

□ 기술해설 □

## 분산 시스템 관리 기술

한국전자통신연구소 김경범 · 김명준\*

| ● 목                  | 차 ●                     |
|----------------------|-------------------------|
| 1. 서 론               | 4.4 그래픽 사용자 인터페이스       |
| 2. 분산 시스템 관리의 정의와 특성 | 4.5 관리 서비스              |
| 3. 분산 시스템 관리의 기능     | 5. 분산 시스템 관리 기술의 동향     |
| 4. 관리의 기술 요소 및 구현    | 5.1 표준화 동향              |
| 4.1 프레임워크            | 5.2 분산 시스템 관리 제품 및 연구동향 |
| 4.2 통신 소프트웨어         | 6. 결 론                  |
| 4.3 데이터베이스           |                         |

### 1. 서 론

컴퓨터가 사용된 이래 그 사용 형태는 여러 가지로 변화하여 왔다. 60년대의 일괄(batch)처리, 70년대의 시간 공유 처리, 80년대의 개인용 컴퓨터 활용의 형태를 지나, 90년대에 들어서는 컴퓨터 통신망을 이용한 분산 처리 환경이 새로운 사용 형태로 주목받고 있다. 분산 처리 환경은 이기종 컴퓨터간 상호 처리를 수행하며, 보다 사용자 중심의 환경으로 변화하는 모습을 보인다.

분산 환경은 고속 통신망을 연결하여 이제까지 독자적으로만 사용되던 컴퓨터 자원들을 공유할 수 있도록 한다. 컴퓨터 통신망에 연결되어 있는 컴퓨터의 자원, 즉 저장 장치, 소프트웨어, 처리 능력 등을 공유하고, 데이터의 처리를 분산시킬 수 있는 처리 방식이 도입되었으며, 현재 고객-서버(client-server) 모형 등의 형태로 활용되고 있다[1].

이러한 분산 처리 환경이 구축되어 활용됨에

따라, 기존에 단일 시스템을 관리하던 방식은 더 이상 통신망을 이용하는 분산 환경에 적용될 수 없게 되었다. 왜냐하면 분산되어 있는 자원들의 다양성, 처리의 분산, 구성 기종들의 이기종성, 확장성 등을 가지고 있는 분산 환경의 특성을 효율적으로 관리할 수 없기 때문이다. 따라서 컴퓨터 통신망 관리와 시스템 자원 관리에 분산 처리의 특징을 고려한 새로운 관리 기법이 필요하게 되었다.

어느 조직을 운영하는 데 관리는 필수불가결한 것이다. 조직의 구성원에 대한 관리, 조직의 운용에 관한 관리를 올바르게 수행하여야만 그 조직을 활용해서 얻을 수 있는 효과를 최대한으로 얻을 수 있다. 컴퓨터를 이용하는 측면에서도 마찬가지이다. 좋은 관리 기법을 적용함으로써 분산 환경의 사용자들에게 신뢰성 있게 자원들을 제공할 수 있는 것이다.

이 논문에서는 분산 시스템 관리 기술에 관하여 그 특성을 살펴보고, 관리 기능들과 이를 제공해주기 위한 기술 요소들을 설명한다. 그리고 이들 기술에 대한 표준화 현황, 제품 현황 및 연구 활동의 동향을 조사한 후, 앞으로의 발전 방향을 생각해 보고 결론을 맺는다.

\*통신회원

## 2. 분산 시스템 관리의 정의와 특성

분산 처리 시스템은 어떠한 목표를 이루기 위해 서로 협조하는 독자적인 처리 장치, 데이터베이스들로 구성된다. 분산 시스템은 공유 기억 장치를 가지고 있지 않은 기계들의 집합체에서 수행되나 사용자에게는 단일 컴퓨터로 보이는 것으로 정의되고 있다. 따라서 분산 시스템 관리의는 시스템의 소유자나 사용자의 요구를 만족시킬 수 있도록 분산되어 있는 시스템 환경을 관리하는 것이다.

분산 시스템 관리의는 분산 컴퓨팅 네트워크에 대한 표준화된 관리 기능을 제공한다. 이는 표준망관리 프로토콜을 지원하고 사용자가 정의한 형태의 망 구성을 관리하는 개방된 플랫폼을 제공한다. 물리 장치, 프로토콜 스택 등에 대한 관리뿐만 아니라, 소프트웨어 응용, 화일 시스템, 접근 허용, 허가 인증 등 분산 컴퓨팅의 시스템 관리

측면을 감시하고 제어한다. Umar는 분산 컴퓨팅을 위한 참조 모델을 그림 1과 같이 나타냈다 [2].

그림에서 알 수 있듯이 분산 시스템은 컴퓨터 통신망 서비스 위에 분산 응용 서비스를 지원하기 위한 분산 컴퓨팅 플랫폼을 제공한다. 이 플랫폼은 고객-서버 지원, 분산 데이터 처리, 트랜잭션 처리, 분산 운영체제 등으로 구성된다. 이 서비스를 이용하여 사용자 응용 프로그램들을 개발할 수 있는 응용 시스템이 구성된다. 이는 사용자 인터페이스, 처리 프로그램, 데이터베이스 등을 포함한다. 각 전문 응용 분야에서는 이러한 기능들을 조합하여 응용 서비스를 만들어 최종 사용자에게 제공하게 된다. 이 때, 분산 시스템 관리의는 하부의 통신망 관리를 기본으로 하여 분산 컴퓨팅을 위한 기술들 관리, 응용 시스템 관리와 최종 사용 서비스에 대한 관리를 포함한다. 여기서도 이 모델을 기준으로 관리 측면을 설명

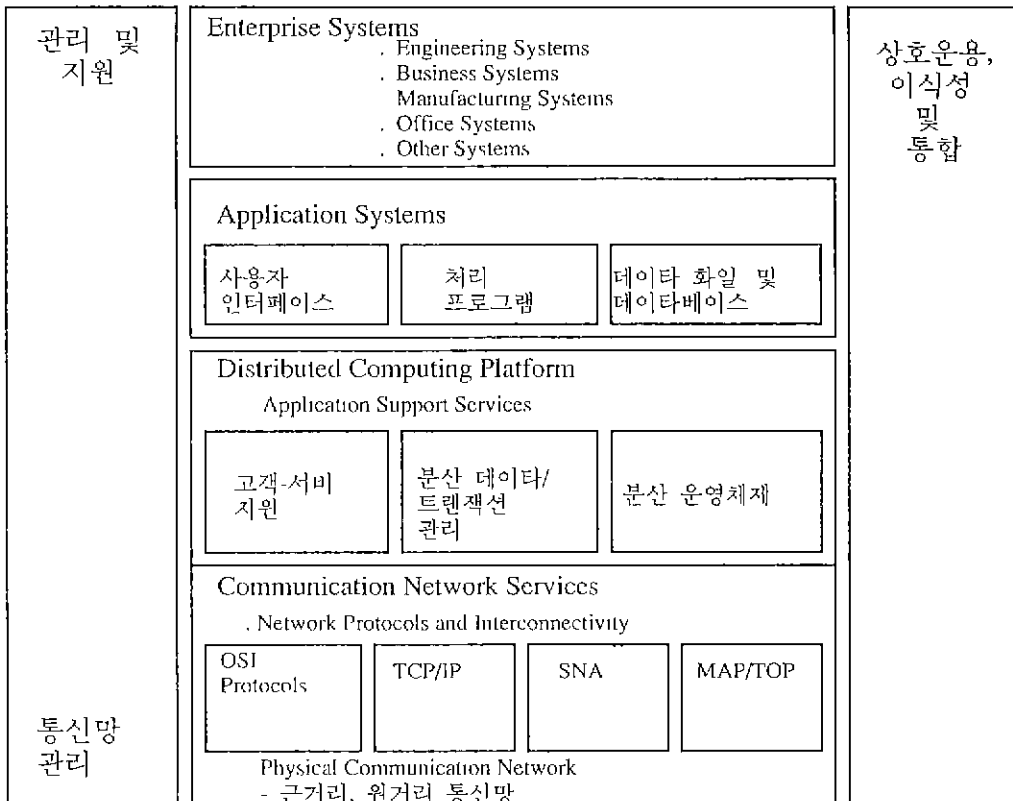


그림 1 분산 시스템의 참조 모델

한다.

한편 Umar는 관리의 주기를 계획, 조직의 구성, 개발 및 활용 그리고 분산 컴퓨팅의 감시와 제어로 나누었다. 다시 말하여, 분산 시스템 관리는 응용 개발을 위한 도구 및 방법의 개발과 같은 준비, 시스템의 가용성을 높이기 위한 시스템 운용, 앞으로의 성능 및 환경 구축을 위한 결정 등의 행정적인 측면이 결합되어야 한다. 그러나 이 논문에서는 환경의 안정된 운용을 위한 분산 컴퓨팅의 감시와 제어를 주로 설명한다.

분산 시스템 관리는 분산 시스템의 특성과 밀접한 연관을 가진다. 즉, 분산 시스템이 가지고 있는 특성을 유지하면서 분산 환경이 제공하는 효과를 최대한으로 이용할 수 있도록 관리 기술이 적용되어야 한다. CMU의 Coyler 등은 분산의 특징을 확장성(scalability), 이기종성(heterogeneity), 보안(security), 분산(distribution), 동기화(synchronization)로 나누었다[3].

분산 환경은 통신망에 독자적인 처리 장치를 갖는 기계를 연결하고 적당한 처리 기법을 운용함으로써 무한히 확장될 수 있다. 따라서 어느 분산 환경에 적용된 관리 기술이 성능의 저하를 초래하지 않으면서 이 확장성을 수용할 수 있어야 한다. 또한 분산 환경은 여러 회사들의 제품 및 운영체제로 구성될 수 있다. 이렇게 다른 특성을 갖는 구성 기기들의 차이를 흡수하여 소기의 목적을 달성 할 수 있는 환경을 제공하여야 한다. 그리고 개방된 환경에서 기기들의 자원이 이용됨에 따라 각 자원에 대한 효과적인 보안 장치가 고려되어야 한다. 이러한 환경에서 사용자가 어느 곳에 있는 자원에도 접근할 수 있으므로 자원에 대한 보안이 상당히 중요하다. 분산과 동기화는 분산 시스템이 가지는 기본적인 문제이다. 한 시스템에서 자원을 수정할 경우, 다른 시스템의 동작에 영향을 미치게 되므로 적절한 상태의 유지를 위하여 이를 잘 관리하여야 한다.

### 3. 분산 시스템 관리의 기능

분산 시스템 관리는 과연 무엇을 어떻게 관리하는 것인가? 어떠한 기능들이 갖추어질 때

분산 환경을 바르게 관리할 수 있는 것인가? 이러한 것을 이 장에서 살펴본다.

기능을 살펴보기 위해서는 어떠한 것들이 관리의 대상이 되어야 하고, 이러한 대상에 대하여 어떠한 일들을 하여야 하며, 어느 정도의 수준으로 지원되어야 하는가를 분류할 필요가 있다. 먼저 관리의 대상은 분산 환경을 구성하는 기기들과 이 기기들을 사용하는 사용자, 그리고 사용자들이 수행하는 업무들로 나누어 볼 수 있다. 분산 환경의 기기에는 컴퓨터 통신망, 하드웨어, 응용 서비스를 포함한 소프트웨어, 시스템들의 자원들이 있고, 사용자에 대한 관리는 사용 허가 및 사용자 등록/삭제 등의 기능이 있다.

이러한 대상에 대하여 일반적으로 구성 관리, 장애 관리, 회계 관리, 보안 및 성능 관리 등의 관리 기능으로 구분하고 있다.

구성 관리는 분산 환경에서 하드웨어나 소프트웨어를 설치하여 올바르게 계속적으로 동작할 수 있도록 하기 위한 것이다. Langsford는 이 구성 관리의 기능을 ABC로 나누었다[4]. A는 Administrative 기능을 나타내는데, 이 기능은 시스템에 요구되는 하드웨어, 소프트웨어들을 결정하고 이에 대하여 서비스를 할당하는 것이다. B는 Binding을 나타내는 것으로, 구성 요소들이 제공하는 인터페이스들이 정확히 정합될 수 있도록 구성 요소들간의 상호 관계를 정의하고 유지하는 것이다. Control을 의미하는 C 기능은 서비스를 동적으로 만들고 삭제할 수 있도록 구성 요소들의 상태를 만들고 감시하고 지정하는 기능이다. 다시 말하여, 환경을 구성하는 여러가지 하드웨어나 소프트웨어의 판(version)에 대한 정보를 유지하고, 이들 구성 요소들 간의 연관 관계를 설정하고 기록한다. 그리고 시스템의 통합을 용이하게 하기 위하여 시스템에 대한 표준을 정하고, 구성 요소들 간의 기능과 인터페이스를 명확히 정하고, 변경 관리를 위한 지침과 책임을 정한다. 구성 관리는 다른 관리 기능, 특히 장애 관리와 성능 관리 기능과 밀접한 관련을 가진다.

장애 관리는 시스템 동작 중의 비정상 상태인 장애를 검출하고 분산 환경을 재구성함으로써 이 장애로부터 정상 상태로 복구하며, 장애를 진단

하고, 재구성과 수리를 통하여 원래의 상태로 회복하여 주는 장애 해결로 나누어진다. 장애 검출은 각 기기들의 장애나 기기들간의 경로의 장애를 검출하고 검출된 장애를 통지하며, 이것이 발생하기 전에 장애를 예측하는 기능이다. 장애의 분리는 문제를 일으킨 기기를 분리하는 것이고, 적절한 행동을 통하여 오류를 회복시키고 다시 일을 수행할 수 있도록 하는 것이다.

**성능 관리**의 목적은 시스템의 성능을 사용자의 응답 요구 시간에 만족시킬 수 있도록 유지하는 것이다. 성능 관리는 우선 도착율, 대기 행렬, 병목 현상, 성능 문턱값 등 시스템의 동작을 측정하기 위한 감시 기능을 가진다. 그리고 이러한 값들이 응답 시간에 어떠한 영향을 미치고 있는지 또는 앞으로 어떠한 영향을 미칠 것인지를 분석하고 예측한다. 또한 시스템 성능을 향상시키기 위하여 통신망의 교통량을 조정하는 등 자원에 대한 조정을 수행한다.

**회계 관리**는 서비스와 자원의 사용에 대한 정보를 수집하고 대조(collate)하는 것이다. 즉, 서비스를 특별한 형태로 감시하고 보고하는 것이다. 서비스의 사용에 대하여 비용을 계산해준다. 자원의 사용에 대한 회계는 서비스 제공 비용을 결정하고 시스템의 효율성을 높인다. 서비스 사용에 대한 회계는 계약자에게 사용료를 청구하며, 서비스를 사용하거나 제공하는 비용을 결정하고, 사용자의 사용 형태를 제어한다.

**보안 관리**는 자원에 대한 접근을 증명하고 제어하는 인증, 데이터에 대한 접근을 확인하여 주는 데이터 기밀성 유지, 정보가 임의대로 변경되거나 손상되지 않도록 하는 데이터 완결성, 허가받지 않은 사용자가 정보 내용을 알 수 없도록 하는 암호/복호화 기능, 접근 제어를 위한 관리 도구의 제공, 보안 침해가 생겼을 때, 이를 알려주는 보안 경고 보고, 여러가지 보안에 관련된 일들을 추적하는 감사 기능 등을 수행한다.

고객에 대한 지원과 운용 지원도 중요한 관리 기능의 하나이다. 사용자가 정보를 요구하고 시스템의 문제를 알 수 있는 도움말창치(helpdesk)가 필요하다. 그리고 시스템의 기동/정지, 백업/재저장과 오류의 복구 등을 포함하는 운용, 사용자 프로파일의 관리와 자원의 사용을 감시하는 시

표 1 관리 기능과 관리 수준

| 서비스 범주  | 지원되는 기술  | 지원되는 구성 요소                                    | 지원의 정도                 |
|---------|--|---|------------------------|
| 장애 처리   | 장애 검출<br>장애 분리<br>장애 해결                          | 통신망<br>하드웨어<br>응용 연결 소프트웨어<br>응용 소프트웨어<br>통신망 | 자문<br>실제 지원<br>도구 및 환경 |
| 성능 관리   | 성능 감시<br>성능 분석<br>성능 조정                          | 하드웨어<br>응용 연결 소프트웨어<br>응용 소프트웨어               | 자문<br>실제 지원<br>도구 및 환경 |
| 구성 관리   | 구성 기기 실행<br>같은 권의 상호 관련성<br>별장 관리<br>참조와 및 개인 보고 | 통신망<br>하드웨어<br>응용 연결 소프트웨어<br>응용 소프트웨어        | 자문<br>실제 지원<br>도구 및 환경 |
| 회계 및 보안 | 인증<br>경고<br>같은 권의 감사                             | 통신망<br>하드웨어<br>응용 연결 소프트웨어<br>응용 소프트웨어        | 자문<br>실제 지원<br>도구 및 환경 |
| 운용 지원   | (helpdesk)(hot-line)<br>순용<br>시스템 관리<br>고려 지원    | 통신망<br>하드웨어<br>응용 연결 소프트웨어<br>응용 소프트웨어        | 자문<br>실제 지원<br>도구 및 환경 |

스템 관리, 고객에게 서비스를 잘 활용할 수 있도록 하는 고객 지원 서비스도 포함된다.

이러한 관리 기능은 분산 환경에서 명칭 서비스 관리와 감시 관리를 통하여 제공되고 운용된다. 명칭 서비스 관리는 분산 환경의 자원들에 대한 위치와 서비스의 이름들을 지정하고 사용자에게 인식시켜 주는 기능을 하며, 감시 관리는 분산 환경의 구성 요소들의 동작을 감시하고 앞에서 언급한 기능들을 수행함으로써 분산 환경의 가용성을 높이는 것이다.

한편 관리 대상 업무의 중요도는 기존하는 단일 시스템 중심 환경의 관점에서 변화되어야 한다. 즉 분산 환경으로 바뀌면서 체어가 분산되고, 통신망이 활용됨에 따라 이와 관련된 업무의 중요도가 높아진다. 그리고 단일 시스템의 구성 관리 등의 정의도 통신망의 구성 관리를 이용할 수 있도록 새로 정의되고 있다.

표 1은 분산 컴퓨팅의 자원들을 감시하고 제어하기 위하여 사용되는 사항을 정리한 것이다.

#### 4. 관리의 기술 요소 및 구현

컴퓨터 통신망으로 연결되고 이기종으로 구성된 시스템들로 이루어진 분산 환경을 관리하기 위하여 효율적이고 적용하기 쉬운 관리 구조가 필요하다. 이 장에서는 분산 시스템 관리의 골격을 이루는 각 구성 요소를 프레임워크, 통신 소프트웨어, 관리 정보, 관리 정보를 저장하는 데이터베이스, 사용자에게 편리한 환경을 제공

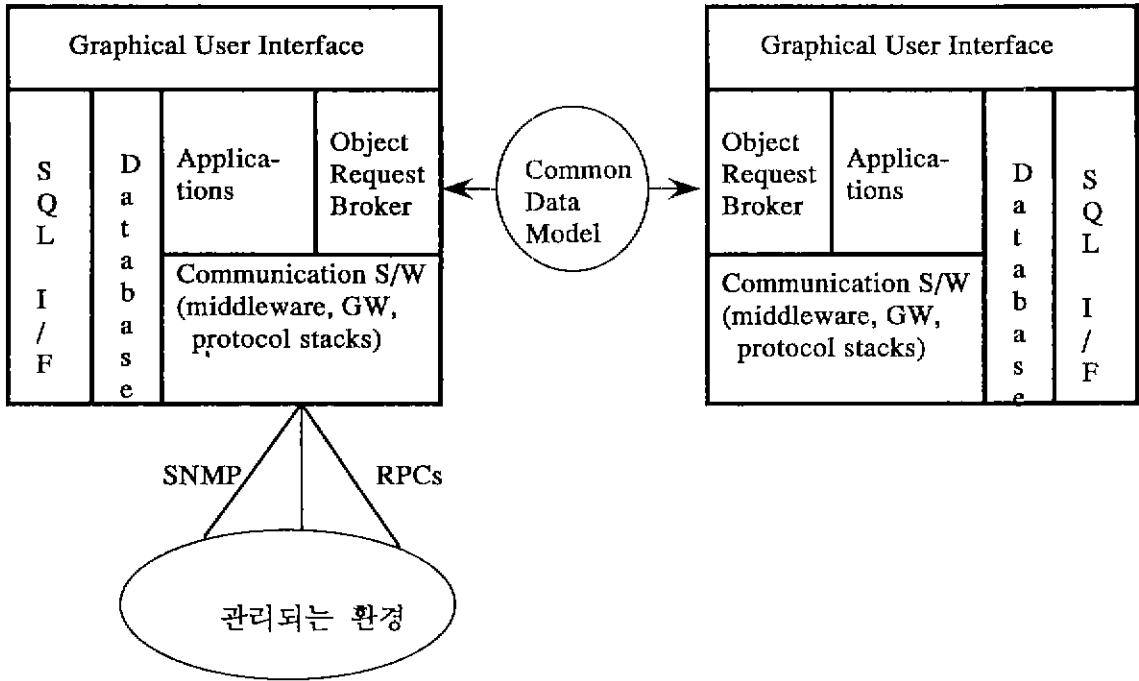


그림 2 분산 시스템 관리 프레임워크

하는 사용자 인터페이스, 관리 응용 서비스 등으로 나누어 살펴본다.

**4.1 프레임워크**

프레임워크는 통신망에 있는 모든 장비들과 시스템들로부터 정보를 수집하여 여러 응용들과 이 정보를 공유하는 소프트웨어의 플랫폼이다 [5]. 프레임워크는 미들웨어, 게이트웨이, 통신 프로토콜 스택을 포함하는 통신 소프트웨어, 응용 서비스, 사용자 인터페이스, 관리 정보의 저장을 위한 데이터베이스, 다른 프레임워크와 관리 정보를 교환하기 위한 객체 요구 중개자로 구성된다. 이 프레임워크를 가지고 여러가지 관리 통신 규약을 사용하는 통신망 기기, 컴퓨터들을 일관성 있는 형태로 관리할 수 있다. 그림 2는 프레임워크와 그들간의 통신을 나타낸다.

**4.2 통신 소프트웨어**

통신 소프트웨어는 시스템들간에 자원을 공유

할 수 있도록 제공되는 통신 규약의 스택과 분산 처리를 제공하는 미들웨어, 그리고 각 통신 장비에 대한 관리 기능을 갖는다. 통신 규약의 스택을 관리하기 위하여 현재 많이 사용되고 있는 관리 규약은 인터넷 사회의 SNMP(Simple Network Management Protocol)과 ISO에서 개방 시스템 환경을 목적으로 표준화하여 제안하고 있는 CMIP (Common Management Information Protocol)이 있다. 관리 규약은 통신망으로 연결되어 있는 시스템들간에 관리에 대한 정보를 교환하여 원하는 결과를 얻을 수 있도록 하는 약속이다.

**4.2.1 SNMP**

SNMP는 TCP/IP suite의 일부로 개발되어 현재는 컴퓨터, 통신 장비 제작 회사 대부분이 이 규약을 제공한다. SNMP에 사용되는 관리 모델은 관리 스테이션, 관리 대리인, 관리 정보 베이스, 통신망 관리 프로토콜 등의 요소를 가진다.

관리 스테이션은 데이터 분석, 장애 복구 등의

관리 응용을 가지고 있으며, 통신망 관리자가 통신망을 감시하고 제어할 수 있는 인터페이스를 제공한다. 그리고 관리자의 요구를 실제로 통신망의 기기에 전달할 수 있는 기능을 가지고 있고 통신망의 관리되는 객체들에 관한 관리 정보로부터 얻은 정보들을 보관하고 있다.

관리 대리인은 통신망 관리의 능동 요소로 관리 스테이션에서 오는 요구에 응답하고, 관리 스테이션이 지정하는 동작을 수행한다. 그리고 관리 스테이션에 정보를 비동기적으로 보내기도 한다.

통신망에서 관리되는 자원은 객체(object)로 표시된다. 이러한 객체들을 모은 것이 MIB(Management Information Base), 즉 관리 정보 베이스이다. 관리 스테이션은 MIB 객체 값을 얻음으로써 감시 기능을 수행한다. 관리 스테이션은 대리인에 어떤 동작을 수행시킬 수 있으며 특정 변수의 값을 수정함으로써 대리인의 구성 값을 변경할 수 있다.

관리 스테이션과 관리 대리인 사이에는 관리 프로토콜이 존재하여 정보를 교환한다. TCP/IP

통신망의 관리를 위해 사용되는 프로토콜이 SNMP이다. SNMP는 대리인의 객체 값을 가져 오는 Get기능, 대리인의 객체 값을 지정하는 Set기능, 대리인이 중요한 사건을 관리 스테이션에게 알려주는 Trap 기능을 기본으로 한다. 그림 3은 SNMP의 역할을 나타낸 것이다.

SNMPv2는 기존의 SNMP에 비해 SMI (Structure of Management Information), 프로토콜 동작, manager-to-manager 기능, 보안 기능 등을 확장하였다. Manager-to-manager(M2M) MIB는 분산된 관리 구조를 지원한다.

SNMP MIB를 정의하고 구성하는 구조를 SMI라고 하는 데, 이는 특정한 MIB의 구조를 정의하고, 각 객체의 구문과 값을 포함한 각 객체를 정의하며 객체 값을 부호화하는 기술을 제공한다. MIB-II는 MIB-I의 super set으로 장애 관리, 구성 관리에 필요한 객체를 추가하고 group을 추가 하였다.

#### 4.2.2 CMIS/CMIP

CMIS/CMIP은 OSI의 관리의 서비스와 이를

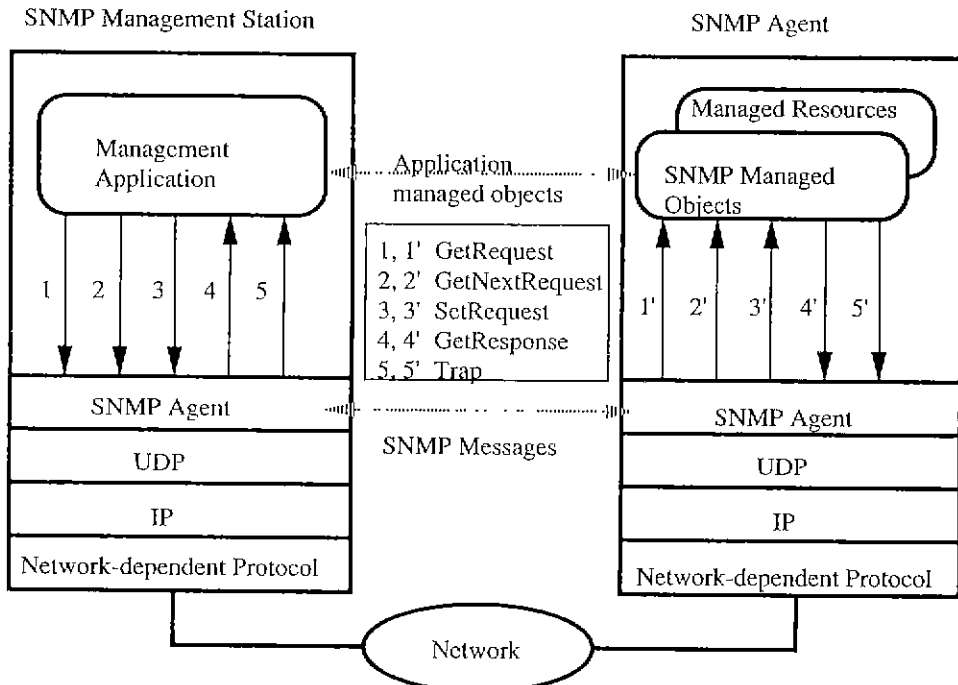
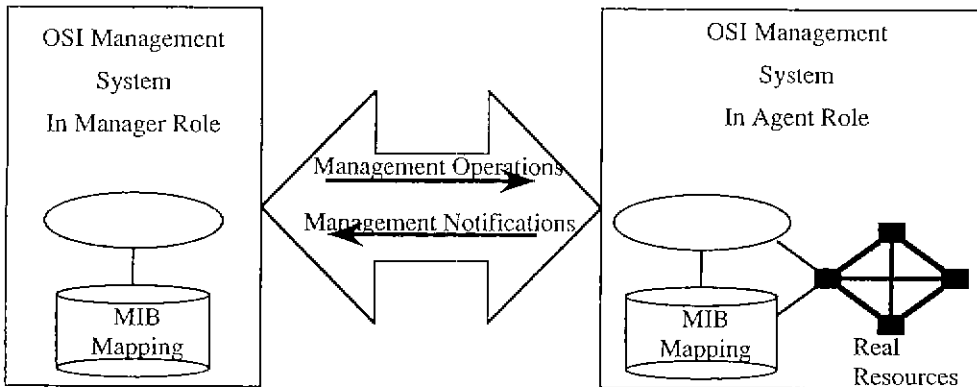


그림 3 SNMP의 역할



Management Operations : M-GET M-SET M-ACTION  
 M-CREATE M-DELETE  
 Management Notifications : M-EVENT-REPORT

그림 4 OSI 관리 시스템

수행하는 통신 규약이다. CMIS(Common Management Information Service)는 공통 관리에 사용되는 서비스 프리미티브와 argument를 정의한 것으로 관리 연계 서비스, 관리 통지 서비스, 관리 동작 서비스와 같은 세가지 유형의 서비스를 지원한다. 관리 연계 서비스는 상대편 CMISE (CMIS Element) 사용자와 연계를 설정하고 단절하는 기능을 지원하고, 관리 통지 서비스는 서비스 사용자에게 사건의 발생을 통지하는 기능이다. 관리 동작 서비스에는 정보의 검색, 검색 취소, 관리 객체의 속성값 변경, 관리 객체에 대한 특정한 동작의 수행 요구, 새로운 관리 객체에 대한 표현의 생성 및 삭제 등의 기능이 있다. 그림 4는 OSI 관리 시스템을 나타낸 것이다.

CMIP은 CMIS에서 정의된 서비스를 지원하는 규약이다. 모든 CMIP PDU (Protocol Data Unit)는 ASN.1으로 정의된다. 동작은 ROSE(Remote Operation Service Element)에서 기술된 대로 수행되므로 CMIP은 상태 천이표, 사건 목록 등을 갖고 있지 않다.

OSI의 관리 기능은 관리 대상체의 생성, 삭제와 구성 등을 정의한 객체 관리, 관리 대상체의 상태 모형을 정의한 상태 관리, 관리 대상체들 간의 연관 관계를 나타내는 연관 관계 관리, 오류 보고 및 정보 검색, 구성 관리, 소프트웨어 분배

관리, 장애 관리, 성능 관리, 보안 및 감사관리, 계정 관리, 기록 관리, 객체 스키마 관리 등이 표준 문서로 정의되어 있다.

SNMP는 단순성과 이식성이 뛰어나기 때문에 현재 가장 많이 사용되고 있다. 그러나 다양해지고 복잡해지는 통신망을 관리하는 데에는 부적합하다. 반면에 CMIS/CMIP은 좀 더 일반적이고 융통성 있게 설계되었으나 복잡하고 구현에 드는 비용이 많이 든다는 단점이 있다. 그러나 장기적으로 볼 때, CMIP도 현재 SNMP가 사용되는 것 이상으로 확대될 것으로 예상된다.

#### 4.2.3 관리 대상체 (Managed Objects)

관리 대상체는 분산 시스템 관리에서 중요한 객체로 컴퓨팅 자원을 나타내는 것이다. 파일 시스템, 장치, 인쇄, 편지 시스템, 사용자, 최종 사용자 응용 소프트웨어 등을 어느 일정한 형식으로 나타내 관리 응용을 확장성 있고, 상호 운용성 있도록 만든다. 관리는 이 관리 대상체들의 값을 검색하고 감시하며, 변경함으로써 수행된다. 관리 대상체의 구조는 SMI로 정의된다. 이는 type, 속성, 값을 가진다.

#### 4.2.4 XOM(X/Open's OSI Abstract data Manipulation)과 XMP(X/Open Management

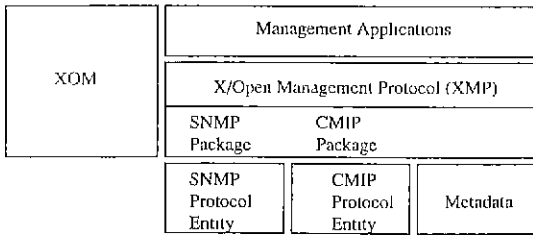


그림 5 XOM/XMP 인터페이스

**Protocol)**

하나의 프레임워크 내에서 SNMP와 CMIP을 사용하는 기기들을 통일된 방법으로 관리하기 위하여 새로운 방안들이 제안되고 연구되고 있다. 그 중의 하나가 X/Open에서 표준으로 제시하고 있는 XOM API와 XMP API이다. XOM은 어떤 구조체를 C 언어로 표현하기 위하여 X/Open에서 정의한 형식이다. 이것은 객체 지향형의 정의로 각 데이터는 속성의 집합이고 각 속성은 그 이름과 형식, 값에 의해 특정된다. XMP는 CMIP PDU를 C 언어로 나타내기 위하여 XOM 표현 형식을 이용하는 CMIP 인터페이스이다. 예를 들어 CMIP 요구, 즉 CMIS-GET-REQ를 할 때, 클래스 Cmis-Get-Req의 XOM 객체를 만든다. 이 때 클래스에 대한 속성은 객체 클래스, 객체 인스턴스 그리고 get에 대한 속성표를 가진다. CMIP에 대한 XOM 객체들이 XMP에 정의되어 있어 CMIP PDU를 XOM 형식으로 송/수신하기 위해서는 XMP 프리미티브를 사용한다. get의 경우는 mp\_get\_req()를 사용한다. 실제로 XMP 객체는 CMIS PDU의 ASN.1 표현 형식에 사상된다.

그림 5는 SNMP, CMIP과 XOM/XMP 인터페이스와의 관계를 나타낸 것이다.

GDMO(Guidelines for Definition for Managed Objects)는 MIB를 기술하기 위하여 ISO에서 정의한 언어이다. MIB를 정의할 때, 관리되는 대상체, 각 대상체가 가지고 있는 속성, 그리고 어떠한 동작이 유용한가를 정의하게 된다. 또한 속성에 대한 구문(syntaxes)도 정의하여야 한다. 이 구문에 대한 정의는 ASN.1 언어에 정의되어 있다. MIB 정의에 대한 ASN.1 구문을 처리하여 XOM 정의를 만드는 데, 변환기나 컴파일러가

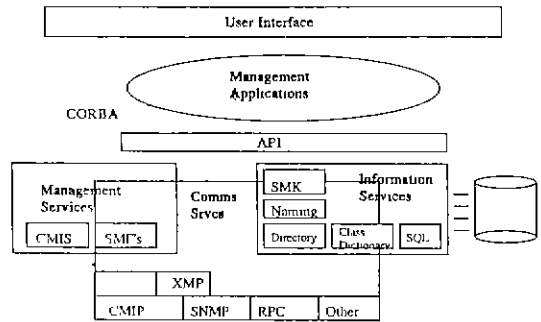


그림 6 OMNIPoint의 구조

필요하다.

현재 NMF(Network Management Forum)에서 제안한 OMNIPoint에서도 이 API를 사용하고 있으며, 여러 제품에서도 이 구조를 받아들이고 있다. OMNIPoint는 다수 공급자 환경에 대한 통합된 관리를 위한 것으로 사용자, 공급자, 그리고 표준화 단체에 의해 공동으로 개발되어 1992년에 규격이 발표되었다. 이것은 근거리 통신망, 분산 시스템, 전기통신 시스템에 대한 단일화된 접근 방식을 제공한다. OMNIPoint는 관리 시스템들간의 상호 운용, 관리 시스템과 실제의 자원 및 장치 사이의 관계, 관리 시스템을 구축하기 위한 기술들을 다루고 있다[6]. 그림 6은 OMNIPoint 1의 구조를 나타낸 것이다.

**4.3 데이터베이스**

데이터베이스는 관리 정보의 보관 및 검색에 이용된다. 그리고 이러한 DBMS를 이용할 수 있는 SQL 접근이나 스키마 관리가 필요하다. 따라서, 관리 정보를 어떠한 방법으로 축적하고 필요한 요구에 따라 어떻게 효율적으로 검색할 수 있는가 하는 것이 많이 연구되고 있다.

**4.4 그래픽 사용자 인터페이스**

관리 정보는 관리자에게 알기 쉬운 형태로 표시되어야 한다. 이를 위하여 그래픽 사용자 인터페이스를 이용한 관리 서비스가 제공되어야 한다. 그래픽 사용자 인터페이스는 관리되는 자



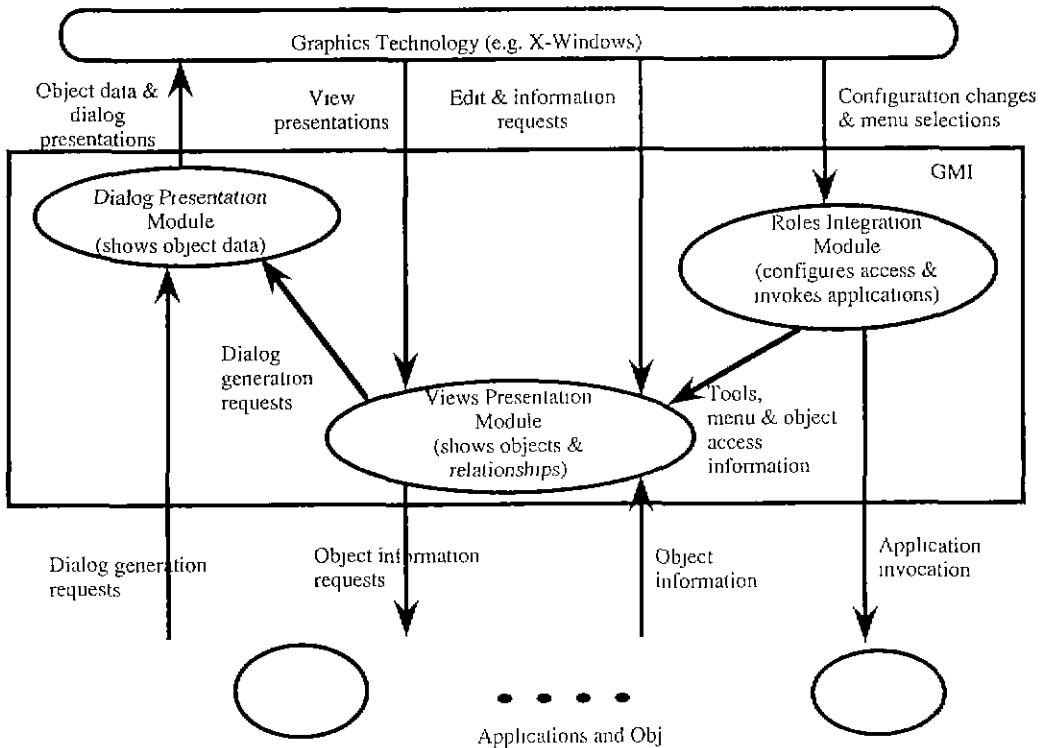


그림 7 그래픽 인터페이스의 역할

원과 그 상태를 표시한다. 이 인터페이스는 사용자의 요구 사항, 즉, 관리되는 환경에 대한 적절한 정보의 제공, 환경을 변화시키기 위한 방식의 제어, 이를 효과적으로 수행하기 위한 도구 제공 등을 만족시켜야 한다. HP의 Bartz는 그래픽 사용자 인터페이스가 관리되는 정보를 표시하는 형태를 그림 7와 같이 표시하였다[7].

그래픽 사용자 인터페이스로는 X-Windows, OSF/Motif 등이 많이 사용되고 있다. 이 그림에서 뷰 표현 모듈은 지도를 그리고 객체들을 표시하며 그들 사이의 연관 관계를 나타내며, 대화(Dialog) 표현 모듈은 요구에 따라 대화 상자, 표, chart, 그래프 등을 만들어 표시하며, 객체 데이터를 표시하고 사용자의 입력을 요구한다. 역할(Role) 통합 모듈은 응용, 도구, 보안 정책 등에 대한 특수한 관리 기능들을 수행한다.

#### 4.5 관리 서비스

관리 응용은 관리 응용을 개발할 수 있도록

제공되는 관리 인터페이스와 실제 관리 응용으로 나눌 수 있다. 관리 인터페이스는 사건 관리, 기록, 객체 관리 등을 수행하고, 관리 응용은 통신망 상의 어느 노드에 대한 구성, 컴퓨팅 자원의 처리 등을 실제로 수행하는 것이다.

### 5. 분산 시스템 관리 기술의 동향

이제까지 망관리 중심으로 수행되어 오던 관리 형태가 하나의 프레임워크 내로 수용되면서 시스템 관리 기능을 보완하여 분산 시스템 관리에 이용되어 가는 움직임이 나타나고 있다. 이 장에서는 이러한 분산 시스템 관리 기술의 동향을 표준화 동향, 회사들의 제품 동향, 그리고 연구 활동 분야를 살펴봄으로써 정리한다.

#### 5.1 표준화 동향

생산자들과 사용자들은 통신망과 시스템 관리의 응용들을 통합하는 표준화 방안을 정의하고

이의 표준화를 위하여 노력하고 있다.

IETF와 ISO는 이제까지 사용되어 오던 통신 관리 규약인 SNMP와 CMIP의 기능을 강화하는 형태의 움직임을 보이고 있다. IETF는 SNMP 기술을 표준화하던 그룹이 관리 프레임워크 활동에 합류하였다. SNMP를 PC, 서버, UNIX 주전산기들로 확장하였으며, SNMPv2는 주전산기와 네트워크를 동시에 감시할 수 있는 다중 대리인(multiple agent)을 지원하고 있다. 현재 DBMS를 관리하기 위한 MIB를 개발하고 있다.

X/Open은 근거리 통신망과 인터넷워크의 UNIX 호스트들의 관리를 포함하여, 특히 시스템 관리를 위한 API를 작성하고 있다. 이는 SNMP와 CMIP을 지원하는 응용을 만드는 데 사용되는 XMP API에 추가될 것이다. XMP API는 Bull, IBM 등의 제품에 포함되고 있다. 그리고 ISO/IEC 7498-4와 ISO/IEC 10040에 기반을 둔 시스템 관리를 위한 참조 모델의 스냅샷을 발표하였다.

ITU는 TMN(Telecommunications Management Network)의 표준화 활동을 수행하고 있다. TMN은 전기 통신 서비스의 관리를 위한 표준 구조를 정의한다. TMN은 "프로토콜과 메시지를 포함한 표준화된 인터페이스를 갖는 구조를 사용하여 운용 시스템과 전기통신 장비 사이에 관리 정보를 교환하기 위한 상호 연결을 할 수 있는 구조"를 제공한다. TMN은 여러 회사의 제품이 운용되는 환경에서 전기 통신망의 운용, 관리, 유지, 분할 등을 수행한다.

OMG(Object Management Group)는 분산 관리 응용들을 단순화 시키고, 개발자들의 생산성을 높일 수 있는 객체 지향형 기술을 표준화하고 있다. OMG에서는 다양한 응용들에서 오는 데이터 요구를 받아들여 정보를 검색하는 방식과 일치시키는 Object Request Broker(ORB)의 API를 작성하였다. 관리 정보를 표현하는 특정한 객체를 정의하는 작업을 수행하고 있다.

COSE(Common Open Software Environment)는 기존의 기술을 기본으로 하여 다른 UNIX 플랫폼에서 동작할 수 있도록 회사의 제품에 종속되지 않게 수정하여 규격화한다. 사용자 그룹 보안 관리, 인쇄 스프링, 백업과 재저장,

소프트웨어 면허 관리, 소프트웨어 분배 등을 위한 API를 작성하고 있다. 이 서비스들을 OMG의 CORBA(Common Object Request Broker Architecture)를 기초로 한 request broker에서 객체들로 나타낼 수 있도록 하는 방법을 정의하고 있다. COSE는 시스템 관리와 응용에 사용되는 UNIX 기본 유틸리티들의 차이점을 줄이는 노력을 하고 있다.

OSF(Open Software Foundation)은 DCE(Distributed Computing Environment)와 DME(Distributed Management Environment)에 관한 작업을 수행하고 있다. DME 작업을 지원하는 회사들간의 의견 불일치로 인하여 계속 출하 시기를 미루어 왔으며 자체의 ORB의 개발을 포기하였다. 대신 IRB(Instrumentation Request Broker)라는 CORBA-compliant ORB를 사용할 것이다.

CMG(Computer Measurement Group Inc.)는 분산 시스템 관리, 특히 성능 관리와 capacity planning에 관한 회의를 개최하고 있다.

NMF는 시스템과 통신망 관리를 위한 지침을 개발한다. 객체지향 프레임워크에서 사용될 수 있는 여러가지 네트워크 구성 요소들을 정의하고 있다. 대부분 CMIP을 사용하고 있으며, OME(Object Management Edge)라는 API들을 OSF의 IRB와 결합하고자 하고 있다. 이 작업의 결과 메인프레임과 기존의 시스템들을 객체 지향의 프레임워크의 제어 아래에 둘 수 있게 한다.

원래 통신망 관리를 위하여 제안된 SNMP는 시스템 관리 분야로 확장되고 있다. desktop에서 주요 표준은 DMTF(Desktop Management Task Force)에서 만든다. 여기에서는 주로 desktop 환경의 프레임워크 부호(code)와 규격을 만든다.

## 5.2 분산 시스템 관리 제품 및 연구 동향

앞에서 언급한 여러 표준화 기관에서 프레임워크를 정하려고 노력하고 있지만, 현재는 각 회사들의 고유한 프레임워크가 활용되고 있다. 각 시는 자기의 프레임워크 내에 각 구성 요소 제품들을 통합하여 사용자에게 제공한다.

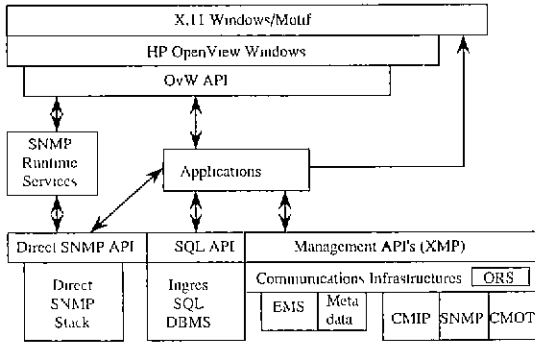


그림 8 HP Open View

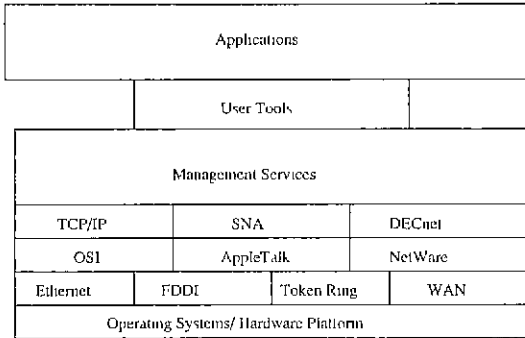


그림 9 SunNet Manager

5.2.1 HP의 OpenView[8]

HP사가 통합된 관리 환경의 플랫폼으로 제시하고 있다. 원래 네트워크 관리를 위해 만들어

졌으나, 후에 다수 공급자 분산 컴퓨팅 환경의 통신망 관리와 시스템 관리를 통합한 도구로 사용되고 있다. OpenView의 통신망 관리 기술의 일부는 OSF/DME에 수용되었다. 그림은 HP OpenView DM(Distributed Management) 플랫폼을 나타내고 있다. 그림에서 알 수 있듯이 객체 정의, 관리통신 규약, 응용 프로그램 인터페이스를 위해 표준을 사용하고 있다.

5.2.2 IBM[8]

IBM은 관리를 business, change, configuration, operation, performance 그리고 problem의 6가지로 나누어 각 부분에 대한 해결책을 제시하고 있다. 통신망 관리는 NetView/6000을 통하여 제공하고 시스템 관리는 SystemView를 사용한다.

5.2.3 Sun[8]

Sun사는 분산 환경으로 자사의 ONC(Open Network Computing)을 제안하고 있다. 그리고 워크스테이션들을 관리하기 위하여 SunNet Manager를 발표하고 있다. SunNet Manager의 구조는 그림 9와 같다.

5.2.4 Tivoli[8]

Tivoli사는 TME(Tivoli Management Enviro-

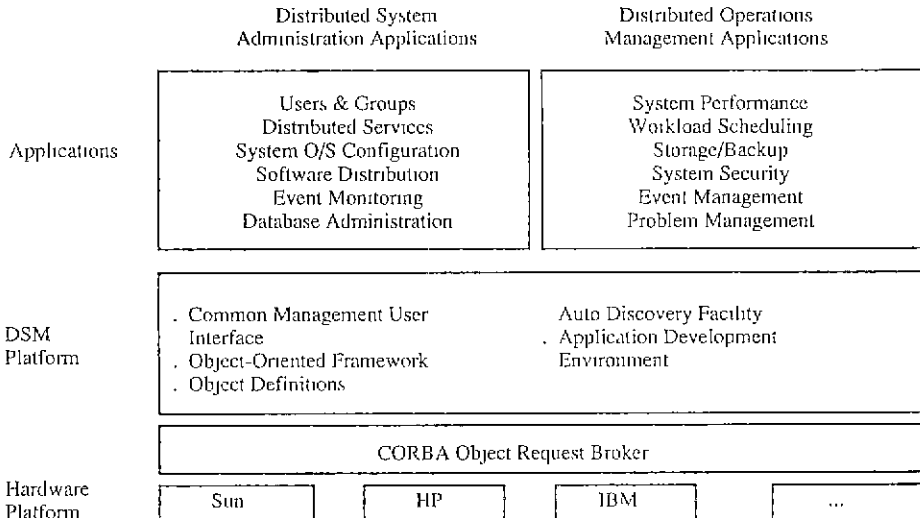


그림 10 TME의 구조와 기능

ment)를 통하여 시스템 관리를 수행한다. TME는 분산 객체 지향의 개발 환경으로 GUI 기반이며, 프레임워크와 설치 기능 및 시스템의 다른 기능들을 관리하는 부분, 소프트웨어 분배 기능, 시스템 감시와 보안 기능, 화일 시스템의 구성과 변경 관리 기능 등을 포함하고 있으며 응용 개발 환경을 제공하고 있다. OSF/DME의 한 기술로 수용되었으며, X/Open에 API가 제출되어 있다. 현재 CORBA 규격에 서비스를 맞추어 가는 단계에 있다. 그림 10은 TME의 구조와 기능을 나타낸다.

**5.2.5 OSF의 DME[9]**

OSF의 DME는 개방 환경에서 이기종의 기기들로 연결된 분산 환경을 관리하기 위한 프레임워크이다. 1993년 11월에 인쇄 서비스, 사전 서비스, 서브시스템 관리 서비스, 소프트웨어 분배 서비스, 허가 관리 서비스들을 포함한 분산 서비스를 DCE/DS Release 1.0이라는 이름으로 발표한 후, 1994년 6월에는 SNMP와 CMIP을 지원할 수 있는 구조를 갖는 NMO(Network Ma-

nagement Option)를 발표하였다. 이전에 주목 받고 있던 것보다는 참여 기업들간의 문제로 인하여 개발 범위나 기간 등이 변경되어 분산 시스템 관리 시장에서 그 영향력을 잃고 있는 것으로 보인다. 그러나 NMO의 구조는 NMF의 OMNIPoint를 구현한 모델로 주목을 받고 있다. 그림 11은 DME의 구조이다. OSF DME는 분산 서비스, 네트워크 부분, 객체 관리 부분으로 구성되는 데, 객체 관리부의 개발은 OMG의 CORBA 규격 개발과 연계되어 수행되고 있다.

**5.2.6 영국 BT 연구소의 프레임워크[10]**

영국의 BT 연구소에서는 새로운 관리 서비스에 대한 비용을 절감하고, 새로운 서비스를 빨리 제공하며, 통합된 관리 기능과 융통성 있는 서비스를 제공할 목적으로 관리 플랫폼을 개발하고 있다. 여기에 적용되는 기술은 개방 시스템, 분산 처리 기술, 공통 기술로 여러 회사들을 지원할 수 있고, 고객-서버 모형을 가지며 객체 지향의 설계 기술 등이다. 이러한 기술들과 POSIX, SQL/RDBMS, NMF/OMNIPoint, Internet RFC,

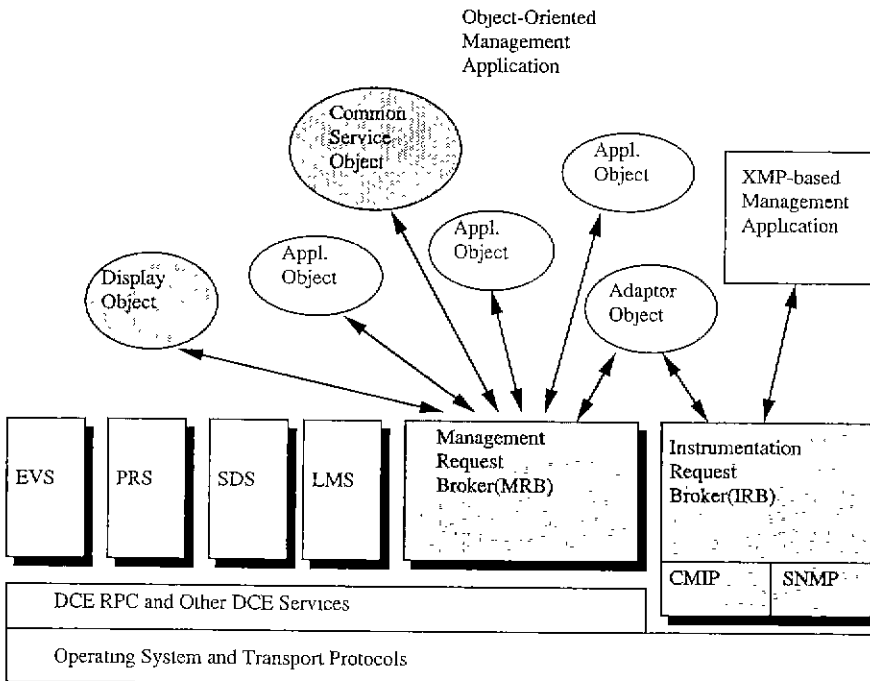


그림 11 DME의 구조

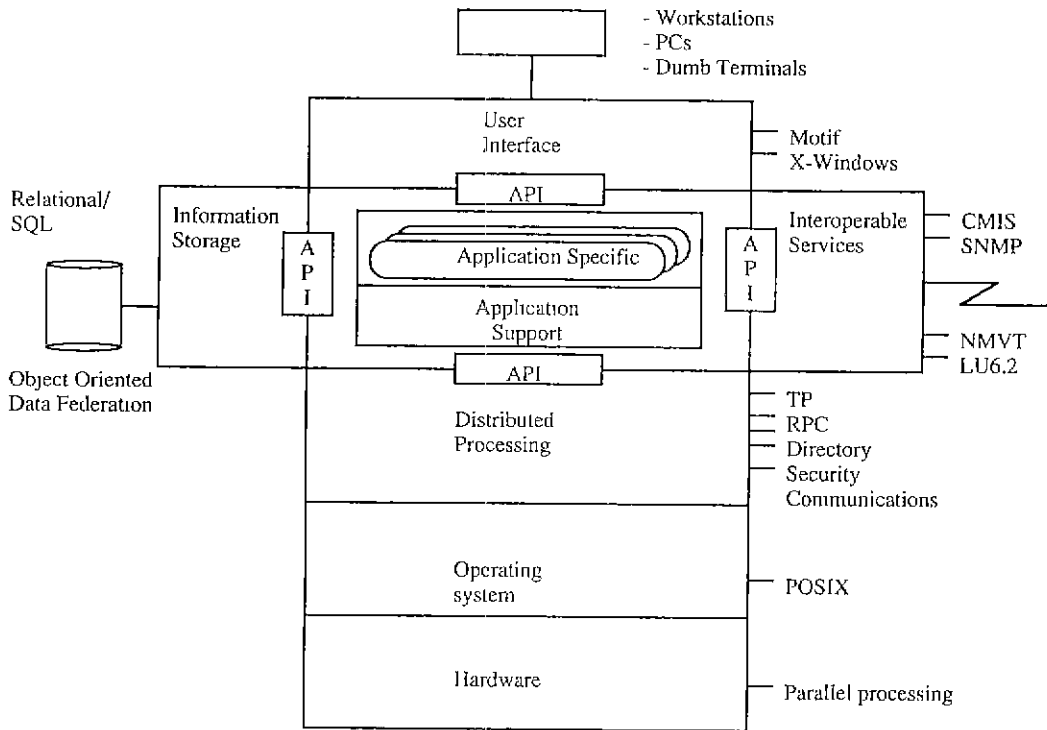


그림 12 BT의 관리 구조

OSF/DCE, DME, OMG/CORBA, X/Open, ITU/TMN 등의 표준 동향을 결합하여 그림과 같은 프레임워크를 제안하고 있다. 그러나 프레임워크 내의 많은 부분들은 컴퓨터 회사로부터 구매하여 구성하며 이 프레임워크를 이용하여 사용자 인터페이스 서비스, 정보 저장 서비스, 상호 운용 서비스, 분산 처리 서비스, 백업/재저장, 사건 통지 등의 지원과 부가 가치 서비스의 창출 등의 응용 개발을 목표로 한다.

**5.2.7 프랑스 Bull S.A.의 ISM (Integrated System Management)[11]**

System Management Language인 “SML”을 만들어 관리 응용 환경에서 요구되는 공통 서버와 서비스에 대한 일관된 인터페이스 형식을 제공하고 있다. Bull의 ISM은 CMIS를 기본으로 하여 응용들은 ISO/OSI 객체로 기술된 관리 정보를 다룬다. 모든 관리 정보와 관리 서비스에 대한 접근은 SML/CMIS를 통하여 이루어진다. 그림은 ISM의 관리 모델을 나타낸다. 주요 기

술은 Agent integrator로 ISM 정보 모델로 변환하여 agent가 가진 프로토콜과 통신을 지원하기 위한 것이다. 이 구조로 새로운 객체를 추가하여 새로운 서비스, 응용들을 확장해 갈 수 있도록 하였으며 SNMP 대리인, OSI 대리인 개발을 용이하게 하기 위한 개발 도구들도 제공한다.

**5.2.8 현재 시장의 동향**

가트너 그룹에서 조사한 분산 시스템 관리 시장의 최근 동향은 다음과 같다.

분산 시스템 관리 제품을 개발, 판매하는 회사 사이에는 지난 18개월 동안 회사 합병과 같은 많은 변화가 있었다. 그 배경은 개방되고 이기종인 분산 컴퓨팅 환경의 관리를 목적으로 하는 컴퓨팅 플랫폼 때문이었다. 93년 초에 AT&T/NCR (현재는 AT&T Global Information Solutions), DEC, HP, IBM, NetLabs, Siemens Nixdorf Information Systems, Sun 등이 각 사 고유의 제품을 가지고 있었다. 그리고 이러한 제

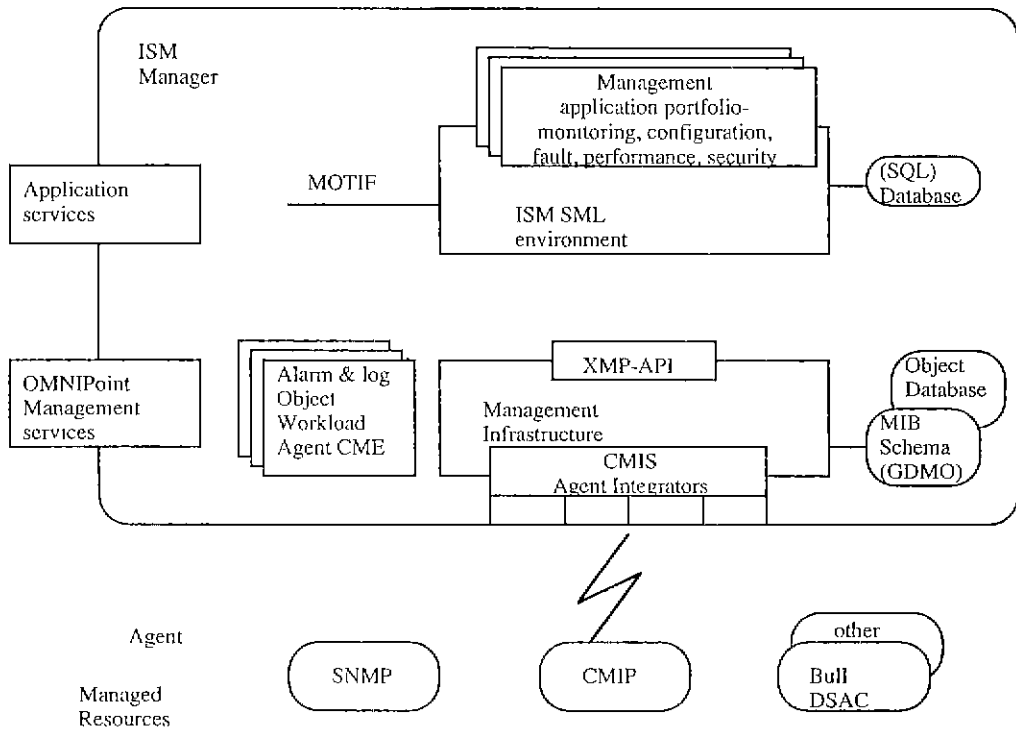


그림 13 Bull S.A.의 관리 구조

품이 전체 시장의 약 80%를 차지하고 있었다. 그러나 90년 여름에 Siemens가 NetLabs으로부터 Dimons를 이용함으로써 시장의 주도권 싸움에서 빠지고, 93년 4월에 AT&T도 같은 방식으로 떨어져 나갔다. 93년 8월에 DEC가 IBM의 NetView/6000 프레임워크를 자사 제품에 채용할 것을 결정하면서 경쟁 회사는 HP, IBM, NetLabs 그리고 Sun 만이 남게 되었다. 그러나 IBM의 NetView/6000은 핵심 부분을 HP의 OpenView를 채용함으로써 독자적이 아니었다. 93년 12월에 Sun이 NetLabs사의 기술을 자사의 SunNet Manager와 결합한 제품을 내놓으면서 제도약의 기회로 삼았다. 94년 6월에는 HP와 NetLabs, AT&T와 HP가 기술 협력 관계를 맺고 IBM도 더 이상 HP의 OpenView kernel을 사용하지 않기로 결정하였다. 1995년에는 플랫폼의 80% 정도는 HP, IBM 그리고 Sun이 차지할 것으로 가트너 그룹은 예측하고 있다[12].

5.2.9 앞으로의 발전 동향

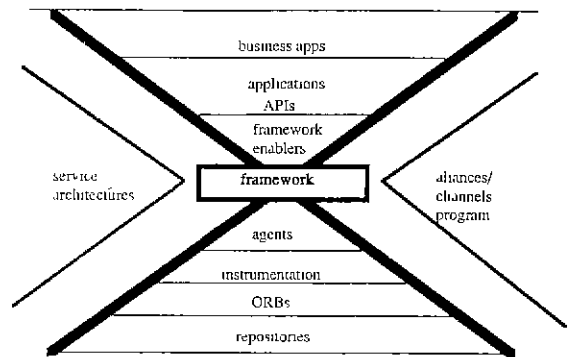
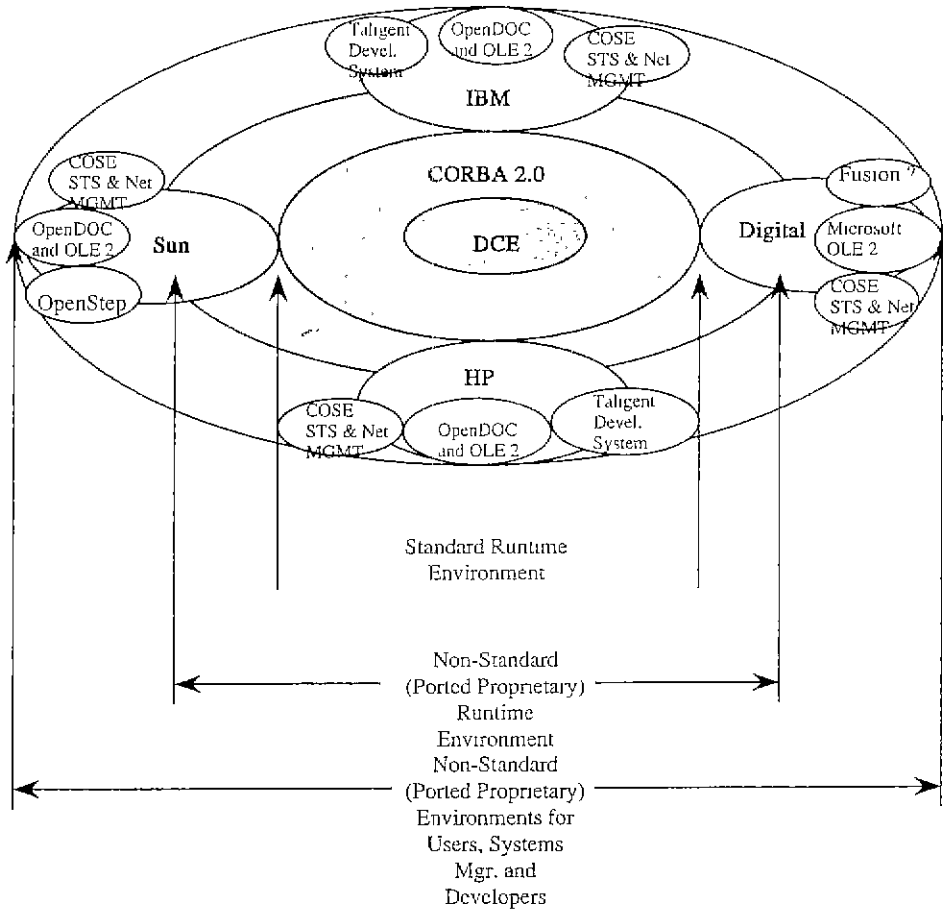


그림 14 프레임워크를 중심으로 한 기술 발전 (Source: Gartner Group)

현재는 프레임워크를 구성하는 것에 집중되어 있으나, 앞으로는 그림에서 알 수 있듯이 프레임워크를 기본으로 business 응용 분야 및 도구의 개발을 추진하고, 미들웨어 회사들은 미들웨어와 API를 연결하여 도구들의 상호 연결을 가능하게 할 것이다. 한편 다른 분야에서는 관리대상 객체에 관심을 두고 새로운 대리인을 만들어 나



(Source : Gartner Group)

그림 15 객체 지향 기법을 이용한 관리 구조의 발전

갈 것으로 예측된다. 데이터베이스 회사들은 저장 장소(repository)에 관심을 가지고 있다. 그리고 분산된 자원의 관리는 더욱 객체 지향적인 모델을 요구하므로 ORB의 개발에 초점이 모아지고 있다. 또한 자동 관리의 운용을 위하여 지능형 대리인(intelligent agent)를 만들어 가는 움직임도 있다.

통신망과 시스템의 관리 자원들을 공통 형식으로 표시하기 위하여 객체를 이용한 기법이 도입되고 있으며, 다음 그림은 객체 개념을 관리에 응용해 가는 움직임을 주요 기업의 활동을 중심으로 나타낸 것이다. 현재 만들어지고 있는 표준이나 기술들은 분산 시스템 관리에 영향을 미칠 것이다.

## 6. 결 론

이제까지 분산 환경에서 각 자원들을 관리하기 위한 기술들과 발전 동향을 살펴보았다. 분산 시스템의 관리는 자원의 분산 처리로 인하여 일어나는 여러가지 문제점들을 해결할 수 있어야 하며, 여러 다른 회사의 제품들 및 기술들을 하나의 프레임워크 내에서 관리할 수 있어야 한다.

분산 시스템 관리는 감시, 운용 등을 하는 분야와 유지 보수, 변경, 계획을 하는 부분 그리고 응용 개발에 대한 관리 부분 등이 통합되어 가면서 어느 플랫폼을 구성하는 방향으로 발전되어 가고 있다. 분산 시스템 관리에 프레임워크를 사용하는 것은 비용의 절감을 가져오고 다양한

서비스에 대한 확장을 가능하게 한다.

현재 주요 컴퓨터 회사들은 분산 시스템 관리 시장을 선점하기 위하여 노력하고 있으며 개방형 분산 처리를 위한 모델을 개발하고 있다. 다운 사이징, 고객-서버 모델, 분산 처리 등의 현상이 나타나는 가운데 분산 시스템 관리도 네트워크 관리와 시스템 관리를 통합한 형태로 발전하여 서비스 관리라는 새로운 개념을 만들고 있다.

이제까지 산재되어 있는 기술들을 하나의 프레임워크 내에서 제공하여 이기종 기기로 구성되는 분산 환경의 관리에 적용할 수 있어야 한다. 그리고 객체 지향의 기술을 적용하여 보다 편리한 관리 환경을 구성해 가는 등 정보 기술의 새로운 변화와 더불어 발전시켜 나가야 한다.

### 참고문헌

[1] 김 명준, Serge Miranda, 고객-서버(Client-Server) 구조의 분류, 개방형컴퓨터통신연구회, 개방시스템, 제 8권 제 4호, pp. 20~32, 1994. 8.

[2] Amjad Umar, Distributed Computing, pp. 12~15, Prentice-Hall, Englewood Cliffs, 1993.

[3] Raman Khanna, Distributed Computing, Implementation and Management Strategies, pp. 107~119, Prentice-Hall, Englewood Cliffs, 1994.

[4] Alwyn Langsford, Jonathan D. Moffet, Distributed Systems Management, pp. 93-97, Addison Wesley Publishers Ltd., Workingham, 1993.

[5] Mary Jander, "Management Frameworks," Data Communications, pp. 58~68, 1994.2.

[6] Morris Sloman, Network and Distributed Systems Management, pp. 526~534, Addison-Wesley Publishers Ltd., Workingham, 1994.

[7] NMF, Discovering OMNIPoint, pp. 19~24, Prentice-Hall, 1993.

[8] Datapro, "System Management: An Overview," DATAPRO, pp. 101~108, 1993.9.

[9] Lance Travis, "Standards Based Distributed Management Environment," pp. 226~252, UNIFORM 94, 1994. 3.

[10] Tony Richardson, "Strategy for Integrated Management Systems and Platform support," pp. 544~554, Proceedings of NOMS, 1994. 2.

[11] Paul Miller, "Integrated System Management," pp. 555~564, Proceedings of NOMS, 1994. 2.

[12] H. Walter Johnson, "Networked Systems Management: Organizational Implications," pp. 11~15, Gartner Group The fifth Korea Briefing, Seoul, 1994. 5.

### 김 명 준



1978 서울대학교 계산통계학과 학사  
 1980 한국과학기술원 전산학과 석사  
 1986 프랑스 Nancy 제 1대학교 응용수학 및 전산학과 박사  
 1980 ~ 1981 아주대학교 종합연구소 연구원  
 1981 ~ 1986 프랑스 낭시 전산학 연구소(CRIN) 연구원  
 1986 ~ 현재 한국전자통신연구소, 책임연구원  
 관심분야: 분산 시스템, 데이터베이스, 소프트웨어 공학

### 김 경 범



1981 인하대학교 전자공학과 학사  
 1983 인하대학교 전자공학과 석사  
 1983 ~ 현재 한국전자통신연구소, 책임연구원 전자계산기 기술사  
 관심 분야: 분사시스템, 컴퓨터 통신