

망 간 연동을 위한 연결관리 연합기능의 설계 및 구현

이 한 영[†] · 임 경 준^{††} · 서 동 선^{†††}

요 약

차세대 정보통신의 관리구조가 분산처리 환경 및 객체지향 개념을 적용하여 정립되어가고 있다. 본 고에서는 전달망에 대한 연결 서비스를 제공하는 연결관리 기능에 망 간 연동을 위한 단계적 연합기능을 설계 및 구현하고, 상이한 분산환경에서 개발된 이러한 서비스 객체 간의 상호운용성을 시험하기 위하여 유사 트레이더와 게이트웨이 기능을 제작하였다. 이러한 기술들은 망관리가 다양한 환경에서 점차 통합되어가는 통신망 관리기능에 망 간 연동을 지원하는 실용적인 기술이 될 것이다.

Design and Implementation of Federation of Connection management for Interworking

Han Young Lee[†] · Kyeong Jun Lim^{††} · Dong Sun Seo^{†††}

ABSTRACT

Object-oriented and distributed processing methods are adopted to develop next-generation telecommunications management architecture. In this paper, we design and implement gradual federation function on connection management system providing connection services to transport network for interworking and encapsulated trader and gateway function for testing interoperability of these service objects between distributed processing environments. These techniques are practically applicable to support interworking between heterogeneous management network systems according to a tendency of integrating telecommunications management systems.

1. 서 론

오늘날 시장규모가 연간 천억달러에 달하는 정보통신망은 다양한 산업체들이 참여하고 있어 그만큼 이질적인 통신망 구성요소가 공존하고 통신망 환경도 점점 대형화, 고도화되고 있으므로, 국제간 또는 상이한 사업자망 간에 상호운용성이 강조되고 다양한 통신망을 관리할 수 있는 새롭고도 일관적인 통신망 관리체계의 필요성이 대두되고 있다. 이에 따라

광대역 정보통신망에 ITU-T M.3010의 TMN 기능구조 및 클라이언트/서버 개념과 ITU-T G.803의 SDH 계층화와 분할화 개념, ISO의 관리영역 분할개념 그리고 ITU-T X.901 RM-ODP의 분산환경 개념을 적용하여 통신망관리구조를 모델링하는 연구가 여러 기구에서 진행되고 있다. [1, 2, 3] 본 고에서는 이 구조의 관리기능 영역인 구성, 장애, 성능, 보안, 과금 관리기능[4, 9] 중에서 망 간 상호운용성을 지원하기 위하여 구성관리 기능 중의 연결관리 기능에 망 간 연합(federation)기능을 구현하였다. 본 고의 구성은 다음과 같다. 연합에 대한 일반적 개념과 연결관리의 연합기능[5]에 대하여 2절에서 논하고, 3절에서는 망 관리 모델[6]에 대하여 요약한다. 4절에서는 연동 시

[†] 정 회 원: 한국통신 통신망연구소

^{††} 준 회 원: 한국통신 통신망연구소

^{†††} 정 회 원: 명지대학교 전자공학과

논문접수: 1996년 2월 8일, 심사완료: 1996년 4월 17일

나리오 및 연합기능을 설계하고 5절에서는 구현[7, 10]에 대하여 설명한다. 6절에서는 이에 대한 시험내역[8]이 간단히 논의되고, 마지막으로 7절에서 결론을 맺는다.

2. 연합기능

2.1 기본개념 및 요구사항

정보통신망을 하나의 거대한 개방형 분산 시스템이라고 볼 때 상이한 공급자로부터의 상이한 시스템들의 자존적 특성을 보존해 주면서도 서로의 연동을 가능케 하는 어떤 구성이 필요하게 되고 더불어 증설, 철거에 따르는 구성변경도 용이하게 되어야 한다. 이런 경우에 상하 관계의 시스템 체계에서는 어떠한 재구성해야 하지만 연합이라는 개념을 도입하면 시스템 간 상호 합의에 의하여 편리하게 대처할 수 있다. 정보통신망에서의 연합에 관한 기본 개념은 자원의 공유와 상호작용 원리의 상호 합의에 근거하여 객체들이 자존적으로 공존하는 분산 시스템의 조직적 구조를 설명하는 것이다. 즉, 연합이란 각자의 제어 및 관리를 포기하지 않고 동등 계층간에 자원, 정보, 능력 등을 공유, 협조하는 조직적인 형태의 하나로 볼 수 있다. 다중 영역간의 장시간 연합에서는 연합의 개시, 종료절차, 연합영역 간의 상이점을 갖추는 연합 투명성, 연합용 객체를 다루는 연합조정 등이 우선적으로 고려되어야 한다.

연합이 이루어질 시스템의 요구사항으로는 광대역에 걸친 분산제어, 구조변경이 가능한 신속성과 발전성, 상이한 구성 요소간의 결합가능 및 변경 영향의 최소화 등을 들 수 있으며, 이러한 요구사항을 만족시키기 위한 시스템 구성요소간의 상호작용원칙은 각 구성 요소의 자존성, 상호작용, 협정 등의 동을 필요로 한다. 이러한 원리가 적용된 좀더 강화된 연합 기능을 구축할 때 필요한 추가 요구사항으로는 좀더 조직화된 연합 영역, 연합용 자원과 국부용 자원의 구별, 각 구조간의 영역경계, 경계에서의 정보 변화, 연합 투명성 등을 들 수 있다.

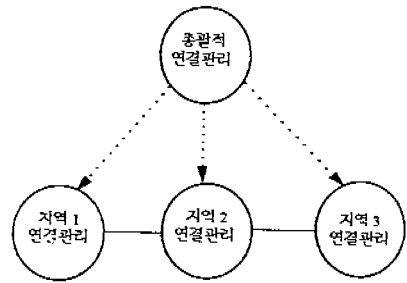
2.2 연결관리의 연합기능

연결관리의 연합기능은 하나의 연결이 여러 영역에 걸쳐 성립될 때 적용된다. 이 연합기능은 앞서 논

한 연합기능 및 요구사항과 일관성이 있어야 한다. 이 기능은 다음 2가지 경우로 모델링된다.

2.2.1 결합적 연합 모델링

여러 지역을 연결해야 하는 하나의 연결 요구가 지역 1 연결관리로 들어오면 이 요구를 전 지역의 연결 정보를 가지고 있는 총괄적 연결관리에게 보내고 총괄적 연결관리는 필요한 지역을 선정하여 해당지역의 연결관리에게 연결지시를 내린다. 이 지시를 접수한 각 지역 연결관리는 각자의 연결을 수행하여 전체적 연결을 구성한다.



(그림 1) 결합적 연합 모델링
(Fig. 1) Combined federation modelling

2.2.2 단계적 연합 모델링

여러 지역을 연결해야 하는 하나의 연결요구가 지역 1 연결관리로 들어오면 지역 1 연결관리는 자신의 연결을 수행하고 나머지 연결부분은 옆에 있는 지역 2 연결관리에게 요청한다. 다음의 연결관리도 동일한 기능을 수행하여 전체의 연결이 완료될 때까지 계속된다. 관리영역 간의 연합기능은 단계적 연합기능으로 모델링될 수 있다.



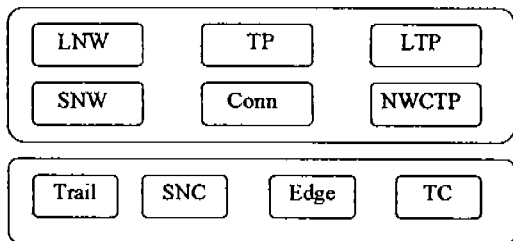
(그림 2) 단계적 연합 모델링
(Fig. 2) Gradual federation modelling

3. 망 관리구조 모델링

여기서는 기존의 관리기능 영역인 구성, 장애, 성능, 보안, 과금관리 기능 중에서 본 고의 범위에 해당되는 구성관리 기능, 즉 자원형상 및 연결관리 기능을 그 모델링 결과인 정보객체와 연산객체를 이용하여 설명하였다.

3.1 정보 모델링

하나의 망은 여러가지 형태의 정보를 전달할 수 있는데 망을 관리영역에 따라 분할시켜 볼 때, 각 형태별 정보는 계층화된 계층망(LNW: Layer Network)을 따라 전달되며 계층망 간은 클라이언트/서버 관계를 유지한다. 하나의 계층망은 분할개념을 적용하여 여러 개의 부분망(SNW: Subnetwork)으로 구성될 수 있다. 부분망은 관리영역 별로 나누어지며, 위상링크로서 서로 연결된다. 최소단위의 부분망은 하나의 물리적 노드(스위치)까지 적용될 수 있다. 망을 연결관점에서 보면 정적인 형상을 표현하는 위상링크(TL: Topological Link)와 연결(Connection) 객체, 그리고 연결요청에 의한 동적연결을 표현하는 연결관리용 정보객체로 트레일(trail), 부분망연결(SNC: Subnetwork Connection), 탄뎀연결(TC: Tandem Connection), 에지(edge) 객체가 있다. 망의 여러가지 종단점으로는 망종단점(NWTP: Network Termination Point), 이들의 묶음인 망종단점풀(NWTPPool), 정적인 위상링크의 종단점인 링크종단점(LTP: Link Termination Point) 객체로 구성되며, 망종단점은 다시 정적인 연결객체의 종단점인 망연결종단점(NWCTP: Network Connection TP)과 동적인 트레일객체의 종단점인 망트레

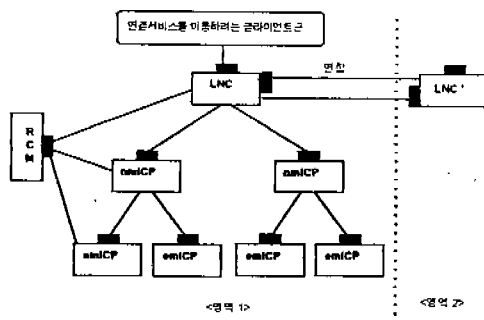


(그림 3) 정보객체
(Fig. 3) Informational objects

일종단점(NWTP: Network Trail TP)으로 구성된다. 트레일은 한 계층망 종단점 간의 연결성을 의미하는데, ATM 계층망의 경우 VPC나 VCC의 연결성을 뜻하며, 트레일의 종단점은 OSI의 NSAP(Network Service Access Point)와 대응된다. 부분망연결은 부분망연결종단점 간의 연결성을 의미한다. 에지는 트레일과 망트레일종단점, 부분망연결과 부분망연결종단점을 결합시키고, 루트에지(Root edge)와 리프에지(Leaf edge)를 구분하여 트레일과 부분망연결에 방향성을 부여한다. (그림 3)에 정보객체들을 정리하였다.

3.2 연산 모델링

연산 모델링은 앞에서 정의한 자원구성 및 연결성 정보객체를 관리 및 제어하는 연산객체를 정의하는 것이다. 이 연산객체들은 (그림 4)와 같이 TMN의 계층구조와 정보객체에 따라 정의된다.



(그림 4) 연산객체
(Fig. 4) Computational objects

3.2.1 LNC(Layer Network Coordinator)

LNC는 연결서비스를 이용하고자 하는 상위의 모든 클라이언트가 접속하는 첫번째 연산객체로 계층망 양단 간의 연결을 나타내는 트레일의 생성, 유지, 복구를 담당한다. LNC가 관리하는 정보객체로는 트레일, 부분망연결, 에지이며 수행할 주요기능은 대역폭산출, 라우팅, RCM으로 발착신 NWTP 생성, 삭제요청, 트레일, 루트에지 및 리프에지 생성, 루트에지를 발신 NWTP와 바인딩, 리프에지를 착신 NWTP와 바인딩, 라우팅정보에 따른 nmlCP 선택 및 부분망연결 설정요청, nmlCP에게 에지의 수정,결합,분리요

청, 생성된 트레일, 부분망연결, 루트에지, 리프에지의 관리 등이다.

3.2.2 CP(Connection Performer)

CP는 분할화에 따라 구성된 부분망의 종류별로 서로 상이한 CP 연산객체(nmlCP, emlCP)가 인스턴스화되어 이에 대한 부분망연결을 설정,유지,복구한다. CP가 관리하는 정보객체로는 부분망연결, 에지, 탄뎀연결이며 nmlCP가 제공할 주요 기능은 라우팅, 부분망연결, 루트 및 리프 에지 생성, RCM으로 발착신 NWCTP 생성 및 삭제 요청, 라우팅 결과로 탄뎀연결 생성, 하위CP를 선택하여 부분망연결객체 생성,수정,삭제요청, 하위 CP에게 수정,결합,분리요청, 생성된 부분망연결, 루트에지, 리프에지의 관리 등이다.

emlCP가 제공할 주요 기능은 부분망연결, 루트 및 리프 에지 생성, RCM으로 Source 및 Sink NWCTP 생성 및 삭제 요청, 에지의 수정,결합,분리요청, 망요소연결 지시, 생성된 부분망연결, 루트에지, 리프에지의 관리 등이다.

3.2.3 RCM (Resource Configuration Manager)

RCM은 망의 형상과 상태정보를 관리하는데 주로 LNC와 CP로부터 관련정보를 요청받아 결과를 응신한다. RCM이 관리하는 정보객체로는 계층망, 부분망, 위상링크, 연결, 링크종단점, 망연결종단점 이며, 제공할 주요 기능은 LNC의 트레일구성에 필요한 망트레일종단점 생성 할당 및 해제, CP의 연결객체 설정시 망연결종단점 생성 할당 및 해제, 부분망구성정보 검색 등이다.

4. 시나리오 및 설계

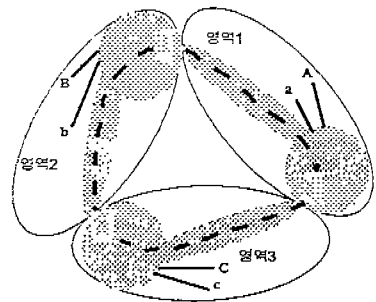
4.1 시나리오

먼저 다음 사항을 가정하자. 관리영역이 서로 다른 3개 통신사업자에 속한 이용자들이 다량의 화상정보를 교환하기 위하여 상호 연결된 채널의 용량 확장이 필요하다. 이용자들의 이러한 요구사항을 각 운용자들은 연결관리의 연합기능을 사용하여 상호운용함으로써 만족시켜준다. 3개의 사업자는 영역1, 영역2, 영역3, 각 망의 운용자는 a, b, c, 이용자는 A, B, C,라 하자. 이후의 절차는 아래와 같다.

- 1)A가 B와 C에게 기설정 채널로 다량의 화상정보를 송신하려 한다.
- 2)채널용량이 부족하므로 A가 a에게 요청하면, a는 b, c와 용량확장에 대한 협의를 하고 운용자들은 각자의 연결관리 시스템으로 용량확장을 지시한다.
- 3)확장된 용량으로 서로 간에 연결이 되면 A는 B, C에 화상정보를 송신한다.

4.2 논리적 망 구성

논리적 망구성은 (그림 5)에서 보듯이 사업자별로 논리적 계층망과 부분망을 1개씩 구성하고 각 경계상에 1개씩의 증계선적 위상링크를 갖도록 설계하였다.



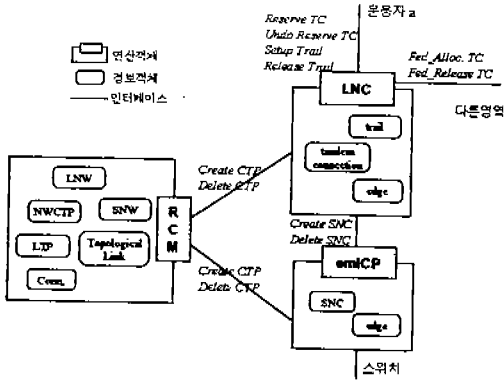
(그림 5) 논리망 구성도 (Fig. 5) Logical network topology

4.3 영역1 설계

이 영역의 스위치는 하나이므로 부분망이 하나로 설계되어 nmlCP가 불필요하므로 연산객체는 LNC, emlCP, RCM이다. 그리고 한 연산객체에 다수의 정보객체를 포함시켜 객체의 수를 줄임으로써 컴퓨팅 부하를 감소시켰다. 또한 다른 영역과의 상호운용을 위한 연결관리 정보를 교환하기 위하여 연합기능을 설계하였는데 이 기능은 2절에서 언급한 단계적 연합방식을 채택하여 LNC의 운용 오퍼레이션 서비스 중의 하나로 구현되었다. (그림 6)에 이 영역의 연결관리 시스템을 구성하는 객체들을 나타내었다.

4.3.1 LNC

LNC가 관리하는 정보객체는 트레일, 탄뎀연결, 에



(그림 6) 영역1의 연결관리 객체 구성도
(Fig. 6) Connection management objects configuration of region 1

지이며 제공하는 서비스는 다음과 같다. 지역 운용자와의 인터페이스는 Reserve Tandem Connection, Undo Reserve Tandem Connection, Setup Trail, Release Trail 이고, 다른 영역 LNC와의 인터페이스(연합)은 Federation-Allocation Tandem Connection, Federation-Release Tandem Connection이다.

4.3.2 emICP

emICP가 관리하는 정보객체는 부분망연결, 에지이며 제공하는 서비스는 LNC와의 인터페이스인 Create Subnetwork, Delete Subnetwork이다.

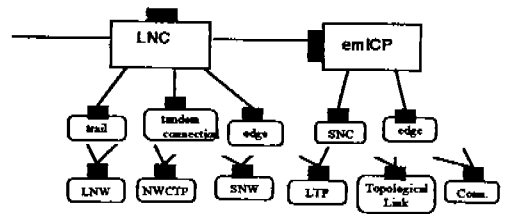
4.3.3 RCM

RCM이 관리하는 정보객체는 계층망, 부분망, 링크종단점, 망연결종단점, 위상링크, 연결객체 등이며 제공하는 서비스는 LNC 및 emICP과의 인터페이스인 Create CTP, Delete CTP이다.

4.4 영역2와 3 설계

이 관리영역에서 망 간 연합에 관련되는 스위치는 하나이므로 부분망이 하나로 설계되어 nmlCP가 불필요하고 RCM 역시 별도로 설계하지 않았으므로 연산객체는 LNC와 emICP 만이다. 또한 정보객체도 연산객체에 포함시키지 않고 독립적으로 인터페이스를 가짐으로써 전체 객체 수가 증가하여 컴퓨팅 부하를 증가시켰으나 객체의 재사용성을 향상시켰다. 따라

서 계층망, 부분망, 위상링크, 망연결종단점 등의 정보객체들을 초기 프로세스로 생성, 구동시켜 분산환경에서 동작할 수 있도록 트레이더에 등록시켜 LNC와 emICP가 사용할 수 있게 하였다.



(그림 7) 영역2의 연결관리 객체 구성도
(Fig. 7) Connection management objects configuration of region 2

5. 구현

여기서는 앞에서 설계된 연산객체들의 인터페이스를 Common Object Request Broker Architecture (CORBA)의 Interface Definition Language(IDL)로

```

//LNC의 idl //
module lnc {
interface LNC_M {
void ReserveTC (in ReserveTC_in reqIn,
out ReserveTC_out resOut);
void UndoReserveTC (in UndoReserveTC_in reqIn,
out UndoReserveTC_out resOut);
void SetupTrail (in SetupTrail_in reqIn,
out SetupTrail_out resOut);
void ReleaseTrail (in ReleaseTrail_in reqIn,
out ReleaseTrail_out resOut);
};
interface FEDERATION_M {
void FedAllocateTC (in FedAllocateTC_in reqIn,
out FedAllocateTC_out resOut);
void FedReleaseTC (in FedReleaseTC_in reqIn,
out FedReleaseTC_out resOut);
};
};

//emICP의 idl //
module emicp {
interface emi_SNC {
OutSNCInfo_t emi_createSNC (in ECreateSNCInfo_t reqIn);
boolean emi_deleteSNC (in string reqIn);
};
};

//RCM의 idl //
module rcm {
interface NWCTP {
void GetNWCTP (in reqLTP, in reqLTP_in, in Ed_BWReq, out reqLTP_out);
boolean deleteNWCTP (in reqLTP_delete, in reqLTP_delete);
};
};
    
```

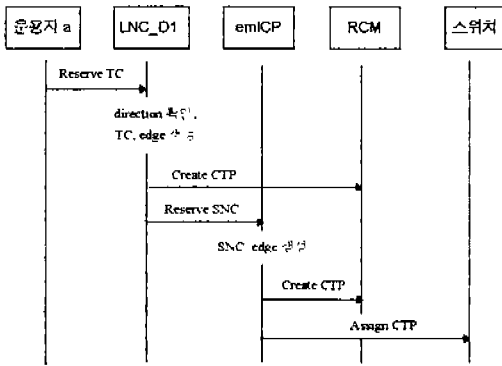
(그림 8) 연산객체들의 IDL기술
(Fig. 8) IDL description of Computational objects

기술하였고, 이에 따른 연산객체 간 연결 시나리오 및 구현환경을 간략히 설명하였다.

5.1 연합 시나리오

여기에서는 전체적인 연산객체들간의 연합 및 상호작용에 대해서 기술하기 위해 다른 영역의 LNC 및 운용자, 스위치를 표시하였다. 다음의 절차는 연결설정에 있어서 영역1이 발신인 경우를 예로 든 것이다.

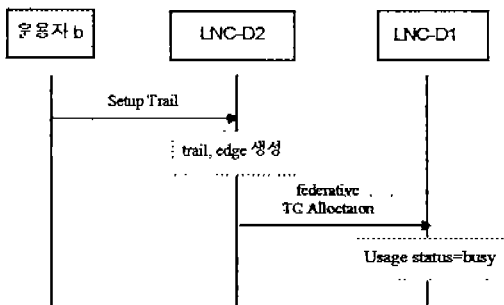
1) 운용자 a의 탄뎀연결 예약



2) 운용자 a가 탄뎀예약의 결과(방향, 탄뎀id, 에지)를 LNC-D1으로부터 받은 후 운용자 b와 연결을 협의하면서 이 정보를 함께 전달한다.

3) 운용자 b는 a의 절차와 동일하게 영역2의 탄뎀을 예약하고 그 결과를 받는다.

4) 운용자 b는 영역2의 탄뎀연결과 영역1의 탄뎀연결을 서로 연합하여 트레이일을 설정하라는 명령을 LNC-D2로 내린다. 그러면 영역2의 LNC-D2는 이 명령을 수행하면서 받은 정보에 포함되어 있는 LNC-D1과 연합을 하게 된다.



5) 영역2의 LNC-D2는 연합이 이루어진 트레이일의 설정을 완료하였다고 b에게 전달한다.

6) 연결의 해제는 이의 역순으로 이루어 진다.

5.2 구현환경

구현된 시스템의 환경은 <표 1>과 같다.

<표 1> 구현환경

<Table 1> Implementation Environment

| | 영역 1 | 영역 2 | 영역 3 |
|---------------|----------------------|----------------------|----------------------|
| LNC간 연합 | ANSAware trader + GW | ANSAware trader + GW | ANSAware trader + GW |
| 연결관리와 운용자간 통신 | DCE-RPC | DCE-RPC | DCE-RPC |
| 연결관리시스템 | DEC WS(OSF1) | DEC WS(OSF1) | SUN WS(SunOS) |
| 분산처리환경 | DEC Objectbroker | DEC Objectbroker | ODIN |
| ATM교환기 | FORE ASX-200 | FORE ASX-200 | FORE ASX-200 |

6. 시 험

영역1과 영역2의 연결관리 모듈은 <표1>에서 보듯이 DEC의 ObjectBroker (CORBA1.1 호환의 분산처리환경)에서 개발하였으나 상대영역과의 연합을 수행할 때 필요한 트레이딩 기능을 이용하기 위해 ANSAware를 별도로 사용하였다. ANSAware는 ObjectBroker와 호환성이 없으므로 ObjectBroker 상에서 수행되는 객체들이 직접 ANSAware 트레이더를 사용할 수 없어 이들을 매핑시켜주는 유사 트레이더를 제작하여 상호동작시켜 보았다. 이 유사 트레이더도 ObjectBroker 상에서 하나의 객체로 인식되므로 서로의 연합을 위하여는 상대영역의 유사 트레이더에 대한 위치를 알아야 하기에 서로의 기준 화일의 이름 및 저장소를 지정하는 등의 환경설정이 필요하였다. 또한 객체 간의 통신을 위하여 두 환경 간에 게이트웨이를 두어 서버/클라이언트 여부와 상대영역 이름 등을 주고 연결관리 시스템의 연합기능을 수행시켰다. 영역2와 영역3의 연결관리 모듈은 DEC의 ObjectBroker와 자체 개발품인 ODIN(CORBA 1.1호환)을 사용하였다. 이와 같이 서로 다른 분산처리환경을 연결하기 위하여

각자의 영역에 게이트웨이를 주어 영역2와 영역3간의 연합은 물론 영역1과 영역3과의 연합에도 사용하였다.

7. 결 론

본 고에서는 연결관리의 연합기능을 모델링하고 이중 단계적 연합방법을 연결관리 기능에 부가하여 설계, 구현 및 시험한 내용에 대하여 설명하였다. 여기서 기술된 내용은 상이한 연결관리 시스템 간의 단계적 연합을 지원하도록 LNC에 연합을 위한 트레이일 설정기능을 부여한 것과, 상이한 분산처리환경 간의 상호운용을 위한 유사 트레이더와 게이트웨이 기능을 두어 구현, 시험한 것으로써 실용적인 연구에 많은 도움을 줄 것이다. 실제로 트레이딩 서비스를 제공할 경우에는 여러가지 이름주기(naming) 시스템이 포함되며, 또한 상이한 시스템이 연합할 때는 그 정보가 포함된 이름주기 시스템도 연합하게 되는데 이름주기 시스템의 연합과 이름주기 서버의 연합 등이 있다. 이 경우에는 2절에서 언급한 강화된 연합기능이 필요할 것이다. 따라서 이 후에는 트레이딩과 이름주기와 연합기능이 복합적으로 고려되어야 할 것이다. 이러한 기술은 다양한 망 관리가 존재하는 환경에서 점차 통합되어 가는 통신망 관리기능에 망 간 상호운용성을 보장하는 실용적인 기술로서 대단히 중요한 의미를 갖는다고 생각된다.

참 고 문 헌

- [1] ITU-T M.3010, Principles for a Telecommunications Management Network, October 1992
- [2] ITU-T G.803, Architecture of transport networks based on the SDH, 1992
- [3] ITU-T X.701, Systems Management Overview for CCITT Applications, SGVII Contribution 109, April 1991
- [4] ITU-T M.3400, TMN Management Functions, December 1991
- [5] Bellcore Doc. SR-NWT-002282, INA Cycle 1 Framework Architecture, Issue 2, Bellcore, April 1993

- [6] Connection Management Architecture, TB_C3.JB.001_1.0_93, TINA-C, December 1993
- [7] James Rumbaugh, Object-Oriented Modeling and Design, Prentice Hall, Englewood Cliffs, N.J, 1991.
- [8] The Common Object Request Broker : Architecture and Specification, OMG Document Number 91.12.1 Revision 1.1 December 1991
- [9] ITU-T M.3610, Principles for Applying the TMN Concept to the Management of B-ISDN, February 1993
- [10] Grady Booch, Object Oriented Design, Benjamin /Cummings publishing, 1991.
- [11] Overall Concepts and Principles of TINA, TB _MDC.018_1.0_94, TINA-C, February 1995
- [12] Bellcore, TA NWT-001248, Generic Requirements for Operations of Broadband Switching Systems, Issue2, October 1993.



이 한 영

1980년 연세대학교 전자공학(학사)
 1984년 연세대학교 전자공학(석사)
 1985년~현재 한국통신 통신망연구소 선임연구원
 관심분야: 통신망서비스관리, 분산처리, 망간연동, 무방해감시, 객체지향

임 경 준

1991 전남대학교 전자공학(학사)
 1993 전남대학교 전자공학(석사)
 1993~현재 한국통신 통신망연구소 선임연구원
 관심분야: 분산처리환경, 통신망관리, 객체지향

서 등 선

1980년 연세대학교 전자공학(학사)
 1985년 연세대학교 전자공학(석사)
 1989년 미국 뉴멕시코대학 전기공학(박사)
 1990년~현재 명지대학교 전자공학과 교수
 관심분야: 초고속정보통신망, 객체지향 등