

# 지능망 플랫폼 시뮬레이션을 위한 간이 SSP 설계

° 송 상 철, 김 연 중, 이 지 영, 최 한 옥, 안 순 신  
고려대학교 전자공학과 컴퓨터네트워크 연구실

## Design of an SSP Emulator for the Simulation of Intelligent Network Platforms

Sangchul Song, Yun-Joong Kim, Ji-Young Lee, Han-Ok Choi, Sunshin An  
Computer Network Lab., Dept. Of Electronic Eng., Korea University

### 요 약

다양한 통신 서비스에 대한 사용자의 요구로 말미암아, 이를 지원하기 위한 지능망 서비스의 도입이 활발해 지고 있으며, 이를 용이하게 하는 지능망 서비스 플랫폼의 개발 역시 중요한 이슈로 자리잡아 왔다. 이 때, 이러한 지능망 서비스 플랫폼 개발시, 그 성능을 미리 평가해 보기 위해 실제 교환기에 바로 적용하는데에는 많은 환경상의 제약이 따른다. 이에 본 논문에서는 소프트웨어적인 간이 SSP를 설계함으로써, 개발하는 지능망 플랫폼의 물리 실체를 시뮬레이션 할 수 있게 하는 방법을 제공한다.

### 1. 서론

다양한 형태의 통신에 의한 정보의 교환이 사회 생활을 영위하는 절대적인 요소로 등장함에 따라 통신 서비스에 대한 사용자 요구 사항이 증가되고 있다. 이를 지원하기 위하여 다양한 사용자 요구 사항에 부응하고 새로운 서비스를 신속히 도입할 수 있는 지능망의 구현 및 보급이 각 국가, 통신사업자 별로 추진되고 있다. 지능망은 새로운 서비스의 도입과 관리를 용이하게 할 수 있으며, 모든 통신 망에 적용할 수 있는 구조적 개념이다.

지능망의 요소를 물리적으로 구분하면, 사용자의 망 접근 및 지능망 호 연결을 위한 교환 기능을 제공하는 SSP(Service Switching Point), 특수 자원을 가지고 사용자와 지능망 사이의 유연한 상호작용을 제공하는 IP(Intelligent Peripheral), 지능망 서비스 수행에 필요한 가입자 및 망 데이터를 가지고 있는 SDP(Service Data Point), 지능망 서비스 처리에 필요한 서비스 로직과 데이터를 가지고 SSP, IP, SDP 등과 연결되는 SCP(Service Control Point), 그리고 그 밖의 요소로 나뉘어진다. 이 때, 지능망 플랫폼 개발의 중추가 되는 SCP 개발에 있어, 이의 동작 여부를 확인하기 위하여 SSP, IP, SDP 등의 요소가 반드시 필요하게 되며, 실제 전화국에 도입되기에 앞서, SCP와 연동하는 SSP, IP, SDP의 개발은 전체 지능망 플랫폼의 동작을 시뮬레이션하기 위해 필요하다. 본 논문에서는 최소한의 기능을 수행하는 간이 SSP를 ITU-T의 IN CS-2(Intelligent Network Capability Set-2)에 기반을 두어 설계하였다. 이 간이 SSP는 현재 구현하고 있는 SCP, IP, SDP와 더불어 전체적인 지능망 플랫폼의 시뮬레이션을 할 수 있는 기능을 갖는다.

본 논문의 2장에서는 표준안에 제시된 SSP의 역할 및 그 구조에 관하여 언급하고, 3장에서 본 논문에 설계된 간이 SSP의 구체적인 구조가 제시되며 4장에서 결론을 맺는다.

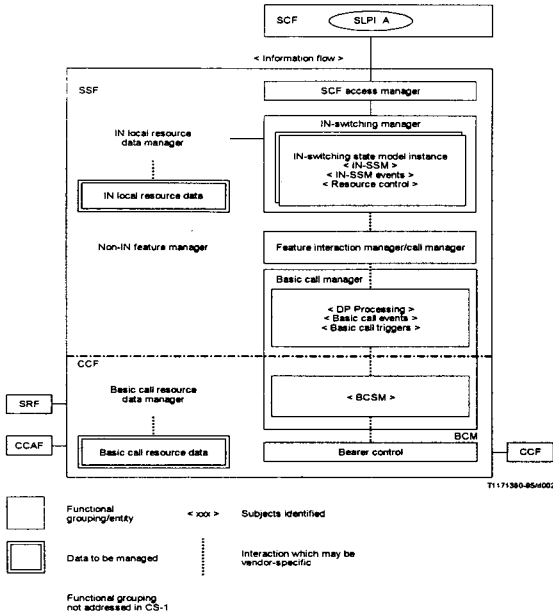
### 2. SSP의 역할 및 구조

SSP는 단국 교환기로 사용될 경우에는 사용자가 망으로 접근할 수 있게 하고, 지능망 호 연결을 위하여 필요한 교환

기능을 제공한다. 이를 위하여 SSP는 지능망 서비스를 감지하는 기능과 SCP와 같은 다른 물리 실체와의 통신 능력과 다른 물리 실체로부터 수신된 명령에 대한 응답능력을 가진다. 기능적으로 SSP는 CCF(Call Control Function)와 SSF(Service Switching Function)를 포함하며, CCAF(Call Control Agent Function), SCF(Service Control Function), SRF(Specialized Resource Function), SDF(Service Data Function)를 선택적으로 포함할 수 있다. 본 논문에서 설계하는 SSP는 CCF와 SSF만을 포함하는 것으로 한다. 그림 1에 ITU-T CS-2에서 제시한 CCF와 SSF를 포함하는 SSP의 구조가 나와 있다. 그림에서 BCM(Basic Call Manager)은 CCF와 SSF에 걸쳐 나타나며, IN 서비스 로직 인스턴스를 호출하거나 이미 활동중인 IN 서비스 로직 인스턴스에 보고되어 지는 기본호와 연결 제어 사건을 받아 들이는 역할을 수행하며, 이를 위하여, BCSM(Basic Call State Model)과 DP(Detection Point) 처리가 구현되어 있다. CCF는 BCM뿐 아니라, Bearer Control, 그리고 그밖에 기본호 자원과 관련된 블록으로 표현된다. 그 밖에, 지능망 교환과 SCP와의 통신을 위한 기능 요소들이 SSF에 나와 있다.

새로운 호 처리 요청이 들어 오게 되면, BCM에서 BCSM을 통해, 요청된 호에 관한 상태 정보를 유지하게 되며, 각각의 DP에서 주어진 사건을 포착하여, 필요한 SCP와의 통신등을 처리하는 것이 그림 1의 모델에서 제시한 궁극적인 목표이다. 여기에서 BCSM은 하나의 기정된 일반적인 상태 천이 모델이며, 각 상태는 DP(Detection Point)와 PIC(Points In Call)로 나타내어진다. 호에 관련된 각 사건(hook-off, dial, hook-on 등)에 대해, 상태 천이가 일어 나게 되며, DP로 천이된 호 상태에서는 TDP(Triggering Detection Point)와 EDP(Event Detection Point) 테이블에서 정의된 Criteria에 따라 조건을 만족할 경우, 알맞은 행동(SCP와의 통신)을 통해 지능망 호 처리를 할 수 있게 하여 주며, SCP에서 반환된 결과에 따라 알맞은 PIC로 상태 천이가 일어나게 되거나, DP 바로 아래의 PIC로의 천이가 일어나게 된다.

여기에서, TDP는 실시간적으로, SCP에서 요구한 사항을 반영하는 DP이며, EDP는 정적으로 정해진 일반적인 DP이다. 본 논문에서 설계된 TDP 테이블과 EDP 테이블은 3장에 구체적으로 제시되어 있다.



[그림 1] ITU-T CCF/SCF Model

### 3. 간이 SSP의 설계

본 논문에서 설계한 SSP는 지능망 플랫폼 개발에 있어 지능망 호 처리 시뮬레이션을 수행할 수 있는 간이 SSP로서, 실제 교환기에 적용되는 모든 기능이 포함되지는 않는다. 예를 들어 호 발생은 실제 전화기가 아닌, 스크립트 파일을 읽어 들이는 방식을 택하였으며, 호 교환 기능은 배제하였다. 다만, CS-2의 BCSM을 충실히 이행하여, SCP, IP 등과의 지능망 호처리에 필요한 상호작용을 표준안에 근거해 설계하였다.

[그림 2]는 간이 SSP의 구조를 보이고 있다. 간이 SSP의 역할은 Call Script File로부터 사용자가 지정한 호 생성 정보(User Input)를 바탕으로 호 요청을 생성하고, 이 호 요청을 처리할 수 있는 호 인스턴스를 생성시킨다. 이후, 주어진 TDP 정보 테이블, EDP 정보 테이블, BCSM State Table 등을 이용하여, SCP, IP와의 통신을 수행할 수 있게 해준다. 또한 호 처리 결과에 대한 내용은 Log File로 기록하여, 개발자가 이를 바탕으로 SCP의 동작에 대한 검증을 수행할 수 있도록 한다.

간이 SSP의 내부 구성 요소는 크게 Call Manager와 Call Generator로 나뉘어지며, 외부 구성 요소는 Log File, Call Script File, 그리고 User Input으로 나뉘어진다. 내부 구성 요소 중, Call Manager는 다시 Call Instance Manager와 Message Handler, Call Instance, 그리고 각종 Table 정보로 구성된다.

### 3.1 간이 SSP의 외부 구성 요소

#### 3.1.1 호 기술 파일(Call Script File)

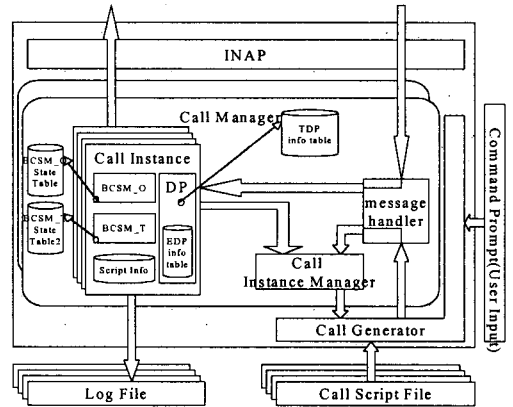
Call Script File 안에는 호에 대한 정보가 저장되어 있고, 발신측의 번호와 착신측의 번호 및 발신자가 호를 진행시키면서 수행할 수 있는 모든 호 관련 동작들이 기술된다. 하나의 Call Script File은 한 호에 대한 정보를 나타내게 된다.

#### 3.1.2 사용자 입력(User Input)

사용자 입력은 Command Prompt 상에서, SSP로 호를 요청하기 위해 Call Script File로 이미 기술된 호 정보를 이용, 새로운 호를 생성시키는 역할을 수행한다. 사용자는 Command Prompt 상에서 스크립트 파일의 파일이름들을 연속적으로 명시함으로써, SSP가 동시 다발적인 호를 발생하도록 요청할 수 있는 방법을 제공해 준다.

#### 3.1.3 로그 파일(Log File)

로그 파일은 하나의 호 인스턴스가 생성되어, Call Script File에 저장된 정보에 따라 호를 진행하고, 종료할 때까지 호의 진행 상황에 대한 정보를 기록한다.



[그림 2] 설계된 간이 SSP

### 3.2 간이 SSP 내부 구성 요소

#### 3.2.1 호 생성기(Call Generator)

사용자 입력으로부터 얻은 호 생성 정보를 이용하여, 해당 Call Script File을 읽고, 호 요청과 함께 Call Script File에 저장된 호 정보를 호 관리자(Call Manager)로 보내게 된다. 또한 호 관리자로 부터의 호 관리자내의 쓰레드 개수에 관한 정보를 관리하여 쓰레드 개수 제약등의 한계 상황 발생시, 다른 호 관리자 프로세스를 fork해주는 역할을 수행한다.

#### 3.2.2 호 관리자(Call Manager)

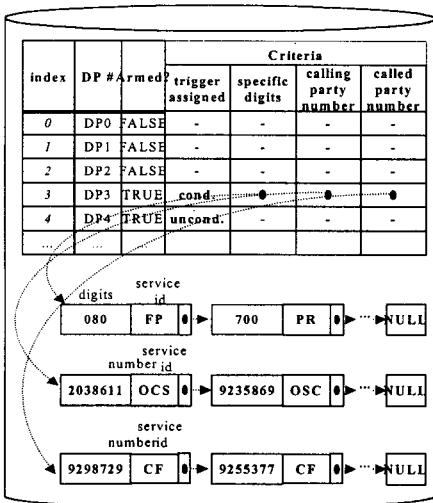
발생된 호를 처리하는 주된 역할을 수행하며, 하나의 프로세스 구성된다. 호 관리자 프로세스 내에서 동시에 여러 호의 처리를 각각 하나의 쓰레드로 처리하게 되며, 만일 하나의 프로세스에서 처리할 수 있는 최대의 쓰레드 개수를 초과하는 경우, 이를 호 생성기에 보고한다.

하나의 호 관리자의 구성요소는 다음과 같다.

- 호 인스턴스 관리자(Call Instance Manager) : 처음 호 관리자(Call Manager) 프로세스가 fork 되었을 때 생성되는 쓰레드로, 필요한 다른 쓰레드를 생성시켜주는 것을 주된 역할로 하게 된다. 또한, 부수적으로 자신이 생성한 호 인스턴스 쓰레드의 처리가 끝났을 때, 이를 해당 호 인스턴스로 부터 통보 받아, 호 생성기에 다시 이 사실을 통보해 준다. 이로써, 호 생성기는 추가의 호 관리자를 fork시켜야 할 때를 판단할 수 있게 된다.
- 메시지 핸들러(Message Handler) : 호 생성기(Call Generator)와 SCP에서 들어오는 메시지를 종합적으로 처리하는 스

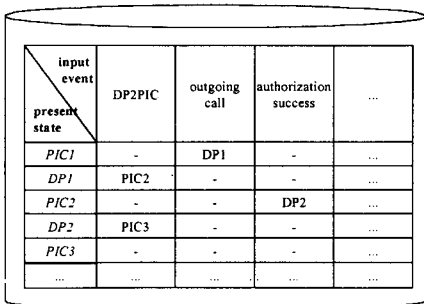
래드이다. 메시지 핸들러로 들어오는 메시지의 내용을 판단하여, 알맞은 쓰레드로 재전달 시켜주는 역할을 수행한다. 이를 위해 각 쓰레드에 대해, 쓰레드 ID 매핑 table 을 유지한다.

- TDP 정보 테이블(TDP Info Table) : Trigger Detection Point 에 관한 정적인 정보를 가지고 있으며, 모든 호에서 이 테이블을 공유하여 참조하게 된다. 이 테이블의 구체적인 구조에 대한 예는 [그림 3]과 같다. TDP Info Table 에는 각 DP 에 대한 Armed 여부와 Armed 되었을 때 SCP 와의 상호작용 여부를 판단할 Criteria 가 trigger assigned, specfic digits, calling party's number 그리고 called party's number 로 구분되어 있고, trigger assigned 항목을 제외한 나머지는 각 데이터가 linked list 로 구현된다.



[그림 3] TDP Info Table 의 구조

- BCSM 상태 테이블(BCSM State Table) : BCSM 상의 PIC(Point In Call)와 DP 의 상태 천이 정보가 들어 있으며, 하나의 호 관리자는 BCSM\_O 와 BCSM\_T 에 대해 각각 하나씩 가지고 있다. 모든 호 인스턴스는 이 두개의 테이블을 공유해 참조한다. BCSM State Table 에 저장되는 정보는 [그림 4]와 같다.

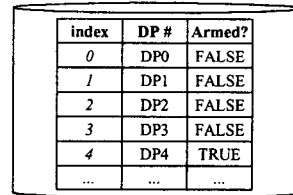


[그림 4] BCSM State Table 의 구성

- 호 인스턴스(Call Instance) : 하나의 호에 대해 하나씩 생성되며, 각 호에 대한 동작을 수행하며, 로그 파일

(Log File)을 생성시키는 역할을 수행한다. 각 Call Instance 에 포함되어 있는 세부 구성 요소는 다음과 같다.

- ✓ BCSM\_O 클래스 : 발신 BCSM 의 역할을 수행하는 클래스로써, 멤버 변수로 BCSM\_O 상태 테이블을 가리키는 포인터를 가지고 있으며, 이를 이용해 상태를 천이하는 멤버 함수를 가지고 있다. 필요시, SCP, IP, BCSM\_T 클래스와 상호동작한다.
- ✓ BCSM\_T 클래스 : 착신 BCSM 의 역할을 수행하는 클래스로써, 멤버 변수로 BCSM\_T 상태 테이블을 가리키는 포인터를 가지고 있으며, 이를 이용해 상태를 천이하는 멤버 함수를 가지고 있다. 필요한 경우, SSCP, IP, BCSM\_O 클래스와 상호동작한다.
- ✓ DP 클래스 : BCSM 의 상태가 DP 에 머물렀을 때, DP 에 대한 적절한 동작을 수행하는 클래스이다. DP 클래스는 다른 호 인스턴스와 공유하는 TDP Info Table 과, 각 DP Class 별로 가지고 있는 EDP Info Table 를 참조한다. EDP Info Table 의 예는 [그림 5]과 같다. TDP Info Table 과는 달리, Criteria 항목은 없기에 EDP Info Table 에는 포함되어 있지 않다.



[그림 5] EDP Info Table 의 구성

- ✓ Script Info 클래스 : 호 생성기(Call Generator)에서 호 인스턴스를 생성할 때 전달하는 Call Script File 에 포함되는 정보를 유지하는 클래스이다. 호 처리 도중 사용자 상호작용이 필요한 경우, 사용자의 행동을 에뮬레이션하는 역할을 수행한다.

#### 4. 결론

본 논문에서 제시된 간이 SSP 모델은 ITU-T CS-2 에 기반을 두어 설계되었다. 이 간이 SSP 모델을 이용하여, 지능망 플랫폼의 전체적인 동작을 시뮬레이션 해 볼 수 있다. 이는 실제 지능망 서비스가 도입되기 이전에 플랫폼의 테스트 및 성능 평가를 위하여 요구되는 사항으로써, 하드웨어적인 전화기의 접속 및 연결, 교환 없이 소프트웨어적인 에뮬레이션으로 지능망 플랫폼의 동작을 가능하게 한다는 점에서 그 의미가 있다. 즉, 지능망 플랫폼의 기타 물리 실체 구현시 이를 실제 교환기에 적용하기 이전에, 본 간이 SSP 를 이용하여 손쉽게 그 성능을 테스트 해 볼 수 있기에, 지능망 플랫폼의 개발시 유용하게 사용될 수 있으리라 기대된다.

#### 참고 문헌

- [1] ITU-T Rec. Q.1224, "Distributed Functional Plane for Intelligent Network CS-2", 1997.
- [2] ITU-T Rec. Q.1220 ~ Q.1223, Q.1225, "Intelligent Network capability Set 2", 1997.
- [3] ITU-T Rec. Q.1228, "Interface Recommendation for Intelligent Network Capability Set", 1997.
- [4] 최고봉, 김기령, 김태일, 윤병남, "지능망 기술", 홍릉 과학출판사, 1996.