

웹서비스 기반 Open API를 이용한 IMS 망의 SIP 세션 제어

정희원 박유미*, 임종철*, 최영일*, 종신회원 김상하**°

SIP Session Control using Web service-based Open API in IMS

Yoo-mi Park*, Jong-chul Yim*, Young-il Choi* *Regular Members*,
Sangha Kim**° *Lifelong Member*

요약

본 논문에서는 통신망의 기능들을 표준화된 인터페이스로 개방하고자 하는 Open API 기술인 Parlay X API를 분석하고, 이에 기반하여 멀티미디어 호를 제어할 수 있는 확장 호 제어 (Extended Call Control) API를 IMS 망에 적용하기 위한 SIP 세션 제어 방법을 제안한다. 제안 방법은 확장 호 제어 Open API들의 조합으로 발생할 수 있는 다자간 멀티미디어 호 제어 시 기본적으로 필수적인 세션의 생성, 종료, 복원을 수행하는 세션 전이 방법을 포함한다. 또한 Open API를 이용한 IT 응용 사례를 통해 SIP 세션 방법에 대해 기술하고, 제 3자 호제어, 네트워크 발생 호제어, 멀티미디어 인터랙션 API를 이용하여 복잡한 멀티미디어 호를 용이하게 제어할 수 있음을 보인다. 마지막으로, 주요 호제어 API를 대상으로 개방형 서비스 게이트웨이에서의 처리 성능을 측정하여 교환기의 호처리 성능과 비교한 결과, Open API의 호처리 성능이 교환기 호처리 성능에 크게 뒤지지 않음을 보였다.

Key Words : Open API, Extended Call Control, Session Management, SIP, IMS

ABSTRACT

This paper reviews the Parlay X and proposes SIP multimedia session control mechanism which is able to be implemented over IMS using Extended Call Control APIs of Parlay X. Parlay X is one of the industrial standards of Open API in the telecommunication to open network resources and capabilities to third party service providers. In this paper, SIP session control mechanism is described by depicting call flows of basic and essential session handling methods including session initiation, transfer, restoration, and termination. We also show how easy Parlay X Extended Call Control APIs can be used for the purpose of complicated calls handling in the IT applications. To verify feasibility of the blended services based on convergence of telecommunication and internet with regard to performance, we accomplish experimental performance of Extended Call Control APIs from IT application through open service gateway. We show Open API could be applied to next generation network based on IMS without serious degradation of the network performance.

※ 본 연구는 정보통신부 및 정보통신연구진흥원의 IT성장동력기술개발사업의 과제 (2005-S-056-03)로 수행되었습니다.

* 한국전자통신연구원 융합기술연구부문 개인화서비스플랫폼연구팀 (parkym@etri.re.kr, hektor@etri.re.kr, yichoi@etri.re.kr)

** 충남대학교 정보통신공학과 교수(shkim@cnu.ac.kr) (° : 교신저자)

논문번호 : KICS2008-04-164, 접수일자 : 2008년 4월 11일, 최종논문접수일자 : 2008년 11월 11일

I. 서론

다양한 분야 간의 컨버전스를 요구하는 시대적 흐름에 따라 통신망의 서비스 제공 방법에도 새로운 패러다임이 요구되고 있다. 이는 지금까지 통신 사업자의 고유 영역이었던 통신망의 기능들이 타 분야의 응용에서 쉽게 이용할 수 있도록 개방되어야 함을 의미한다.

그러나 최근까지 통신 사업자들은 통신망 자원의 보호와 안전한 서비스 제공을 위해 중앙 제어식 통신 서비스 제공 방법을 고수해 왔으며, 이는 통신 분야가 다른 분야와 융합하기 어려운 다소 폐쇄적인 분야로 보이는 원인으로 작용해 왔다. 대표적인 통신 서비스 제공 방법의 하나로 1990년대부터 부가 서비스 개발에 이용되어 온 지능망 프로토콜은 프로토콜의 내용과 사용이 복잡하고 통신 사업자의 망에 의존적이어서 사업자 별로 중복하여 서비스가 개발되고 관리되는 문제점을 가지고 있었다. 2000년대 들어서는 IP망에서 통화 서비스를 구현하기 위해 등장한 SIP (Session Initiation Protocol)가 현재 IMS 기반의 차세대 통신망에서 세션 제어를 위한 표준 프로토콜로 자리잡아 가고 있다. 그러나 지능망 프로토콜이나 SIP가 통신 전용 서비스 개발에는 유용한 방법이나, 일부 통신 기능을 이용하고자 하는 일반 IT 개발자에게는 프로토콜의 복잡성과 개발 환경의 부족 등으로 적합하지 않은 문제가 있다.

따라서 일반 IT 응용에서 통신 기능을 손쉽게 이용하도록 망의 기능을 개방하고자 2000년대 초반부터 3GPP, ETSI, Parlay 그룹을 중심으로 통신 분야의 개방형 서비스 인터페이스^[1-3] 표준화 작업을 활발히 추진해 왔다 (그림 1). 그러나 개방형 서비스 인터페이스 표준화 기술의 발전에 비해 인터페이스의 하부망 연동에 관한 연구 사례는 드물어 망 연동

시의 고려사항과 문제점을 공유하지 못하는 문제가 있었다. 현재까지 개방형 서비스 인터페이스 표준화 기술 중 CORBA 기반의 Open API 기술에 대해서는 SIP 네트워크 적용 사례가 발표되고 있으나^[4-5], 웹서비스 기반 Open API 기술의 망 적용 연구 사례는 발표되지 않고 있다. 웹서비스 기반의 Open API는 CORBA 기반의 Open API에 비해 망의 기능을 더욱 추상화시켰기 때문에 응용에서 사용하기에 편리한 반면, API마다 내포된 기능의 폭이 넓어 구현 시 하부망 연동을 위한 API의 기술적 분석이 필수이다.

본 논문에서는 웹서비스 기반 Open API 중 확장 호 제어 API에 대한 하부망 연동 기술 중 세션 제어 방법을 제안한다. 차세대 통신 서비스 네트워크로 표준화된 IMS에 적용 가능한 세션 생성, 종료, 복원, 전환 과정을 통해 기본적으로도 필수적인 SIP 세션 제어 방법을 제안함으로써 추후 IT 융합 응용에 이용될 다양한 호제어 API의 호출에 따른 세션 제어의 가이드라인을 제시하고자 한다. 이를 위하여 확장 호 제어 API를 제공하는 기능들 간의 상관 관계를 분석하여 데이터 모델을 정립하고, API 호출 시 발생하는 세션 변환에 따른 SIP 호 흐름도를 제시한다. 또한 주요 호제어 API를 대상으로 처리 성능을 측정하고, 결과를 통해 통신과 IT가 결합된 서비스의 상용화 가능성을 확인한다.

본 논문의 구성은 다음과 같다. II장에서는 Parlay X ECC 웹서비스를 설명하고 세션 제어를 위한 공유 정보와의 관계를 분석하며, III장에서는 IMS 구조 상에서의 ECC 제어 구조를 기술한다. IV장에서는 IMS에서 ECC 웹서비스를 적용한 SIP 세션 관리 방법을 상세히 기술하고, V장에서는 구현 성능을 보이고 마지막으로 VI장에서 결론을 맺는다.

II. 확장 호 제어 웹서비스

웹서비스 기반 Open API 기술 규격인 Parlay X에는 지능망 프로토콜 개념에 따라 call-leg 단위로 호를 제어하고, SIP 수준으로 멀티미디어 세션을 다룰 수 있으며, 호 진행 중 사용자와 인터랙션 하는 기능을 제공할 수 있는 확장 호 제어 (Extended Call Control, 이하 ECC) 규격이 제정되었다. ECC 규격은 제 3자 호 제어 (Third Party Call) 웹서비스^[6], 호 알림 (Call Notification) 웹서비스^[7], 오디오 콜 (Audio Call) 웹서비스^[8], 멀티미디어 컨퍼런스 (MultiMedia Conference) 웹서비스^[9]로 구성된다. 본

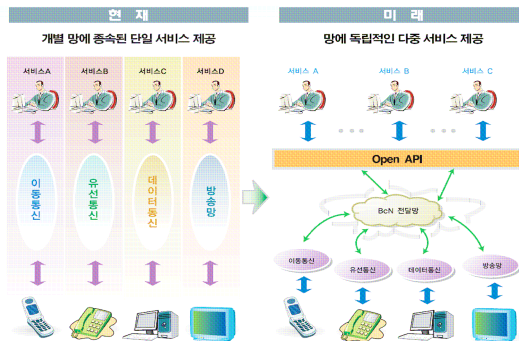


그림 1. 통신 서비스의 발전 방향

장에서는 ECC 웹서비스들의 간단한 소개와 공유 정보를 중심으로 그들의 관계를 분석한다. ECC 웹서비스들에서 제공하는 API의 상세한 내용은 Parlay X 3.0 규격^[6]을 참조한다.

2.1 ECC 웹서비스

- 제 3자 호 제어 (Third Party Call, 이하 TPC) 웹서비스^[6] : 제 3자에 의해 호 세션을 생성하고, 단말을 세션에 초대하고, 세션 간 이동을 제어하는 (단말 간의 전화연결/종료를 가능하게 하는) API 집합을 제공하는 웹서비스
- 호 알림 (Call Notification, 이하 CN) 웹서비스^[7] : 통신망의 단말에서부터 발생한 호를 응용 측으로 알려주고 응용이 이 호에 대한 처리 (호 중단, 호 경로 변경, 호 진행)를 하도록 위임하는 API 집합을 제공하는 웹서비스
- 오디오 콜 (Audio Call, 이하 AC) 웹서비스^[8] : 음성 안내나 화상 등을 통화에 실어 보내고, 단말로부터 음성을 녹음하거나 digit을 수집하는 API 집합을 제공하는 웹서비스 이 API를 사용하기 위해서는 제 3자 호제어 웹서비스나 호 알림 웹서비스의 API로 세션을 먼저 생성해야 함
- 멀티미디어 컨퍼런스 (MultiMedia Conference, 이하 MMC) 웹서비스^[9] : 제 3자에 의해 단말 간의 멀티미디어 컨퍼런스를 생성/해제하고 동적으로 회의 참가자들을 제어할 수 있는 API 집합을 제공하는 웹서비스

2.2 데이터 모델

ECC 웹서비스들 간 API 호출에 의해 세션 (Session), 참가자 (Participant), 미디어 정보 (MediaInfo)가 생성되거나 변경 또는 삭제되는데, 이들은 ECC 웹서비스들에서 공유되므로 세션 진행 중 일관성 있게 관리되어야 한다. 그림 2는 공유 정보 간의 상호 관계를 나타낸 E-R 다이어그램으로서 정보들과 웹서비스 간의 관계도 함께 나타낸다. 즉, Read/Write 권한이 있는 웹서비스와 공유 정보 간에는 긴 점선으로, Read 권한만 있는 웹서비스와 공유 정보 간에는 짧은 점선으로 표시하였다. TPC 웹서비스, MMC 웹서비스, 또는 CN 웹서비스에서 만들어진 세션은 자신들의 웹서비스 뿐 아니라 다른 웹서비스에서도 공유되며, 세션을 생성하는 과정 또는 기존의 세션을 다루는 과정에서 참가자들을 초대하거나 삭제할 수 있고 참가자 초대 시 미디어를 설정

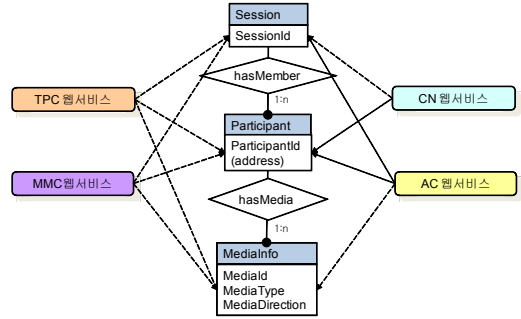


그림 2. ECC 웹서비스들 간의 정보 공유 관계도

할 수 있다. 또한 AC 웹서비스에서는 기존 세션의 참가자들에게 새로운 미디어를 추가하거나 삭제할 수 있는 API를 제공한다.

이들이 공유하는 세션, 참가자, 미디어 정보의 세부 내용은 다음과 같다.

- Session 정보 : 세션이 생성되고 소멸될 때까지 유지되는 정보로서 TPC::MakeCallSession() 또는 MMC::CreateConference() 호출에 의해 생성되고, TPC::EndCallSession() 또는 MMC::EndConference() 호출로 삭제된다. 세션정보는 ECC 웹서비스들에서 참가자를 추가하거나 삭제하는 과정을 통하여 공유되며, 참가자의 삭제 또는 이동 등으로 세션에 참가자가 존재하지 않게 되면 자동 종료되어 세션 정보가 삭제된다.
- Participant 정보 : 세션이 유지되는 동안 세션에 참가하고 있는 참가자들에 대한 정보로서 TPC::MakeCallSession() 호출로 세션이 생성될 때 파라미터로 지정된 참가자 (1명 또는 2명)가 최초의 참가자가 되며, TPC:: AddCallParticipant() 또는 MMC:: InviteParticipant() 에 의해 1명씩 세션에 추가되고, TPC::DeleteCallParticipant() 또는 MMC:: DisconnectParticipant()에 의해 1명씩 세션에서 삭제된다. 참가자는 TPC:: TransferCallParticipant()에 의해 다른 세션으로 이동될 수 있고, call transfer 등의 응용에서 이용 가능하며 세션에 참가하고 있는 동안 참가자의 상태 (Invited, Connected, Disconnected)가 유지되어야 한다.
- Media 정보 : 세션이 유지되는 동안 세션에 참가하고 있는 참가자들의 미디어 정보로서 참가자는 하나 이상의 미디어 타입으로 연결된다. 참가자의 미디어 타입은 Audio, Video, Chat, Data 중 하나이고, 방향은 In (수신만

가능), Out (송신만 가능), InOut (송수신 가능) 중 하나로 설정이 가능하다. 예를 들어 참가자 A는 Audio와 Data 타입으로 연결하고, Audio 타입은 InOut으로, Data 타입은 Out으로 지정하면 Audio는 양방향으로 Data는 송신만 가능함을 의미한다. 미디어 정보가 API 호출 시 명시적으로 지정되지 않을 경우 네트워크에서 지정하는 미디어 타입으로 자동 설정된다.

III. IMS 상에서의 확장 호 제어 구조

서론에서 기술한 개방형 서비스 구조를 통신망에 적용하기 위해서는 그림 3에서와 같이 개방형 서비스 게이트웨이와 서비스를 수행하는 응용 서버가 필요하다. 개방형 서비스 게이트웨이는 통신망의 네트워크 장치로서 서비스 개발자들이 통신망 자원에 접근할 수 있는 표준화된 API를 제공하며, API를 통신망의 프로토콜로 변환하고 제어하는 기능을 수행한다. ECC 웹서비스를 비롯해 표준화된 API들은 개방형 서비스 게이트웨이를 통해 제3의 서비스 제공자에게 제공된다. 본 장에서는 3GPP 표준 규격을 참조하여 IMS 기반의 유무선 통합망에서 개방형 서비스 게이트웨이의 위치와 ECC 제어 구조에 관하여 기술한다.



그림 3. 개방형 서비스 게이트웨이와 응용 서버

3.1 참조 네트워크 모델

그림 4는 IMS 에서 서비스가 제공되는 기능적인 구조^[10-11]를 도시한다. 기존 CAMEL 기반의 서비스는 IM-SSF를 통해 지원되며, IMS 서비스는 다양한 SIP Application Server를 통해 제공된다. 또한 3rd

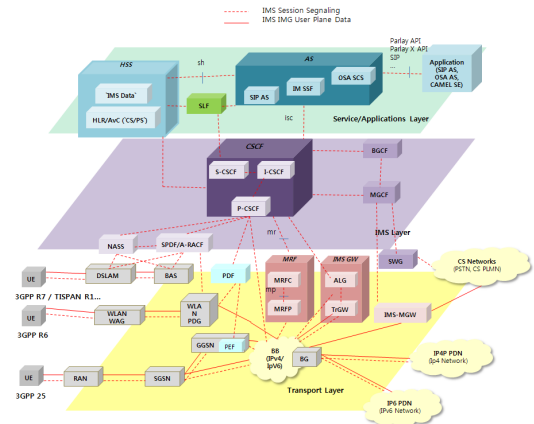


그림 4. IMS 네트워크 참조 모델

Party를 통한 다양한 서비스 개발 및 운용이 가능하도록 SCS (Service Capability Server)를 통해 OSA (Open Service Access) API를 제공하고 있다. Open API를 처리하는 개방형 서비스 게이트웨이는 OSA SCS의 한 사례로 볼 수 있으며, 세션 제어를 위해 isc 인터페이스로 S-CSCF와 연동하며, 가입자 정보를 얻기 위해 sh 인터페이스를 사용한다. ECC 웹서비스는 IMS 기반 망에서 세션 제어를 하기 위해 SIP 기반의 isc 인터페이스를 기준으로 삼아야 한다.

3.2 세션 제어 모델

ECC 웹서비스에서 처리하여야 할 세션은 크게 응용에서 세션 설정을 요구하는 Application Initiated Session과 네트워크의 단말로부터 세션 설정을 시작하는 Network Initiated Session으로 나뉜다. 일반적으로 세션 제어는 단말 간의 세션을 대상으로 하지만 다자간 호 (컨퍼런스)나 안내 방송 송출 등의 부가 서비스를 위해서는 미디어 서버와의 연동이 필요하다.

- Application Initiated Session : ECC 웹서비스의 전형적인 세션 제어 방법으로서, 호의 시작이 응용에서부터 시작되며 TPC, AC, MMC 웹서비스가 이에 해당한다. ECC 웹서비스는 서로 다른 SIP 다이얼로그를 가진 두 개의 세션 시작 요청을 전송하고, 2개의 다이얼로그를 연동함으로써 물리적인 2개의 다이얼로그가 마치 하나의 다이얼로그처럼 동작하도록 한다. 이런 형태로 동작하는 ECC 웹서비스는 RFC 3261^[13]에 명세되어 있는 B2BUA로 동작한다고 할 수 있다.
- Network Initiated Session : 종래의 호 발생의

경우와 동일하게 단말에서 시작하여 생성된 세션을 Network Initiated Session이라 하며, ECC 웹서비스 중 CN 웹서비스를 이용하여 Network Initiated Session에 다양한 부가 서비스를 제공할 수 있다. IMS에서 Network Initiated Session에 대해 두 가지 모델¹²⁾이 참조되는데, 응용 서버 (Application Server)가 SIP 프락시로 동작하는 경우와 응용 서버가 라우팅 B2BUA로 동작하는 경우이다. IMS의 응용 서버로서 동작하는 개방형 서비스 게이트웨이의 CN 웹서비스에는 두 가지 모델이 모두 적용 가능하다. CN 웹서비스 API 호출의 결과로 응용에서 'End' (호 마침) 또는 'Continue' (호 진행) 응답을 한 경우에는 SIP 프락시로 동작하며, 'Route' (다른 번호로 우회)로 응답했을 때는 라우팅 B2BUA로 동작한다.

- 미디어 서버와의 연동 : 세션이 생성되는 도중이나 생성된 후에 필요에 따라 미디어 서버와 연동될 수 있다. 예를 들어, MMC 웹서비스에서 생성된 멀티미디어 컨퍼런스의 경우 미디어 믹싱과 분배를 수행하는 미디어 서버와 연동이 요구되고, AC 웹서비스에서 안내 방송 출력이나 사용자와의 상호작용 (전화번호 수집 등) 등을 위하여 미디어 서버와 연동이 필요하다. IMS 네트워크 모델 (그림 4)에서 미디어 서버는 미디어 자원을 제어하는 MRFC (Multimedia Resource Function Controller)와 실제 미디어 자원을 처리하는 MRFP (Multimedia Resource Function Processor)라는 논리적 기능으로 구성되는데, ECC 세션의 미디어 제어는 IMS의 응용 서버로서 S-CSCF를 통하여 MRFC와 연동하여 수행할 수 있다.

IV. ECC를 이용한 IMS SIP 세션 제어

본 장에서는 응용에서 호출되는 ECC API를 IMS에 적용할 경우, 세션 생성, 전환, 복원, 삭제에 관한 세션 제어¹⁵⁾ 방법을 SIP 기반의 호 흐름도를 이용하여 제안한다. 단, SIP 시그널링 과정을 단순화시키기 위해 개방형 서비스 게이트웨이 (Parlay X GW)와 단말 간에 시그널이 전달되는 IMS 망 내의 내부 요소들 중 S-CSCF 외에는 나타내지 않는다.

SIP 세션 제어 과정을 설명하기 위하여 다음과 같이 IT 응용 시나리오를 가정한다.

- (가) 펀드 매니저 User A는 동료 직원 User B에게 오늘의 주식 현황을 토의하기 위해 데스크탑 컴퓨터에 설치된 응용 프로그램을 이용하여 통화를 시작한다 (세션 생성).
- (나) 통화 도중 상사에게 확인할 내용이 있어 그의 상사인 User C를 초대하여 세 사람이 함께 통화한다 (컨퍼런스 세션 생성).
- (다) 삼자 통화 도중 고객 User D와의 중요한 약속이 기억난 User A는, User D와의 상담을 위해 음성 통화를 새롭게 시작한다 (세션 생성 및 세션 전환). 이때 원래 User A와 통화 중이던 User C와 User B는 계속 통화 가능하다.
- (라) User A와 User D가 음성 통화로 상담하던 중, User A는 User D로부터 비밀번호를 입력 받기 위해 ARS에 연결하여 음성안내 메시지를 들려주고 비밀번호를 수집하게 한 후, 메시지를 종료하고 다시 음성 통화를 지속한다 (세션 복원). User A와 User D가 상담을 마치고 음성 통화를 종료한다.

그림 5에서 위의 시나리오를 수행하기 위한 Open API의 호출과정을 도시하며 다음과 같은 요소로 구성된다.

- User A' s Application : 전화 걸기, 통화에 초대하기 등을 인터넷에서 수행할 수 있는 인터넷 응용 프로그램으로서 본 시나리오에서는 User A가 이용한다.
- AS (Application Server) : User A가 사용하는 인터넷 응용 프로그램의 요청을 Open API로 변경하는 서비스 로직을 수행하며, 이를 개방형 서비스 게이트웨이로 전달하는 응용 서버이다.
- Parlay X GW : 개방형 서비스 게이트웨이의 실현 시스템으로 Parlay X 규격의 ECC 웹서비스를 제공하며 AS로부터의 ECC API 요청을 처리한다.
- User A' s phone, User B' s phone, User C' s phone, User D' s phone : 시나리오에 등장하는 사용자들이 소유한 단말로서, 유선 단말, 모바일 단말, VoIP 단말이 모두 가능하나 본 시나리오에서는 SIP 단말로 가정한다.
- ARS (Audio Response system) 서버 (미디어 서버) : 음성안내 등의 부가 서비스를 제공하기 위한 시스템이다.

4.1 세션 생성

그림 5 (가)의 과정의 일부로 User A의 응용에서 'User B에게 전화걸기'를 클릭했을 때, AS에서

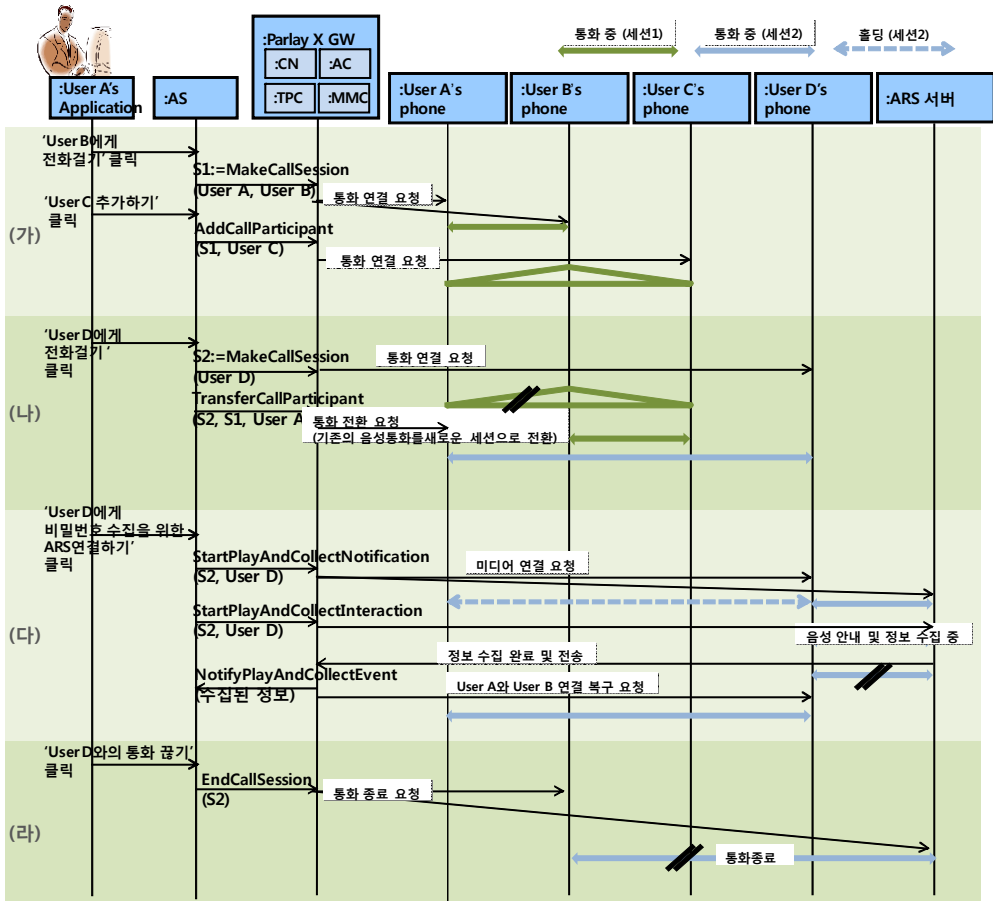


그림 5. 응용 시나리오를 위한 Open API 호출 순서도

MakeCallSession (User A, User B)의 호출로 게이트웨이 시스템에 첫 번째 세션이 생성된다. 그림 6과 같이 게이트웨이 시스템에서는 세션 구분자 S1을 생성하고, 세션과 그의 참가자, 미디어 정보가 생성되어 저장되며, Two-Party (call-leg)를 연결하는 SIP 시그널링을 수행한다. TPC 웹서비스에서 Application Initiated Session의 생성을 위한 세션제어 방법에 따라 INVITE 요청은 IMS 망의 해당 S-CSCF, I-CSCF, P-CSCF를 통해 단말에 전달된다. 게이트웨이 시스템에서는 하부망으로 세션 생성을 위한 SIP 시그널링을 시작하면서 응용에 세션 구분자 S1을 반환하고, 응용은 S1을 이용하여 세션 생성 결과를 확인하는 GetCallSessionInformation (S1) API를 호출할 수 있다.

4.2 세션 전환

그림 5 (가)의 과정의 일부로 User A의 응용에서 'User C 추가하기'를 클릭했을 때, AS에서

AddCallParticipant (S1, User C)를 호출하여 User C를 초대한다. 그림 7에서와 같이 게이트웨이 시스템에서 세션 S1에 User C를 초대하기 위해 User A와 User B가 연결된 일반 세션 S1을 컨퍼런스 세션으로 전환한다. 컨퍼런스 세션은 미디어의 믹싱과 분배가 제공되는 미디어 서버에 참가자를 연결시키는 방식으로 수행된다. 이때 Parlay X GW 내의 ECC 로직은 미디어 서버에 컨퍼런스 룸 생성을 요청하고, 일반 세션에 존재하던 참가자들을 우선 Dial-Out 방식^[14]으로 연결시키며, AddCallParticipant로 추가되는 참가자를 마지막으로 컨퍼런스 룸에 연결시킴으로써 컨퍼런스 세션으로 전환한다. 이로써 세션 S1에는 User A, User B, User C 간에 컨퍼런스 통화가 가능하다.

4.3 세션 생성 후 참가자 전환

그림 5 (나)의 과정의 일부로 User A의 응용에서 'User D에게 전화하기'를 클릭했을 때, 우선 AS에서

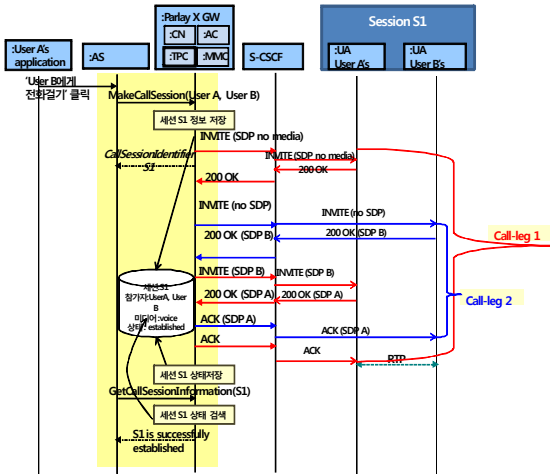


그림 6. 세션 생성을 위한 IMS SIP 세션 제어 절차

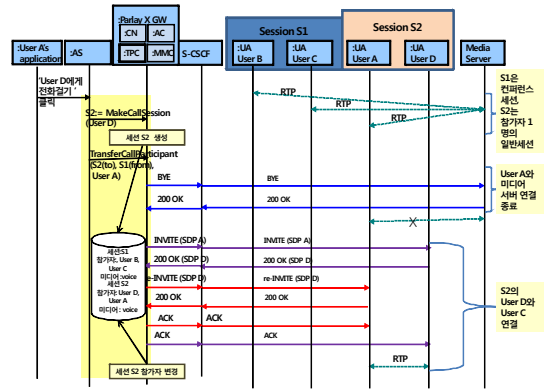


그림 8. 세션 간 참가자 전환을 위한 IMS SIP 세션 제어 절차

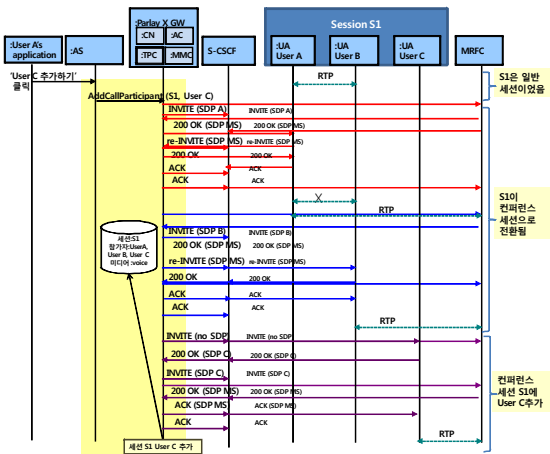


그림 7. 일반 세션에서 컨퍼런스 세션으로 전환하기 위한 IMS SIP 세션 제어 절차

User D만의 세션을 만들고, User A와의 연결을 대기 시킨다. 즉, 그림 8에서와 같이 MakeCallSession (User D)의 호출로 게이트웨이 시스템에 두 번째 세션 S2를 생성하고 세션 정보에 User D의 정보를 저장한다. 이 과정에서 게이트웨이 시스템은 다른 참가자가 조인될 때까지 User D 초대를 위한 SIP 시그널링을 하지 않고 기다린다. 세션 S2가 생성된 후, User A를 세션 S2의 User D와 연결하기 위해 세션 S1에서 세션 S2로 전환한다. 그림 8의 AS에서 TransferCallParticipant (S2(To), S1(From), User A)를 호출함으로써 게이트웨이 시스템에 세션 S1의 User A를 세션 S2로 전환하기 위한 SIP 시그널링을 수행한다. 여기서 S1의 참가자 수 감소로 인해 컨퍼런스

스 세션에서 일반 세션으로 전환하는 방식은 시스템의 효율성 측면에서 고려되어야 할 사항이다. 즉, 참가자가 2명 이하의 상황에도 미래에 추가되는 참가자에 대비하여 컨퍼런스 룸을 삭제하지 않고 컨퍼런스 세션을 유지하거나, 현재의 상황에 최적화시켜 컨퍼런스 룸을 삭제하여 일반 세션으로 전환할 수 있다. 전자의 경우, 일반 세션과 컨퍼런스 세션 간의 전환 평풍이 발생하여도 최소의 시그널링만으로 세션을 유지할 수 있으나 유휴 자원을 미리 할당해 둬으로써 자원을 낭비하는 단점이 있다. 후자의 경우에는 자원을 절약하여 더 많은 유효한 컨퍼런스 세션을 지원할 수 있는 반면 세션 간 평풍이 발생할 경우 다수의 SIP 시그널링 트래픽을 유발시킬 수 있는 단점이 있다.

4.4 세션 복원

그림 5 (다)의 과정으로서 'User D에게 비밀번호 수집을 위한 ARS 연결하기'를 클릭하여 User D에게 음성안내 출력과 정보를 수집하는 과정 후 세션을 복원하는 과정이다. 그림 9의 AS에서 StartPlayAndCollectInteraction (S2, User D)의 호출로 게이트웨이 시스템에서는 세션 S2에 있는 User A와 User D 중 User D에게만 음성안내 방송을 내보내고 정보를 수집한다. 이 경우, 게이트웨이 시스템은 User A의 미디어를 홀딩하고 User D만 미디어 서버와 연결함으로써 정보를 수집하며 이 정보를 NotifyPlayAndCollectEvent 호출을 통해 AS에 전달하여 이를 처리하도록 한다. 수집이 완료되면 홀딩되어 있던 User A 미디어를 User D의 미디어와 연결하는 세션 복원을 위한 SIP 시그널링이 필요하며, 이는 NotifyPlayAndCollectEvent로 응용에 결과를 알려준 후 게이트웨이 시스템에서 내부적으로 수행해야 하는 세션 제어 절차이다.

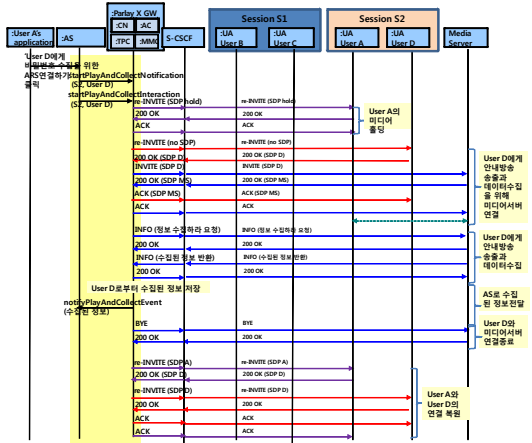


그림 9. 음성 안내와 정보 수집 후 세션 복원을 위한 IMS SIP 세션 제어 절차

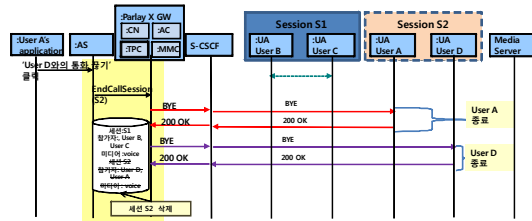


그림 10. 세션 종료 후 세션 종료 후 IMS SIP 세션 제어 절차

4.5 세션 종료

그림 5 (라)의 과정으로 ‘User D와의 통화 끊기’를 클릭할 때, AS에서 EendCallSession(S2)의 호출로 게이트웨이 시스템에서는 세션 S2를 종료한다. 그림 10과 같이 개방형 서비스 게이트웨이는 세션 S2의 참가자에게 모두 BYE를 보냄으로써 참가자와의 연결을 해제하고, 세션을 종료하며 시스템 내의 세션과 참가자 정보를 모두 삭제한다. 이 과정 이후에도 세션 S1의 User B와 User C간에는 음성통화가 가능하다. S1의 종료는, User B와 User C의 단말에서 통화를 종료하거나 User A가 통화를 시작한 응용에서 강제로 종료함으로써 이루어진다.

V. 구현 성능

Parlay X GW에 탑재된 ECC 웹서비스 성능 측정 환경은 그림 11과 같이 동시에 다중 요청 (multiple request)을 발생하는 성능 측정용 로드박스 (Load Box), Parlay X GW, Parlay X GW와 연동하는 IMS 망 시뮬레이터로 구성된다^[15].

성능 측정 절차는 성능 측정용 로드박스에서 발생

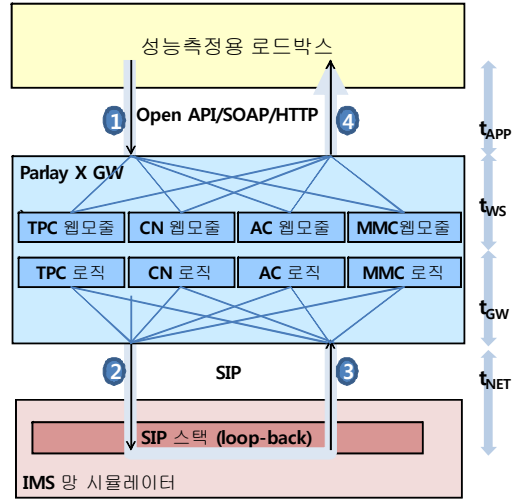


그림 11. 성능 측정 방법

한 하나의 request를 Parlay X GW가 IMS 망으로 보내어 처리 결과를 받아 반환하는 과정 (①~④)으로 구성된다. 성능 측정용 로드박스에서 request를 발생한다는 것은 ECC API의 호출을 의미하며, 곧 통신망에서의 호제어 기능 수행을 내포한다. ECC API 호출은 통신망에서의 호처리 결과를 기다렸다가 반환하는 Synchronous 요청과 요청 후 임의의 시점에 결과를 확인하는 Asynchronous 요청으로 구분된다. Notification 설정/해제와 같이 망의 특정 서버에 정보를 습득/지정/해제하는 API들은 호출 후 통신망의 처리가 완료된 결과를 반환하는 Synchronous 방식으로 제공된다. 반면, 통신망의 자원과 기능을 주로 사용하는 호 진행 관련된 API들은 Asynchronous 요청을 수행한 후 다른 API 호출로 결과를 확인하는 방식으로 제공된다. 이는 Best-Effort 기반의 인터넷 응용 프로그램 등에서 통신망에서의 처리 시간을 기다리지 않고 다른 작업을 수행할 수 있도록 고려된 사항이다. 성능 측정용 로드박스과 게이트웨이 간은 웹서비스 구간으로 SOAP/HTTP로 통신하고, 게이트웨이와 IMS 망 시뮬레이터와는 SIP으로 통신한다. IMS 망 시뮬레이터는 게이트웨이를 IMS 망에 적용시키기 위하여 구축된 시험환경으로서 ②와 ③ 구간의 소요 시간은 하부 네트워크에 의존적이므로 본 성능 측정은 IMS 망 시뮬레이터 입구에서 프로토콜을 루프백 시키는 방법으로 ECC 웹서비스 자체 성능 측정에 초점을 두었다.

ECC 웹서비스 성능 측정은 1 API 호출 시 걸리는 시간 (①~④)을 반복 측정하여 평균 turn around 시간을 측정하는 방법과, 동시에 다중 request를 받

표 1. 성능 측정 대상 API

ECC웹서비스	대표 API	API 요청 유형	IMS 연동 Protocol
TPC	MakeCallSession(A,B)	Asynchronous	SIP
CN	StartNotification(A)	Synchronous	SIP
AC	PlayTextMessage(S,A)	Asynchronous	SIP
MMC	InviteParticipant(S,A)	Asynchronous	SIP

생하여 일정 시간동안 처리된 request의 수를 측정하는 방식을 함께 이용하였다. 1 API 호출 시 구간별 소요 시간 (tAPP, tWS, tGW, tNET)과 총 소요 시간 (tTOTAL) 및 처리 성능 (Throughput)은 다음과 같다.

- tAPP=API를 SOAP 메시지로 인코딩하여 전달하는 시간 + API 수행 결과를 디코딩하는데 걸리는 시간
- tWS=Parlay X GW 내의 웹 모듈에서 SOAP 메시지를 디코딩하여 해당 웹서비스를 호출하는 시간 + 반환할 결과를 인코딩하는 시간
- tGW=해당 웹서비스에서 API를 해석하여, 네트워크 프로토콜로 변환하여 하부망에 전달하는 시간 + 하부망에서 올라온 프로토콜 메시지를 해석하여 결과를 얻는 시간
- tNET=네트워크에서 처리하는 시간
- tTOTAL(①~④) = tAPP+tWS+tGW+tNET
- Throughput (TPS)= 1 sec / tTOTAL

성능 측정용 로드 박스는 PC (Windows XP, Pentium 3, 512M RAM)에서 C#으로 구현되었고, Parlay X GW는 SUN 서버 (Solaris 10, DualUltraSPARC IIIi 1.593GHz, 8GB RAM)에서 Java로 구현되었으며, 웹서비스 플랫폼으로 IBM WAS 6.0를 이용하였다. Parlay X ECC의 성능은 표 1과 같이 웹서비스 별로 대표 API에 대해 초당 처리수 (TPS: Transaction Per Second)로 측정되었으며, SIP를 이용하여 IMS 망 시뮬레이터와 연동하였다.

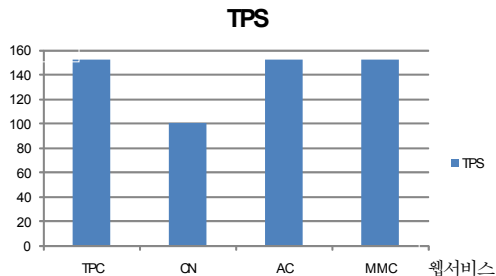


그림 12. 성능 측정 결과

ECC 웹서비스의 성능 측정 결과, 그림 12와 같이 TPC 웹서비스는 최대 152 TPS, CN 웹서비스 최대 101 TPS, AC 웹서비스는 최대 150 TPS, MMC 웹서비스는 최대 152 TPS로 나타났다.

실제로 Parlay X GW가 통신망에 적용되었을 경우 통신망 내에서의 소요 시간이 본 측정 결과에 추가되어야 한다. 그러나, 요청 후 임의의 시점에 결과를 확인하는 Asynchronous 요청은 통신망 처리 시간을 고려하지 않아도 되므로 본 측정에서의 TPC, MMC, AC 웹서비스의 성능은 통신망에 적용했을 때와 같다. 반면, CN 웹서비스와 같이 통신망의 처리 결과를 기다렸다가 반환하는 Synchronous 요청은 1 request(call) 마다 통신망 처리 시간을 추가하여 고려해야 한다. 만약 Parlay X GW가 100만 BHCA (Busy Hour Call Attempts) 급 교환기와 연동할 경우 (3.6 ms/1 call), CN 웹서비스 1 request (9.9 ms/1 call) 처리 시간은 총 13.5 ms/1 call로 예측된다. 이는 교환기 처리 성능에 ECC API 처리로 부과되는 시간을 합산하여 고려할 때, 1 초당 74 call (266,400 BHCA)를 처리함을 가능할 수 있다. 즉, ECC의 최저 성능 (CN 웹서비스 성능)을 고려하더라도 100만 BHCA 교환기와 연동하여 기존의 성능 대비 25% 이상의 성능을 제공할 수 있는 것으로 나타났다.

VI. 결 론

본 논문에서는 통신망의 기능들을 단위 서비스화하여 웹서비스로 표준화한 개방형 서비스 인터페이스 기술을 분석하고, 그 중 확장 호 제어 인터페이스에 대하여 IMS 망에서의 세션 제어 방법을 제안하였다. 제 3자 호 제어, 네트워크 발생 호 제어, 호 진행 중 멀티미디어 인터랙션 등을 위한 세션 생성, 종료, 복원, 전환 과정에서의 SIP 세션 제어 방법을 구체적으로 제안함으로써 추후 IT 융합 응용에 이용될 다양한 호제어 API의 호출에 따른 세션 제어의 가이드라인을 제시하였다. 또한 호제어를 수행하는

개방형 서비스 게이트웨이의 성능 측정을 통하여 기존 네트워크의 호 처리 성능을 크게 저하시키지 않고 IT 망에 적용될 수 있음을 보였다.

통신 사업자의 자원으로 관리되던 통신망의 기능이 웹서비스 기반의 표준 인터페이스로 개방됨으로써 IT 응용에서 다양한 부가 서비스 개발에 용이하게 활용될 수 있으며 IMS 기반의 차세대 통신망에서 IT와 통신 융합의 첨병 역할을 수행할 것으로 기대된다. 추후 연구에서는 CORBA 기반의 Open API와 웹서비스 기반의 Open API의 처리 성능을 비교하고자 한다.

참 고 문 헌

[1] Ard-Jan Moderdijk, "Open Service Architecture: Concepts and Standards," *Ericsson Technical Report*, 2001.

[2] Jean-Yves Cochenec, "Activities on next-generation networks under global information infrastructure in ITU-T," *IEEE Communications Magazine*, 40(7), pp. 98-101, Jul. 2002.

[3] Jyh-Cheng Chen and Tao Zhang, *IP-Based Next-Generation Wireless Networks : Systems, Architectures, and Protocols*, John Wiley & Sons, 2004.

[4] Roch H. Glitho, "Developing Applications for Internet Telephony: A Case Study on the Use of Parlay Call Control APIs in SIP Networks," *IEEE Network*, 18(3), pp. 48-55, May/June 2004.

[5] Nikolaos D. Tselikas, Nikolaos L. Dellas, Eleftherios A. Koutsoloukas, Sofia H. Kapellaki, George N. Prezerakos, and Iakovos S. Venieris, "Distributed service provisioning using open APIs-based middleware: "OSA/Parlay vs. JAIN" performance evaluation study," *The Journal of Systems and Software*, pp. 765-777, 2007.

[6] 3GPP TS 29.199-2, "Open Service Access (OSA); Parlay X 2 Web Services; Part 2: Third Party Call".

[7] 3GPP TS 29.199-3, "OSA; Parlay X Web Services; Part3:Call Notification".

[8] 3GPP TS 29.199-11 V7.1.0, "Open Service Access (OSA) Parlay X Web Services Part11: Audio Call".

[9] 3GPP TS 29.199-12 V7.0.0: "Open Service Access (OSA); Parlay X Web Services; Part 12: MultiMedia Conference".

[10] 3GPP TS 23.002, "Network Architecture".

[11] 3GPP TS 23.228, "IP Multimedia Subsystem (IMS); Stage 2".

[12] 3GPP TS 23.218, "IP Multimedia (IM) session handling; IM call model; Stage 2".

[13] IETF RFC 3261, "SIP:Session Initiation Protocol".

[14] 3GPP TS 24.229, "IP multimedia call control protocol based on Session Initiation Protocol (SIP) and Session Description Protocol (SDP); Stage 3".

[15] sipping-cc-framework-07, "A Call Control and Multi-party usage framework for SIP".

박 유 미 (Yoo-mi Park)

정회원



1991년 2월 숙명여자대학교 전산학과 졸업
 1997년 8월 충남대학교 컴퓨터과학과 석사
 1991년 3월~현재 한국전자통신연구원 책임연구원
 <관심분야> 개방형 서비스 기술, 개인화 서비스, SIP

임 종 철 (Jong-chul Yim)

정회원



1997년 2월 서울시립대학교 전산학과 졸업
 2000년 2월 서울시립대학교 전산학과 석사
 2000년 10월~현재 한국전자통신연구원 선임연구원
 <관심분야> 개방형서비스기술, SIP

최 영 일 (Young-il Choi)

정회원



1983년 2월 서울대학교 전자공
학과 졸업
1998년 2월 충남대학교 컴퓨터
과학과 석사
2002년 3월 충남대학교 컴퓨터
과학과 박사
1996년 정보통신기술사

1985년~1986년 Bell 연구소 연구원

1983년~현재한국전자통신연구원 개인화서비스플랫폼
연구팀장

<관심분야> NGN, 차세대 통신서비스

김 상 하 (Sangha Kim)

중신회원



1980년 2월 서울대학교 화학과
학사
1984년 6월 Univ. of Huston
석사
1989년 6월 Univ. of Huston
박사
1992년 2월~현재 충남대학교
컴퓨터공학과 교수

<관심분야> 무선/이동 QoS, 무선 멀티캐스트