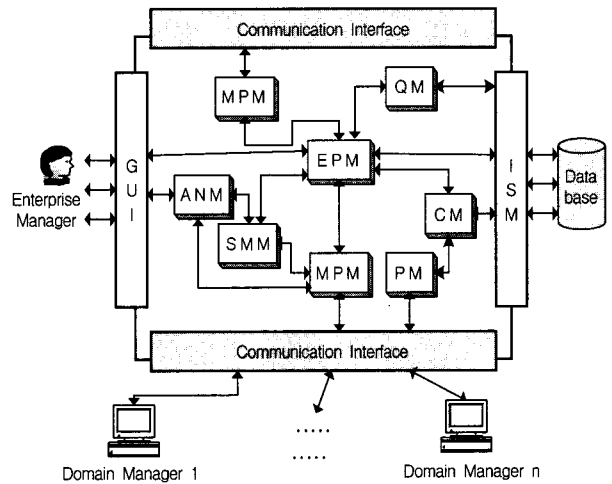
상기의 개별 호스트 관리를 원활하고 조직적으로 관리하기 위하여 그룹으로 피관리 대상을 구별하고 관리하기 위한 기능이다. 세부적인 기능은 동일하나 그룹으로 관리하므로 정책적인 선택에 의해 통합 관리가 수월하다.

● 자산 관리(Asset Management : AM)

피관리 대상의 시스템들을 정보자원으로 간주하고 이를 관리하기 위한 기능이다. 조직적이고 세분화된 자산 관리를 통해 관리자는 정보 검색 모듈과 함께 관리의 조직화가 가능하다.

3.2.3 주 관리자 계층(1st-Tier)

최상위의 관리자 계층으로서 2nd-tier의 영역 관리자들을 관리하고 네트워크 장비 및 서버 장비들을 관리 하기 위한 관리 계층이다. 내부 기능 중 2nd-tier에서 수행하는 기능은 설명을 배제하였으며, 주 관리자 계층의 내부 구조도와 추가되는 모듈은 다음 (그림 7)과 같다.



(그림 7) 1st-Tier 내부 구조도

1) 환경 설정 모듈 (Configuration Module : CM)

서브 관리자 계층에서의 마찬가지로 시스템이 관리하고자 하는 관리 대상에 대한 초기화를 수행하며 관리 정책 및 동작을 위한 설정을 수행하는 모듈이다.

- 라우터 회선 관리정보 : 성능 분석 모듈(PM)에 의해서 주기적으로 라우터의 회선에 폴링을 하기 위한 환경 설정이다. PM은 이 정보를 바탕으로 라우터의 회선을 관리하기 위한 폴링을 수행한다. 이 정보를 위한 항목은 다음 <표 2>와 같다[2].

2) 성능 분석 모듈(Performance Module : PM)

환경 설정 모듈에서 설정된 회선 관리 정보의 내용을 바탕으로 각 시스템으로부터 정보를 수집하고 분석하며 이를 바탕으로 분석결과를 보고(Reporting)하는 모듈이다. 분석결과가 임계값(Threshold)을 초과하는지 조사하고 만약 초과시에는 관리자에게 보고한다. 또한 여러 가지 출력 형태로 표현(Presentation)되는 결과를 통해 관리자는 네트워크에 대한 상황을 파악할 수 있는 자료로 응용할 수 있으며 장애에 대한 문제를 사전에 탐지할 수 있다[8, 9].

- 주기적 폴링 : 주기적으로 관리 시스템에 대한 정보를 얻어오기 위해 폴링 쓰레드가 동작하며 주기나 항목에 대한 정보는 환경 설정 모듈에 의해 저장된 데이터를 이용한다. 라우터 회선의 경우 주기적인 폴링으로 수집된 데이터를 여러 가지 형태의 보고서를 통해 관리자가 네트워크 관리에 대한 정책 수립을 위한 자료로 활용하도록 한다.
- 보고 기능 : 폴링으로 수집된 데이터를 일간분석/주간분석/월간분석/실시간 분석으로 분류하여 관리자에게 제공하는 기능이다.

<표 2> 라우터 폴링 항목

항 목	관련 MIB 객체	의 미
선로 이용률	ifInOctets, ifOutOctets, sysUpTime, ifSpeed	라우터의 입출력 패킷량을 분석하여 실제 선로 이용률을 계산한다.
입력 패킷량	IfInUcastPkts, ifInNUcastPkts	인터페이스로 입력되는 패킷량을 단위시간으로 표현한다.
출력 패킷량	IfOutUcastPkts, ifOutNUcastPkts	인터페이스로 출력되는 패킷량을 단위시간으로 표현한다.
입력 바이트량	ifInOctets	단위시간당 입력 바이트량을 출력한다.
출력 바이트량	ifOutOctets	단위시간당 출력 바이트량을 출력한다.
입력 에러량	ifInErrors, ifInUnknownProtos, ifInDiscards, ifInUcastPkts, ifInNUcastPkts	인터페이스로 유입시 에러를 포함하는 패킷의 양을 단위시간으로 표현한다.
출력 폐기량	IfOutOctets, ifOutUcastPkts, IfOutNUcastPkts, IfOutDiscard, IfOutErrors,	인터페이스로 유출시 에러를 포함하는 패킷의 양을 단위시간으로 표현한다.

성능 분석은 <표 2>의 폴링된 관련 MIB객체로 특정 공식에 의하여 산출된다. <표 3>은 성능 분석항목의 산출 계산식을 나타낸 표이다.

<표 3> 성능 분석항목의 산출 계산식

분석 항목	계 산 식
선로 이용률	$\frac{Max (if In Octets, if Out Octets) * 8 * 100}{((sysUpTime/100) * speed)}$
입력 패킷률	$\frac{if In UcastPkts + if In NUcastPkts}{(sysUpTime/100)}$
출력 패킷률	$\frac{if Out UcastPkts + if Out NUcastPkts}{(sysUpTime/100)}$

3) 알람 통지 모듈(Alarm Notification Module : ANM)

성능 분석 모듈에 의해서 주기적으로 관리 시스템으로부터 정보를 폴링하고 분석하는 과정에서 임계값(threshold)을 초과하는 경우나 장애가 발생한 경우 이를 관리자에게 통지하기 위한 모듈이다. 관리자는 관리 시스템이나 네트워크, 혹은 장비의 이상 징후를 이 모듈을 통해서 통지 받을 수 있으며 이를 통해 현 상황을 판단하는 자료로 사용할 수 있다.

4) 정보 검색 모듈(Query Module : QM)

서브 관리자 계층에서와 같이 관리 대상으로 등록된 시스템들에 대한 정보를 검색하기 위한 모듈이다. 정보 검색 모듈은 서브 관리자 계층에서 수행하는 검색 외에 다음과 같은 검색이 가능하다.

- Router별 정보 검색 : 관리 대상이 되는 라우터의 정보를 검색하는 기능이다. 이를 통해 현재 관리되는 네트워크의 상황을 파악하고 나아가 관리적 측면에서 이를 활용할 수 있다.
- 네트워크 장비별 정보 검색 : 관리자에 의해서 관리가 되고 있는 네트워크의 장비별로 정보를 검색하는 모듈이다. 이 정보를 바탕으로 관리자는 네트워크 장비의 현재 상태 및 잠재적인 문제들을 파악할 수 있다.

5) 통신 인터페이스(Communication Interface : CI)

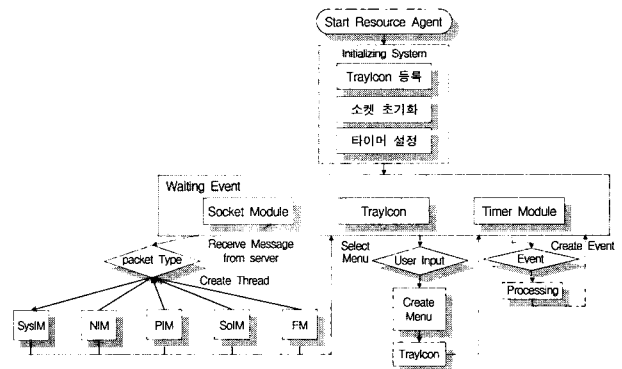
서브 관리자 계층에서 정의된 인터페이스 외에 다음과 같은 인터페이스가 추가된다.

- EM-to-Agent Communication : 전사적 관리자(EM)와 자원 관리대행자간의 통신으로서 EM이 관리 영역내의 자원 관리대행자 및 피관리 네트워크 장비들에 대하여 직접 통신을 하기 위한 인터페이스이다.

4. IRMS의 설계 및 구현

4.1 자원 관리대행자(Resource Agent)의 내부 흐름도

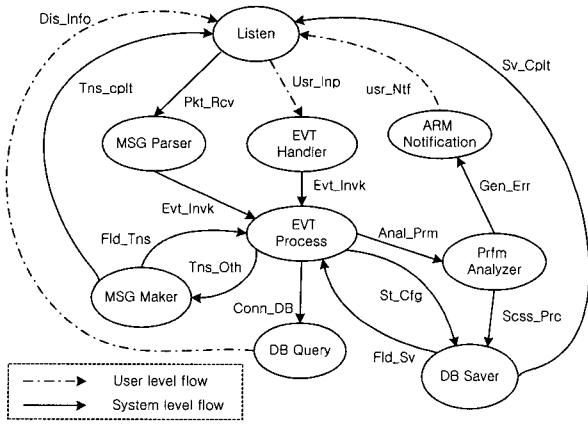
(그림 8-1)은 자원 관리대행자의 내부 흐름도로 크게 시스템 초기화모듈, 요구처리모듈, 화면 인터페이스 모듈, 타이머 모듈로 구성된다. 시스템 초기화 모듈은 다이얼로그의 생성, 트레이 아이콘 생성, 타이머 설정, 메인 소켓을 생성하여 서버로부터의 요구를 대기하는 모듈로 구성된다. 타이머 모듈은 주기적으로 수행되어야 하는 백업 기능을 수행하는 모듈로 시스템 가동 시 타이머를 설정하고, 이를 처리한다. 요구처리 모듈은 서버로부터 메시지 수신 시 요구를 처리하고 그 결과를 서버로 전송하는 모듈로 각 기능별로 쓰레드를 생성하여 동작한다.



(그림 8-1) 자원 관리대행자의 내부 흐름도

4.2 관리자의 상태 천이도

(그림 8-2)은 관리자의 상태 천이도를 나타낸 것이다. 이는 관리자가 실행되고 동작되는 동안의 모든 사건과 그에 대한 상태 천이를 표현한 것이다.



(그림 8-2) 관리자의 상태 천이도

다음 <표 4>는 관리자의 상태를 천이하기 위한 사건의 종류와 그것의 의미를 나타낸 표이다.

<표 4> 관리자의 사건의 의미

Event Meaning	
Dis_Info : Display Information	Fld_Tns : Failed Transportation
Tns_cplt : Complete Transportation	Tns_Oth : Transport to Other-layer
Pkt_Rcv : Receive Packet	Conn_DB : Connect Database
Usr_Inp : User Input	Fld_Sv : Failed Saving
Usr_Ntf : Notify User	St_Cfg : Set Configuration
Sv_Cplt : Complete Saving	Anal_Prm : Analyze Performance
Evt_Invk : Invoke Event	Scss_Proc : Succeed Process
Gen_Err : Generated Error	

또한 <표 5-1>과 <표 5-2>는 각 상태에 대한 수행과정을 pseudo code로 나타낸 것이다.

<표 5-1> 관리자의 pseudo code

Module pseudo code	
Listen : Receive the Event If (Event == User_Input) Goto EVT_Handler : Else if (Event == Pkt_Input) Goto MSG_Parser : EVT_Handler : Extract ID from Event If (ID == DB_Qry) Goto DB_Query : Else if (ID == Ctl_Msg) Goto MSG_Maker :	DB_Query : Analyze and create SQL Connect DB Execute SQL Close DB Display the result Return DB_Saver : Analyze and create SQL Connect DB Execute SQL Close DB If (complete saving) Return Else Goto EVT_Handler:

Listen, EVT_Handler, DB Query, DB_Saver, Prfm_Analyzer, MSG_Parser, ARM_Notification, MSG_Maker는 각 상태(State)를 나타내며 이에 대한 설명이 이루어진다.

<표 5-2> 관리자의pseudo code

Module pseudo code	
Prfm_Analyzer : Analyze captured Data Read saved Data Compare captured Data With saved Data If (excess threshold) Goto ARM_Notification : Else Goto DB_Saver : MSG_Parser : Extract header from MSG If (MSG == Ctl_Msg) Goto MSG_Maker : Else if (MSG == Cnf_Msg) Goto DB_Saver : Else if (MSG == Perform_MSG) Goto Prfm_Analyzer :	ARM_Notification : Analyze input data Notify the result Return MSG_Maker : Analyze request data Extract msg If (msg == sysInfo) Make (WM_HOST_SYSINFO) Else if (msg == NetInfo) Make(WM_HOST_NETINFO) Else if (msg == softInfo) Make (WM_HOST_SOFTINFO) Else if (msg == pslst) Make (WM_HOST_PLIST) Else if (msg == filelst) Make (WM_HOST_FILELIST) Return

4.3 계층간 통신 메시지 형식

각 계층에서 사용하는 메시지의 형식은 다음과 같다[4].

Source	Destination	MSG Type	Information
--------	-------------	----------	-------------

(그림 9) 메시지 형식

(1) EM-to-DM 통신

전사적 관리자와 영역 관리자간의 통신으로서 이 경우 Source, Destination, MSG Type가 나타내는 값은 <표 6>과 같다.

(2) DM-to-Agent 통신

영역 관리자와 자원 관리대행자간의 통신을 나타내는 것으로서 Source, Destination, MSG Type가 나타내는 값은 <표 6>과 같다.

(3) EM-to-Agent 통신

전사적 관리자와 자원 관리대행자간의 통신으로서 Source, Destination, MSG Type가 나타내는 값은 <표 6>과 같다.

<표 6> 메시지 형식의 값

Source	MSG Type			
0 EM	0 WM_HOST_SCREEN	6 WM_HOST_PEXE		
1 DM	1 WM_HOST_PLIST	7 WM_HOST_SRVSET		
Destination	2 WM_HOST_SYSINFO	8 WM_HOST_ONOFF		
0 EM	3 WM_HOST_NETINFO	9 WM_HOST_FILELIST		
1 DM	4 WM_HOST_NETMOD	10 WM_HOST_FTRANSFER		
2 Agent	5 WM_HOST_PKILL	11 WM_HOST_SOFTINFO		

5. 실험 및 고찰

5.1 구현 환경 및 사용 언어

IRMS는 visual C++ 5.0 이상의 라이브러리와 Windows API를 지원할 수 있는 윈도우 상에서 개발했다. 자원 관리 대행자는 Windows 98/2000을 대상으로 하였고, 관리자들은 Windows 2000 Professional에서 구현하였다.

사용언어는 윈도우상에서 최소의 시스템 자원을 요하는 Windows API를 핵심으로 사용하였으며 일부 MFC(Microsoft Foundation Class)를 사용하여 객체 지향의 이점을 최대한 살렸다. 이를 위한 통합 개발환경으로는 Microsoft Visual C++ 6.0을 사용하였다.

또한 설계한 관리자과 자원 관리대행자 시스템을 구현하여 본 논문에서 설계한 시스템을 테스트하였으며 그 환경과 시스템 사양은 다음 <표 7>과 같다.

<표 7> 시스템 사양 및 테스트 환경

구분	종류	사양
Enterprise Manager/ Domain Manager	CPU	Intel Pentium III 733 MHz
	Main Memory	256 MBytes
	OS	Windows 2000 Professional
	Network Interface Card	3Com EtherLink XL 10/100 PCI Tx NIC (3C905B-TX)
	IP Address	203.252.53.46/203.252.53.45
Resource Agent	CPU	Intel Pentium III 733/900 MHz
	Main Memory	256 MBytes
	OS	Windows 98/2000
	Network Interface Card	3Com EtherLink XL 10/100 PCI Tx NIC (3C905B-TX)
관리 네트워크	Domain 1	203.252.53.40 ~ 203.252.53.59
	Domain 2	203.255.252.141 ~ 203.255.252.151
IRMS	소프트웨어	IRMS(통합 관리 시스템)
	실험 방법	실험대상 네트워크에 각 계층별로 관리 시스템 설치 후 관리 수행
	관리 항목	시스템의 관리항목 및 라우터의 회선 분석

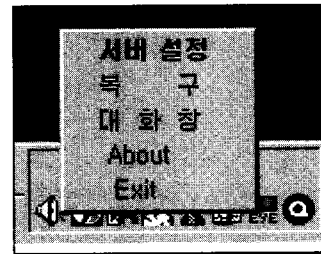
5.2 IRMS 테스트 시나리오

- (1) 영역은 크게 두 개로 구성하였으며 203.252.53.40부터 203.252.53.59까지의 IP주소를 사용하는 호스트 20대로 영역 A를 구성하였고, 203.255.252.141부터 203.255.252.151까지의 IP주소를 사용하는 호스트 10대로 영역 B를 구성하였다.
- (2) 영역 A는 다시 203.252.53.40부터 203.252.53.49까지를 그룹 1로 구성하고 203.252.53.50부터 203.252.53.59를 그룹 2로 구성하였다.
- (3) 영역 B는 203.255.252.141부터 203.255.252.146까지를 그룹 3으로 구성하고 203.255.252.147부터 203.255.252.151까지를 그룹 4로 구성하였다.

- (4) 구성된 두개의 영역 중에서 203.252.53.46과 203.255.252.142의 주소를 갖는 시스템에 영역 관리자를 설치하였으며 두 개의 영역 관리자에 의해 각각의 영역을 관리하였다.
- (5) 이 중에서 203.252.53.46의 IP주소를 갖는 호스트에 전사적 관리자를 설치하여 계층적인 관리 시스템을 설치하였다.
- (6) 각 관리 호스트에는 자원 관리대행자가 탑재되었으며 서버장비와 네트워크 장비등에 대한 등록을 설정하였다.
- (7) 이후 라우터 회선의 주기적인 분석 및 시스템에 대한 관리를 수행하였으며 각각의 시스템에서 제공하는 화면 인터페이스는 다음의 설명과 같다.

5.3 자원 관리대행자

자원 관리대행자는 각 영역의 관리 시스템마다 설치되어 피관리 시스템과 관리자와의 통신을 수행하여 정보를 수집하거나 설정하는 역할을 수행한다. 자원 관리대행자는 피관리 시스템의 부팅과 함께 자동으로 Tray Icon으로 등록되며 사용자는 (그림 10)에서 제공되는 간단한 기능을 수행할 수 있다. 그러나 이 기능은 관리 대상의 설정등에만 사용될 뿐 모든 기능은 백그라운드에서 관리자와의 통신에 의해서 수행된다.



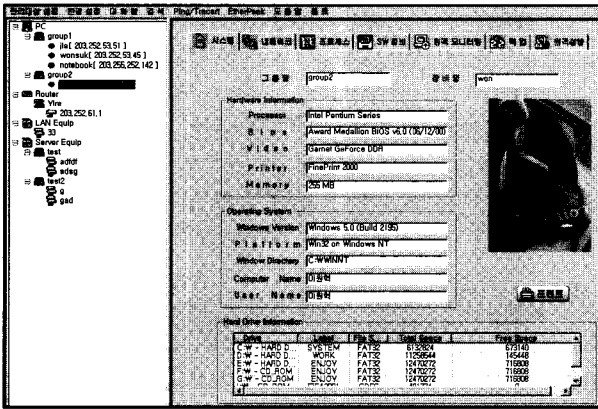
(그림 10) 자원 관리대행자의 TrayIcon

5.4 관리자 시스템

관리자 시스템에서는 관리 영역의 모든 관리 시스템들을 등록하여 그룹화된 시스템의 관리는 물론 라우터의 회선에 대한 분석을 수행한다. 이 분석에는 일간/주간/월간의 수집된 데이터를 분석하는 것과 실시간으로 나타내는 분석이 있다.

5.4.1 시스템 관리

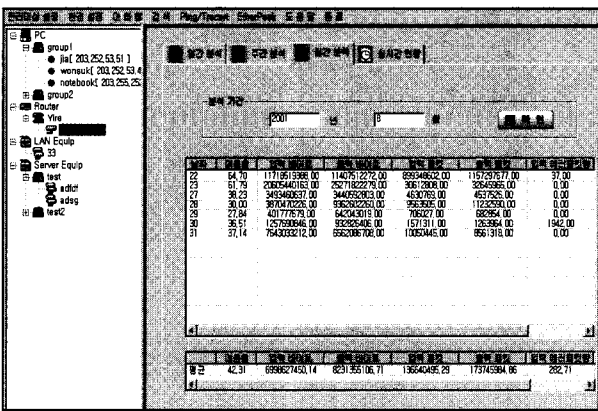
(그림 11)에서와 같이 왼쪽의 관리 대상을 선택하면 오른쪽의 뷰에서는 선택된 시스템에 대한 시스템, 네트워크, 프로세스, 소프트웨어, 모니터링, 백업등의 모든 관리를 할 수 있다. 각 피관리 시스템의 Up/Down상태 여부는 트리의 색깔로 구분할 수 있으며, 이중 녹색은 해당 호스트가 정상적으로 동작중임을 나타낸다. 각 세부 관리 항목에는 시스템, 네트워크, 프로세스, 소프트웨어 정보등의 내용을 그룹별로 통계하여 보여줌으로써 관리자는 한 시스템으로서 피관리 시스템의 모든 상태를 비교 분석할 수 있는 자료로 사용될 수 있다.



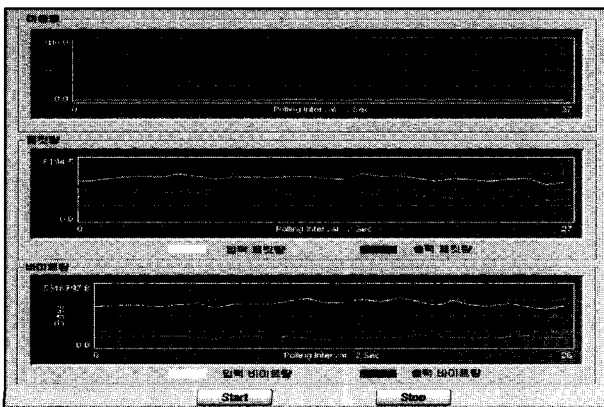
(그림 11) 시스템 관리화면

5.4.2 네트워크 장비 관리

라우터와 같은 네트워크 장비에 대한 특정 회선에 대한 정보를 사전에 정의된 항목에 따라서 수집하고 분석하여 이를 관리하도록 하는 기능이다. 왼쪽에서 관리 장비의 회선별로 일간/주간/월간 분석 결과를 보여주며, 실시간 현황을 나타내기도 한다. (그림 12)는 월간 분석 결과의 예를 나타낸 화면이고, (그림 13)은 실시간 분석 현황을 나타낸 그림이다.



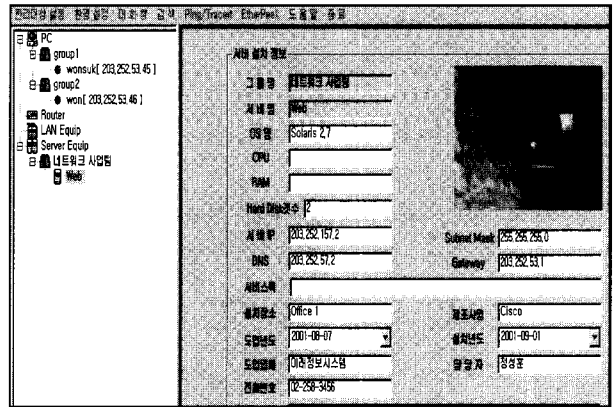
(그림 12) 월간 분석 실행화면



(그림 13) 실시간 분석 실행화면

5.4.3 서버 장비 관리

Hub/Switch나 Web Server, FTP Server와 같은 각종 서버 장비들을 통합 관리하기 위한 관리 기능이다. 이것은 각각의 영역 내의 모든 장비들에 대하여 자원관리를 통계적으로 수행하여 장비 현황을 손쉽게 볼 수 있도록 관리하는 기능이다. (그림 14)는 서버 장비에 대한 정보를 나타내는 관리 화면이다.



(그림 14) 서버 장비 관리 화면

6. 결론

본 논문의 목표는 PC 기반의 시스템 및 네트워크 장비를 관리하는 통합 관리 시스템 개발을 통해 PC와 통신 장비를 통합적으로 관리하여 네트워크 상의 모든 정보 자원 시스템들을 효율적으로 관리하고자 하는 데 있다. 이에 본 시스템에서는 관리자가 윈도우 기반의 관리 시스템에서 관리 대상 그룹의 PC, 라우터, 그리고 회선을 통합 관리하는 시스템으로 계층적 모델을 기반으로 개발하였으며, 이와 더불어 인터넷 통신 장비인 라우터, 허브 등에 관련된 성능분석 항목들을 정의하고 이를 기반으로 데이터를 측정하고 실측 데이터의 의미를 분석하였으며 시스템에 대한 관리 항목을 정의하고 이를 그룹으로 총괄 관리하기 위한 메시지를 정의하여 정보자원 시스템에 대한 관리를 용이하게 하였다. 또한 라우터 등의 회선에 대한 관리 항목을 정의하고 이를 바탕으로 네트워크 자원에 대해서도 모니터링하고 문제발생시 진단하여 이를 해결할 수 있는 근거를 마련하였다.

본 논문에서 제시된 IRMS는 SMS와 NMS를 통합하여 관리하는 시스템이다. 따라서 시스템 관리와 네트워크 관리가 통합되어 있어 특정 분야에 전문적인 지식을 갖추지 못한 관리자라도 손쉽게 네트워크의 모든 정보자원을 관리하고 검색할 수 있는 모델이다. 각각의 관리를 수행하기 위해 두개의 시스템이 동작하는 기존의 방법보다 IRMS는 적은 수의 내부 모듈을 구성할 수 있으며, 적절한 통합 모듈로 발생하는 메시지의 수를 줄일 수 있다. 또한 통합 모듈을 사용함으로써 각각의 시스템이 수행해야 할 수행시간이나

관리 복잡도가 적어질 수 있는 특성을 가진다.

또한 시스템의 모델을 정보 수집을 위한 자원 관리대행자와 영역을 관리하는 영역 관리자, 그리고 최상위인 전사적 관리자의 3계층으로 나누어 피관리 시스템들을 관리하는데 있어서 효율성을 높이고 확장성에 있어서도 유리한 모델을 제시하고 구현하였다. 이를 통하여 관리자는 보다 손쉽게 시스템과 네트워크 장비등의 모든 정보자원을 통합 관리할 수 있을 것으로 기대된다.

참 고 문 헌

[1] John Blommers, "Practical Planning for Network Growth," Prentice Hall PTR, 1996.

[2] 신상철, 안성진, 정진욱, "SNMP를 이용한 인터넷 분석 파라미터 추출 시스템의 설계 및 구현", 정보처리학회논문지 제6권 3호, pp.710-721, 1999.

[3] 조규익, 안성진, 정진욱, "SNMP를 이용한 PC관리 시스템의 설계 및 구현", 한국정보교육학회논문지 제3권 제1호, pp.86-93, 1999.

[4] Jiahai Yang, Peiyu Wang, Jianping Wu, "A scalable, web-based architecture for hierarchical network management," Global Telecommunication Conference, GLOBECOM '99, Vol.3, 1999, pp.1882-1888, Vol.3, 1999.

[5] K. Ikehara, K. Ikeda, K. Motomura, "Proposal of a hierarchical network management method based on network management protocol monitoring," 2000, IEEE/IFIP Network Operations and Management Symposium, 2000.

[6] Ehab Al-Shaer, "HiFi: A New Monitoring Architecture for Distributed Systems Management," Proceedings of the 19th IEEE International Conference on Distributed Computing Systems, 1999.

[7] Adarshpal S. Sethi, "A Hierarchical Management Framework for Battlefield Network Management," MILCOM 97 Proceedings, 1997.

[8] Leinwand A., Fang K., "Network Management: A Practical Perspective, Second Edition," Addison-Wesley, 1996.

[9] William Stallings, "SNMP, SNMPv2, SNMPv3, and RMON1 and 2 Third Edition," Addison Wesley, 1999.

[10] 안성진, 정진욱, "SNMP MIB-II를 이용한 인터넷 분석 파라미터 계산 알고리즘에 관한 연구", 정보처리학회, 제5권 제8호, pp.2102-2116, 1998.

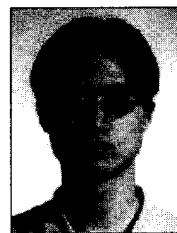
[11] Stamatiopoulos F., Chiotis T., Marglaris B., "A Scaleable, Platform-Based Architecture for Multiple Domain Network Management," IEEE International Conference on Communications '95, June, 1995.

[12] Waldbusser S., "Remote Network Monitoring Management Information Base," RFC1271, Nov., 1991.

[13] Waldbusser S., "The Trend Towards Hierarchical Network Management," the Simple Times, Vol.2, No.6, Dec., 1993.

[14] K. Marzullo, R. Cooper, M. D. Wood, and K. P. Birman, "Tools for distributed Application Management," IEEE Computer, 24(8), pp.42-51, August, 1991.

[15] D. Olege, K. Schwan, and R. Snodgrass., "Application-Dependent Dynamic Monitoring of Distributed Systems," IEEE Transactions on Parallel and Distributed Systems, 21(4), pp.593622, December. 1989.



이 원 혁

e-mail : whlee@songgang.skku.ac.kr
 2001년 성균관대학교 전기전자 및 컴퓨터공학부 졸업(학사)
 2001년~현재 성균관대학교 전기전자 및 컴퓨터공학부 대학원 석사과정
 관심분야 : 네트워크 관리, 시스템 관리, 컴포넌트, 미들웨어



안 성 진

e-mail : sjahn@comedu.skku.ac.kr
 1988년 성균관대학교 정보공학과 졸업(학사)
 1990년 성균관대학교 대학원 정보공학과 졸업(석사)
 1990년~1995년 한국전자통신연구원 연구전산망 개발실 연구원
 1996년 정보통신 기술사 자격 취득
 1998년 성균관대학교 대학원 정보공학과 졸업(박사)
 1999년~현재 성균관대학교 컴퓨터교육과 조교수
 관심분야 : 네트워크 관리, 트래픽 분석, Unix 네트워킹



정 진 욱

e-mail : jwchung@songgang.skku.ac.kr
 1974년 성균관대학교 전기공학과 학사
 1979년 성균관대학교 대학원 전자공학과 석사
 1991년 서울대학교 대학원 계산통계학과 박사
 1982년~1985년 한국과학기술 연구소 실장
 1981년~1982년 Racal Milgo Co. 객원연구원
 1985년~현재 성균관대학교 전기전자 및 컴퓨터공학부 교수
 관심분야 : 컴퓨터 네트워크, 네트워크 관리, 네트워크 보안