

개방형 교환시스템에서 SDL을 이용한 MEGACO 프로토콜의 설계

엄남경^U 우성희* 이병선** 이상호
충북대학교 컴퓨터과학과, 청주과학대학 컴퓨터과학과*
한국전자통신연구원**
family8@cnlab.chungbuk.ac.kr

Design of MEGACO using SDL in Open Switching System

Nam-Kyoung Um^U Sung-Hee Woo* Byoung-Sun Lee** Sang-Ho Lee
Dept. of Computer Science, Chungbuk National Univ.

요 약

인터넷 서비스의 이용자가 증가함에 따라 데이터 망의 구축을 위한 투자의 필요성이 제기되고 있으며, 여러 벤더들로부터 최적의 장비를 구입하여 망을 구축할 수 있도록, 통신 장비들의 상호 운용성을 지원하는 개방형 구조의 망 및 교환 시스템을 표준화하는 작업이 요구되고 있다. 본 논문에서는 개방형 교환 시스템에서 사용되는 MEGACO 프로토콜을 SDL 형식으로 설계한다. 개방형 교환 시스템은 시스템의 각 구성 요소마다 시장 경쟁 개념이 도입될 수 있도록 수평적 계층구조를 갖추고 있다. 각 계층마다 표준 인터페이스를 정의하고, 이 규격을 만족하는 다수의 벤더들로부터 성능이 우수하고 기능이 뛰어난 모듈을 제공받을 수 있으므로, 망 사업자들은 특정 벤더에 종속되지 않고 개선된 양질의 서비스를 경제적으로 제공할 수 있는 개방형 교환 시스템을 쉽게 구축할 수 있게 된다.

1. 서론

통신 기술의 비약적인 발전과 고품질 서비스에 대한 수요가 증대함에 따라 무한 경쟁의 통신 시장 환경이 도래되고 있다. 이는 기존 서비스의 경제적 제공 및 신규 서비스의 유연한 수용의 필요에 따라 종래와는 다른 모습의 통신망 인프라 구축의 필요성이 요구되고 있으며, 그것을 실현하기 위한 기술로써 등장하고 있는 것이 개방형 멀티서비스 통합 교환 기술이다. 기존의 통신망 인프라에서는 하나의 서비스 제어 및 운용 방식으로 멀티서비스를 수용하는 방식을 이용하여 서비스별로 독립적인 자원 관리 및 망 운용 관리가 이루어지지 않는데 반하여, 개방형 멀티서비스 통합 교환 기술에서는 계층간 개방형 인터페이스가 적용되는 수평적 계층 구조를 가지며 각 계층마다 여러 통신 장비업체가 경쟁적으로 장비나 서비스 개발이 가능하다는 장점을 가진다. 이를 통해 멀티서비스 통합 관리를 통한 망 운용 유지비를 최소화시킬 수 있다[1].

MEGACO(Media Gateway Control)는 서로 다른 망을 연동하기 위해 물리적으로 분산되어 있는 MG(Media Gateway)와 MG를 제어하기 위한 MGC(Media Gateway Controller)사이에서 쓰이는 프로토콜이며, 특히 회선망과 패킷망을 연결시키기 위해 쓰인다[2][3]. 예를 들면, PSTN(Public Switched Telephone Network: 전화통신망)과 IP(Internet Protocol)망, 혹은 ATM(Asynchronous Transfer Mode: 비동기 전송 모드)등을 연결하기 위해 사용한다. 이를 위한 인터페이스로 Tone Signaling, ISDN, ISUP, QSIG, GSM과 같은 다양한 SCN(Switched Circuit Network) Signaling을 지원할 것이며, 이는 스위치와 제어를 분리해내려는 개방형의 움직임에 맞춰 MG와 MGC로 분리해서 처리해보려는 개방형 프로토콜 기술 중의 하나라고 할 수 있다[4][5].

본 논문에서는 개방형 교환 시스템에서 사용되는 프로토콜인 MEGACO를 SDL/GR(Specification and Description/Graphical Representation)을 이용한 설

계 방법을 제어흐름을 중심으로 제시한다. MEGACO에서는 MG와 MGC간에 Command를 이용하여 연결관리를 한다. 따라서 특히 중점을 두어 설계할 부분은 MG와 MGC간 연결되는 Command의 흐름이다[6].

본 논문의 구성은 다음과 같다. 2장에서는 MEGACO와 SDL에 대해 기술하고, 3절에서는 MEGACO 프로토콜의 Command에 따른 흐름을 SDL로 설계하는 방법을 제시한다. 결론 및 향후 연구 방향은 4절에서 제시된다.

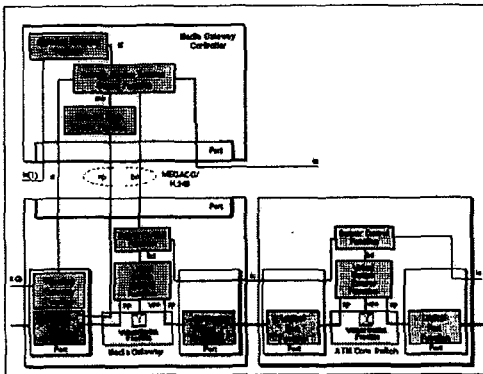
2. MEGACO 프로토콜과 SDL

2절에서는 MEGACO 프로토콜의 개념과 MG와 MGC를 구성하기 위해서 MSF(Multiservice switching forum)에서 제시하는 MEGACO 프로토콜의 위치에 대해서 살펴본다. 또한 Command와 Descriptor에 대해서 알아본 후, SDL의 개념과 적용 규칙에 대해서 살펴보자.

2.1 MEGACO 프로토콜

MEGACO는 IETF(Internet Engineering Task Force)에서 나온 규격으로 MGC와 MG사이에 자료 흐름을 위한 프로토콜이다. 현재 많이 쓰이는 MGCP(Media Gateway Control Protocol)가 주로 음성처리를 위한 프로토콜이라면, MEGACO는 음성뿐 아니라 영상회의, 동영상과 같은 멀티미디어 정보처리를 위한 프로토콜이다.

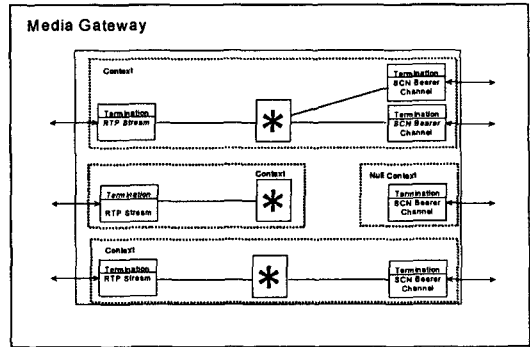
망에서 MEGACO가 구성되는 위치는 [그림2]와 같다.



[그림2] MSF에서 제시하는 MEGACO의 위치

MGC는 하나의 MG내의 미디어 채널을 위해 연

결제어에 합당한 call state의 부분을 제어한다. MG는 미디어를 변환시키고, 관련된 기능들을 지원하는 역할을 한다. 특히 MEGACO는 MG내에 연결모델(Connection Model)을 두며, 이것은 하나 이상의 스트림을 고정시키거나 통과시키는 논리적인 지점을 나타내는 Termination과 그 집합으로 표현할 수 있는 Context로 구성된다. 또한 End point의 Termination을 연결시켜주는 Topology로 나타낸다.



[그림3] MG의 연결모델

2.2 SDL

SDL은 통신 시스템의 동작을 명세하고 서술하기 위한 언어로서, SDL에서 시스템은 상호 연결된 수많은 모듈 즉 블록으로 구성되고, 각 블록의 동작은 하나 혹은 여러 개의 프로세스에 의해 모델링 되어지며 이들 사이의 통신은 채널과 무한 용량의 큐를 통하여 비동기적으로 이루어진다[8][9]. 특히 다양한 통신 시스템이 SPC(Stored Program Control) 기법을 사용하게 됨에 따라 소프트웨어 구성 및 관리의 문제가 발생하여, 통신 시스템의 동작을 명확히 명세하고 서술하기 위한 형식언어로서 사용되고 있다. 또한 데이터는 추상적인 데이터타입으로 정의된다. SDL은 SDL/GR(Specification and Description Language/Graphical Representation), SDL/PR(Specification and Description Language/textual Phrase Representation), SDL/PE(Specification and Description Language/Pictorial Element)의 3가지 방법으로 표현될 수 있다. 그중 SDL/GR은 여러 개의 정해진 기호들을 사용하여 순서도와 유사하게 작성된다.

본 논문에서는 SDL/GR을 이용하여 MEGACO의 MGC와 MG 측면에서 Command의 제어 흐름에 따라 개략화된 구조의 설계방법을 제시한다.

3. MEGACO 프로토콜의 SDL 설계

3.1 제약 사항

실세계의 프로토콜은 규모가 방대하고 복잡하여 실제로 모델링하기가 어렵다. 따라서 본 논문에서 사용하는 모델은 다음의 사항을 가정한다[8][9].

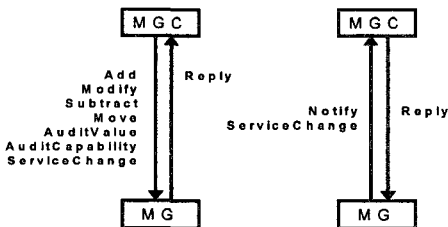
- 1) 완전히 명세되었다(Completely specified).
- 2) 강하게 연결되었다(Strongly connected).
- 3) 최소 밀리 머신이다(Minimal mealy machine).
- 4) 단일모듈명세이다(Single module specification).
- 5) 리셋 기능이 있다(Reset feature).

3.2 MEGACO Command의 표현

MEGACO는 MGC와 MG사이의 프로토콜로써 각각의 장치들이 Command로 제어된다. 즉, Command는 프로토콜 연결 모델의 논리적 요소의 조작 또는 Context에 Termination을 추가, 변경, 삭제하거나 Termination에서 다루어질 Event들의 내용 및 행동, 사항을 표현한다. [표1]은 MEGACO의 Command의 종류를 나타내고, [그림4]는 MGC와 MG간에 Command의 방향성을 나타낸다.

Command	Comment
Add	context에 termination 추가, null context라면 하나의 새로운 context 생성
Modify	termination의 속성, 이벤트, 신호 수정
Subtract	termination의 연결 해제
Move	다른 context로의 이동
AuditValue	현상태의 termination의 속성, 이벤트, 신호를 리턴
AuditCapability	MG에 의한 termination 속성, 이벤트를 위한 모든 가능한 값을 리턴
Notify	MG에서 현재의 이벤트를 MGC에게 알리기 위한
ServiceChange	restart등의 명령을 알리게 위해 사용

[표1] Command



[그림4] MGC와 MG간 Command의 방향성

Add, Modify, Subtract, Move, AuditValue, AuditCapability와 같은 명령어는 MGC에서 MG로 내리는 명령어이다. 이 명령어들에 대해 MG는 받았다

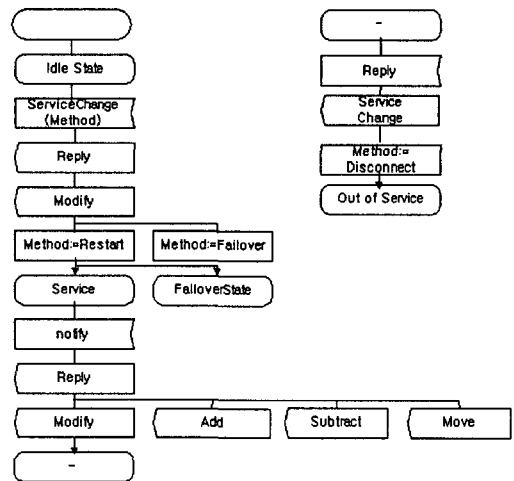
는 Reply를 보내주어야 한다. Notify는 MG에서 MGC로 보내는 명령어이다. 이는 Termination에 일어난 이벤트를 MG가 제어측인 MGC에게 알리기 위한 명령어이며, ServiceChange는 MG와 MGC 모두 먼저 시동할 수 있는 명령어이며, 이때 함께 보내는 ServiceChange 기술자의 Method에 일어날 서비스의 종류를 명시한다[7].

3.2 MEGACO와 SDL의 대응관계

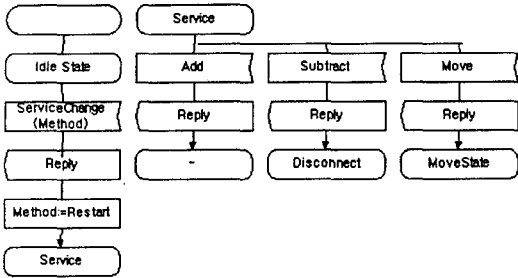
[표2]와 같이 MEGACO 항목에 따라 SDL/GR 항목을 일대일로 대응시켜볼 수 있다. Command와 Reply는 경우에 따라 입력신호나 출력신호가 될 수 있다. Action은 Task로 표시하며, State는 상태가 될 수 있다.

MEGACO 항목	SDL 항목	SDL 기호
Command Reply	입력 신호(Input signal)	
Action	태스크(Task)	
Command Reply	출력 신호(Output signal)	
State	상태(State)	

[표2] MEGACO와 SDL의 대응관계



[그림5] MGC 측의 SDL 표현



[그림6] MG 측의 SDL 표현

[그림5][그림6]과 같이 ServiceChange Command의 파라메터 Method를 통해 원하는 서비스의 종류를 시동하면 MGC나 MG측에서는 Reply를 리턴하며, 서비스상태에 들어가 원하는 MGC측은 제어를 맡고, MG측은 MGC에서 기술하는 Command에 따라 스위칭기능을 할 수 있다. 따라서 제어정보와 실제 운용되어지는 데이터정보를 따로 관리할 수 있게 된다.

4. 결론 및 향후 연구 방향

기존의 통신망 인프라에서는 하나의 서비스 제어 및 운용 방식으로 멀티서비스를 수용하는 방식을 이용하여 서비스별로 독립적인 자원 관리 및 망 운용 관리가 이루어지지 않는데 반하여, 개방형 멀티서비스 통합 교환 기술에서는 계층간 개방형 인터페이스가 적용되는 수평적 계층 구조를 가지며 각 계층마다 여러 통신 장비업체가 경쟁적으로 장비나 서비스 개발이 가능하다는 장점을 가진다. 이를 통해 멀티서비스 통합 관리를 통한 망 운용 유지비를 최소화시킬 수 있다. 또한 MEGACO는 개방형 교환 시스템에서 MG와 MGC와 같은 컴포넌트간의 프로토콜을 정의하고 있기 때문에, MEGACO를 이용하여 제어하는 쪽과 교환서비스를 하는 쪽의 신축성있는 서비스를 기대해 볼 수 있다. 또한 아직 MEGACO에 관한 연구가 미흡하여 IETF나 ITU-T의 규격 문서외의 자료들이 부족한 실정이다. 이 논문을 통해 MGC와 MG간의 MEGACO 프로토콜에 대한 관련 솔루션의 개발환경에 있어 기반을 제공하고 있다.

향후에는 MEGACO 프로토콜에 대해서 개방형 교환 시스템 환경에서의 시험열 생성방법에 대해서 제안하고, SDL을 이용하여 데이터흐름과 제어흐름을 모두 고려한 시스템 설계를 통해 통합된 환경에서 구현하는 것이 요구된다.

참고 문헌

- [1] 이정규, 이순석, 김영부, 전경표, "개방형 멀티서비스 통합 교환 기술", 한국통신학회지 17권 2호, pp.153~166, 2000. 2.
- [2] M.Arango, A.Duran, I.Elliot, C.Huitema, S.Pickett, "Media Gateway Control Protocol(MGCP)", RFC 2705.
- [3] B.Hill, N.Greene, C.Huitema, A.Rayhan, B.Rosen, J.Segers, "draft-ietf-megaco-protocol-05.txt", "Work-in-progress"
- [4] N.Greene, M.A.Ramallo, B.Rosen, "draft-ietf-megaco-reqs-08.txt"
- [5] P.Blatherwick, P.Holland, "draft-ietf-megaco-ipphone-02.txt"
- [6] N.Greene, C.Huitema, A.Rayhan, B.Rosen, J.Segers, "draft-ietf-megaco-protocol-08.txt"
- [7] 김화숙, "MEGACO/H.248 프로토콜 기술", 『개방형 네트워크 기술 워크샵』 205-228p.
- [8] 박병욱, "통합 흐름을 고려한 테스트 스위트 생성 기법", 충북대학교 석사학위 청구논문. 1996.
- [9] 오병호, "자료 흐름의 변칙성을 고려한 프로토콜의 테스트 스위트 생성", 충북대학교 박사학위 청구논문, 1996.