

제3회 정보통신표준화 우수논문집

장려상

7가지 상태 전이를 기반한 SIP 신호처리

A SIP Signal Processing based on the Seven States Transition

신용경

LG전자 소프트웨어센터

Yong-Kyung Shin

S/W Center, LG Electornics

I. 서론

II. SIP 인터넷 전화연결과정

III. Seven State 기반 전화연결

IV. 시스템 검증

V. 결론

7가지 상태 전이를 기반한 SIP 신호처리

A SIP Signal Processing based on the Seven States Transition

신용경 / LG전자 소프트웨어센터

Yong-Kyung Shin / S/W Center, LG Electornics

| 요약 |

본 논문은 기존의 표준인 RFC3261 SIP 프로토콜의 취약한 신호처리방식을 보완하여 IP전화통신에서 좀 더 효율적으로 전화연결 서비스를 설계할 수 있도록 제시한다. 사용자의 요구에 따라 수시로 변하는 7가지의 상태를 기반으로 하여 세션을 연결함으로써 안정적인 SIP 신호처리를 할 수 있다. 기존의 RFC3261에서 제시하는 Call-Setup과정을 따라 인터넷 전화연결 시스템을 설계하면 예외처리 및 과도한 트래픽이 발생하여 시스템 오류를 불러 올 수 있다. 본 논문에서는 시스템의 State를 보고 예상된 이벤트일 경우 정형화된 처리루틴으로 대처하고 그렇지 않을 경우도 예외상황을 효율적으로 처리 할 수 있는 표준 시그널 처리기법을 제시한다. 제시기법을 6 Sigma의 DMAIC process로 증명하여 보인다.

I. 서론

IP 텔레포니 기술이 발전하면서 SIP 프로토콜을 이용한 전화연결 서비스가 보편화되고 있다[1]. 현재 상용화되고 있는 인터넷 전화에 이용되는 프로토콜로는 SIP와 H.323이 있는데 SIP가 text를 기반으로 하여 H.323보다 복잡도가 낮고, 확장성이 높아 최근에는 SIP를 선호하는 추세이며[12, 13], 따라서 다양한 서비스를 개발하기 위한 SIP구성요소의 확보가 중요하다[12]. RFC 3261에서 제시하는 SIP 프로토콜은 security문제 뿐만아니라 상용화 수준으로 개발하기는 많은 구성요소들이 보완이 되어야 하며, 현재도 RFC 3311("SIP UPDATE Method")

이나 RFC 3313("Private SIP Extensions for Media Authorization")등과 같이 SIP 프로토콜의 구성요소를 보완하는 작업을 IETF에서 계속해서 진행 중에 있다[5~7]. 이와 같이 SIP 프로토콜이 상용 망에서 많이 이용되지만 아직까지 문제점들을 많이 드러내고 있다고 볼 수 있다[9, 11]. SIP 프로토콜을 이용하여 상대방에게 전화를 걸거나, 상대방으로부터 전화가 걸려오는 경우 또한 여러 가지 예외상황이 발생할 수 있다. 과 트래픽 전송문제, 망 오버헤드 문제, 패킷로스 및 재전송 등의 예상치 못한 문제에 대해서 일일이 기술하여 캐치하기는 개발자들에게는 어렵다[4, 8]. 또한, 이와 관련된 대처방안을 제시하고 있는 RFC문서는 존재하지 않는다.

본 논문에서는 사용자가 전화를 걸거나 받을 때 7가지 State를 기반으로 Event가 발생하면 그에 적합한 수식을 적용하고 어떻게 효율적으로 전이하는지를 보인다.

본 장에 이어서 제 2장에서는 SIP 인터넷 전화연결과정을 보이고, 제 3장에서는 Seven State기반 전화연결 과정을 보이고, 각 state를 분류하는 기준 및 Event가 들어오면 state가 어떻게 전이하는지를 설명한다. 제 4장에서는 위의 내용을 시스템에 적용하여 검증한 결과를 보이고, 제 5장에서는 결론 및 향후 연구 계획을 기술하고자 한다

II. SIP 인터넷 전화연결과정

본 논문은 Seven States를 두어 전화를 거는 단말(Mobile Originator, 이하 MO라 함)과 전화를 받는 단말(Mobile Terminator, 이하 MT라 함)에서 발생할 수 있는 예외상황을 효율적으로 처리하는 과정을 설명한다. RFC3261에서 제시하는 Proxy Server나 Relay Server에 의해 나타날 수 있는 MO와 MT의 에러상황을 제외하고는 Server 시스템 내부 에러상황에 관련된 내용은 본 논문의 범위를 벗어나므로 다루지 않는 것으로 한다

2.1 기존 표준에서 제시하는 전화연결 과정

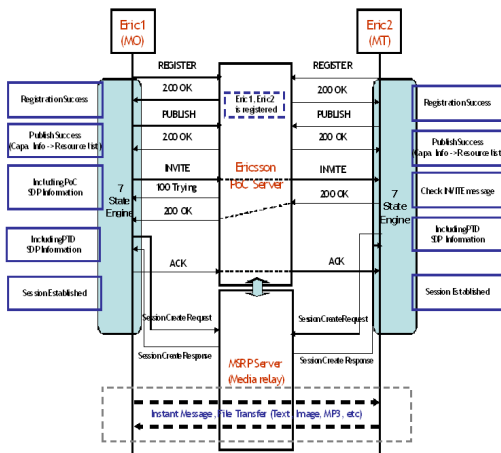
(1) MO의 전화연결 과정

MO는 REGISTER과정을 통하여 Proxy Server에 주소 및 Session정보를 등록한다. 등록과정이 성공적으로 끝나면 사용자는 연결하고자 하는 상대방을 선택하여 전화연결을 시도하는데 MO는 현재의 상태정보를 가지고 'INVITE' Method를 text기반으로 encoding하여 proxy 서버에 보내면 상대방의 상태에 따라 '100 Trying'이 오거나 '200 OK' response message가 온다. 같이 전달되는 SDP message를 해석하여 Call-Setup이 성공하였는지 실패하였는지 아니면 내가 가지고 있는 상대방의 미디어정보 및 주소정보를 변경해야 하는지를 판단하게 된다. SDP정보를 해석하여 적합한 미디어 채널을 열어 음성이나 영상을 주고받는다. 전화연결을 끊을 경우는 MO이든 MT이든 관계없이 한쪽에서 'BYE' request message를 생성하여 보내고 '200 OK' response message를 받으면 미디어 세션 및 SIP 인터넷 전화연결이 끊어진다.

(2) MT의 전화연결 과정

MT도 REGISTER과정이 우선되어야 하며, 등록과정이 성공적으로 끝나면 상대방으로부터 전화가 걸려온다. 상대방이 연결 시도하는 채널의 형태가 음성인지 영상인지에 따라 MT는 encoding및 decoding가능한지를 결정해야 한다. [1]에 따르면 MO가 실어 보내는 SDP의 session정보를 해석하여 미디어 채널 생성이 가능한지를 판단하여 '200 OK' response message를 보내고 그렇지 않을 경우 'Request Failure 4xx', 'Sever Failure 5xx', 'Global

Failure 6xx'중에서 MT의 상태를 알릴 수 있는 적절한 Response message를 선택하여 보낸다. '200 OK'를 보낸 경우 MO가 잘 받았는지 확인하기 위하여 'ACK' request message를 기다린다. 'ACK'를 받으면 미디어를 위한 채널을 열어 데이터를 주고 받는다

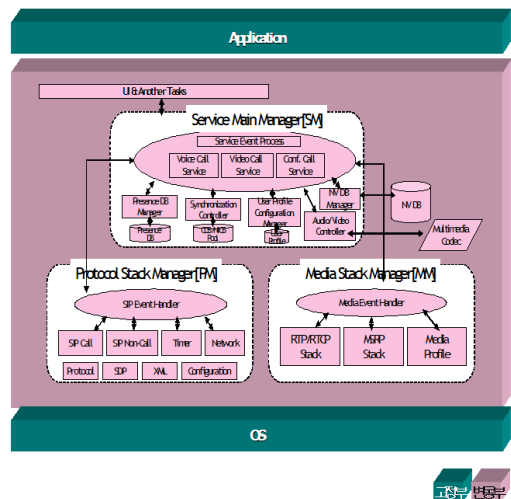


(그림1. SIP 1:1 인터넷 전화연결 과정)

2.2. SIP 인터넷 전화연결 시스템 구조

본 논문에서 제시하는 Seven States를 설명하기 이전에 적용되는 시스템의 구조를 우선 이해하여야 한다. 그림2는 Seven States를 이용한 SIP인터넷 전화연결 시스템 구조를 보인다. 크게 3개의 모듈로 구분하며, UI와 다른 Task와의 핸들링을 담당하면서 하부 모듈에서 발생하는 이벤트를 받아 처리하는 Service Main Manager(이하 SM이라 함)가 있고, 전화연결을 위하여 SIP 프로토콜을 encoding/decoding하는 프로토콜 관련 Stack이 필요하다. 이를 관리하는 Protocol Stack Manager(이하 PM이

라 함)가 있다. 마지막으로 전화연결 프로토콜을 이용하여 통신이 이루어지면 음성이나 영상과 같은 미디어 전송이 필요하다. 이를 Media Stack Manager(이하 MM이라 함)에서 관리한다.



(그림2. SIP 인터넷 전화연결 시스템구조)

2.2.1 Service Main Manager[SM]

SM은 UI와 다른 System task들과의 Interface 역할을 하며 SIP call과 연관된 이벤트를 생성 및 처리하고 사용자 정보관련 DB를 관리한다. Service Event Process와 Presence DB Manager, Synchronization Controller, User Profile Configuration Manager, NVDB(Non-Volatile DB) Manager와 Audio/Video Controller를 가지고 있어 각 DB와 Multimedia Codec와 연계하여 사용자 요구사항을 만족시킨다.

■ Service Event Process

UI에서 들어오는 정의된 이벤트를 해석하여 다른 모듈에 전달하고, 다른 모듈에서 들어오는 이벤트를 해석하여 UI로 결과를 알린다. SIP로 전화 연결하여 원하는 서비스를 Application이라 하면 Service Event Process는 어떤 Application을 위한 SIP 전화 연결인지를 구분하여 관리한다.

■ Presence DB Manager

Buddy의 Presence 및 Group 정보를 저장 관리한다. Buddy의 Presence 정보로는 통신 상태(Status), 그룹 정보, 이름, 성별, 기호 등이 있으며, Application의 요구 사항에 따라 이곳에 Presence정보를 추가할 수 있다. Buddy의 Group 정보로는 Group 이름, Group ID 등이 있으며, 마찬가지로 Application의 요구 사항에 따라 이곳에 Group 정보를 추가할 수 있다. DB 형태로 저장되는 Buddy DB, Group DB를 관리할 수 있는 Function 들로 구성되어 있다. 또한 DB의 Access, Delete, Add, Update를 위해 Linked-List Mechanism을 응용하여 적용한다.

■ Synchronization Controller

Overlap Transaction, Mid-Call Transaction을 처리한다. 동시에 여러 개의 Transaction 또는 Dialog를 구별하고, 동시에 처리되는 각각의 Transaction 또는 Dialog들 상호간의 독립성을 최대한 유지하는 역할을 한다.

다.

■ User Profile Configuration Manager

Application 별로 Provisioning Profile Data를 저장, 초기화, 관리한다. 예를 들면 Authentication Parameter, Registration(Subscription) Expire Value를 관리한다. PM에서 이 Profile Data를 이용하여 Call Object의 초기 Data를 구성한다.

■ NVDB Manager

Server URI/IP Address, Service Enable/Disable Parameter, Receiving Mode, Login Process 등을 결정하는 Non-Volatile Data를 관리한다.

■ Audio/Video Controller

멀티미디어 코덱을 관리한다.

2.2.2 Protocol Stack Manager [PM]

Protocol Stack Manager내에서 SIP Dialog 또는 SIPCall Procession을 위해 사용되는 Call Object를 관리한다. RFC 3261에서 제시하는 SIP기본 mechanism을 사용하고, SM에서 전달받은 사용자 정보를 기본으로 하여 Call을 생성한다. Network Manager가 있어 PPP Connection 설정, 해제, 상태체크 등을 수행하고 SIP를 위한 Socket Control 및 SIP Message Sending, Receiving을 수행한다. SDP Manager에서 Media 전송 및 수신에 위하

여 Message Forming 및 Negotiation을 수행한다.

2.2.3 Media Stack Manager [MM]

SIP Call 연결이 설정되면 두 단말은 데이터를 주고 받는다. 이때 데이터로는 음성, 이미지, 동영상 등 다양한 형태를 지원한다. SIP 프로토콜로는 Call 연결만 가능하고 데이터를 위한 채널은 다른 프로토콜을 이용하여야 한다. MM은 데이터를 주고 받기 위한 추가적인 프로토콜을 관리한다. 본 논문에서 제시하는 데이터 교환 프로토콜로는 MSRP(Message Session Relay Protocol)[2]를 사용한다. Call 연결시 SDP[3]로 데이터 교환을 위하여 어떤 프로토콜을 이용할 것인지 정의하고 포트번호를 negotiation한다. 서로간의 네트워크 채널이 정상적으로 열리면 MSRP 형식으로 데이터를 만들어 전달하고, 이벤트 및 전송상태를 MM에서 체크한다. SIP Call이 해제되면 MSRP 채널도 자동으로 닫혀야 하므로 SM에서 실시간으로 이벤트를 주고 받을 수 있다.

III. Seven State 기반 전화연결

3.1 SIP전화연결시 발생하는 이벤트 정의

SIP 전화연결을 위해서 Application은 이벤트를 통하여 상태전이를 한다. 7개의 상태를 설명하기 앞서 신호처리상에서 발생하는 이벤트를 미리 정의하여 사용한다. 본 논문에서 처리되는 이벤트 naming rule은 다음과 같이 정의한다.

(대분류)_(중분류)_(접미사)

(1) (대분류) : 이벤트를 처리하는 주체

SM|PM|MM

- SM : 그림 2에서의 Main Manager
- PM : 그림 2에서의 SIP Stack Manager
- MM : 그림 2에서의 Media Stack Manager

(2) (중분류) : 의미 전달이 명확하도록 이벤트의 Operation에 대한 동사나 명사

(3) (접미사)

- 반드시 짝을 이루어야 하는 경우에 대한 이벤트 postfix

- i. REQ(REQuest): Upper Layer에서 Lower Layer로 특정 Operation을 요청하는 경우
- ii. RES(RESponse): REQ에 대한 Operation을 완료한 이후에 대한 응답을 보내는 경우
- iii. RPT(RePorT): Lower Layer에서 Upper Layer로 특정 Operation을 요청하는 경우
- iv. CNF(CoNFirm): RPT에 대한 처리가 완료되었음을 Lower Layer로 다시 알려주는 응답을 보내는 경우

- 짝을 이루지 않고 홀로 존재하는 이벤트 postfix

- i. IND(Indication): Lower Layer에서 Upper

Layer로 특정 상황을 알리고, 이에 대한 추가 이벤트 primitive 교환이 필요하지 않은 경우

- ii. CMD(Command): Upper Layer에서 Lower Layer로 특정 상황을 알리고, 이에 대한 추가 이벤트 primitive 교환이 필요하지 않은 경우

3.2 Event

위의 naming rule에 따라 SIP인터넷 전화연결 시 발생 하는 이벤트를 정의하면,

(1) SM_ESTABLISH_CMD

발신자 요구에 따라 SIP전화연결 요청이 시작되는 이벤트

(2) SM_END_CMD

발신자가 전화연결을 해제하는 이벤트, 전화 연결시도 중에서 이벤트 발생이 가능함

(3) SM_INCOMING_RPT

수신자에게 전화가 왔음을 알리는 이벤트

(4) SM_INCOMING_CNF

수신자가 걸려온 전화에 승인 응답을 주는 이벤트

(5) SM_RELEASED_IND

상대방에 의해 전화연결이 해제되었음을 User에게 알리는 이벤트

(6) PM_SENDINVITE_CMD

‘INVITE’ message를 만들어서 수신 측에 전달하도록 요청하는 이벤트

(7) PM_FINALRESPONSE_IND

수신 측으로부터 ‘INVITE’에 대한 response message가 왔음을 알리는 이벤트

(8) PM_END_REQ

BYE message를 만들어 상대방에게 전달하도록 요청하는 이벤트

(9) PM_END_RES

수신 측으로부터 BYE에 대한 response message가 왔음을 알리는 이벤트

(10) PM_ACK_CMD

‘200 OK’ response를 수신하였음을 알리기 위하여 ACK message를 만들어서 수신 측에게 전달하도록 요청하는 이벤트

(11) PM_ACK_IND

발신 측에서 ACK message가 왔음을 알리는 이벤트

(12) PM_RECEIVEINVITE_IND

발신 측에서 ‘INVITE’ message가 왔음을 알리는 이벤트

(13) NALRESPONSE_CMD

‘INVITE’ message를 수신하고 response를 상대방에게 전달하도록 요청하는 이벤트

(14) PM_END_RPT

발신 측에서 BYE message가 왔음을 알리는 이벤트

(15) PM_END_CNF

BYE message에 대한 response를 상대방에게 전달하도록 요청하는 이벤트

(16) PM_SUBSCRIP_CMD

서버에게 SUBSCRIBE message를 보내도록 요청하는 이벤트

(17) MM_SESSIONCREATE_REQ

Multimedia Data를 받기 위하여 Session을 생성하도록 요청하는 이벤트

(18) MM_SESSIONCREATE_RES

Session 생성결과를 알리는 이벤트

(19) MM_SESSIONDEL_CMD

Multimedia Data를 받거나 주기 위하여 생성한 Session을 제거하도록 요청하는 이벤트

다음절의 Seven States에서 제시하는 각 State의 전이는 다음의 이벤트를 기본으로 설명하고 있다. 이벤트 설명에서 발신자는 전화를 거는 사람, 수신자는 전화를 받는 사람, 발신 측은 전화를 거는 단말(MO), 수신 측은 전화를 받는 단말(MT), 상대방은 발신 측 혹은 수신 측을 말한다.

3.3 Seven States

본 논문에서 제시하는 Seven States는 NULL, CLEARING, ESTABLISHING, WAITMEDIAANSWER, WAITUSERANSWER, RESPONDING, ACTIVE이다. 각 State의 상태전이도는 그림 3에서 보인다.

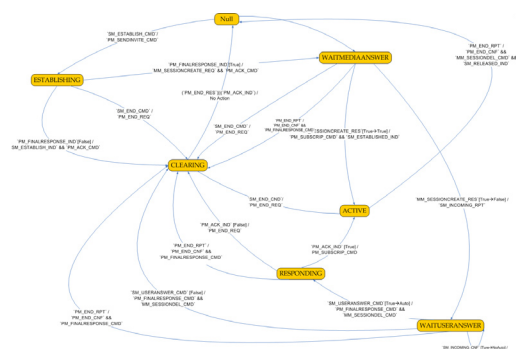
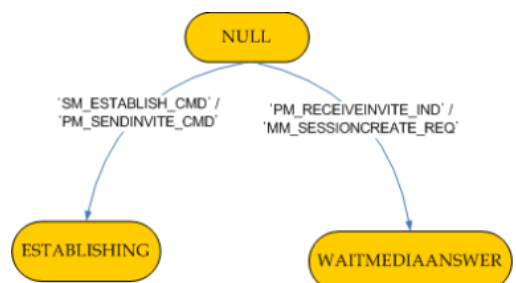


그림 3. 7 State 상태전이도

(1) NULL

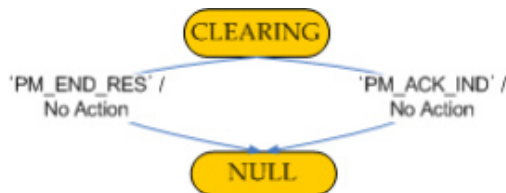
MO와 MT가 Proxy Server에 등록을 거치고 유효한 사용자 인증을 받은 후 초기화 되는 state이다. 단말은 NULL상태에서 'INVITE' request를 보내는 쪽이 MO가 되고 받는 쪽이 MT가 된다.



(그림4. NULL state의 전이과정)

(2) CLEARING

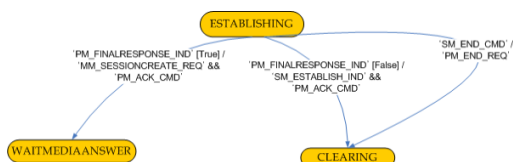
RFC 3261에서 제시하는 SIP Signaling은 ‘200 OK’를 보내고 나서도 ‘ACK’를 받아야지만 최종적으로 미디어 세션이 열린다. ‘200 OK’가 아닌 시스템상의 문제로 연결이 실패할 경우라도 Failure message를 보내고 ‘ACK’를 받아야 한다. 그렇지 않을 경우 재전송 해야 한다. 따라서 전화연결이 실패할 경우 이전 state에서 NULL상태로 전이하게 되면 ‘ACK’를 받았는지 확인 할 수 없으므로 상대방의 시스템이나 서버에 치명적인 error를 일으킬 수 있다. 따라서 CLEARING 상태로 우선 전이 한 다음 NULL상태로 전이하기 위한 처리를 반드시 해야 한다.



(그림5. CLEARING state의 전이과정)

(3) ESTABLISHING

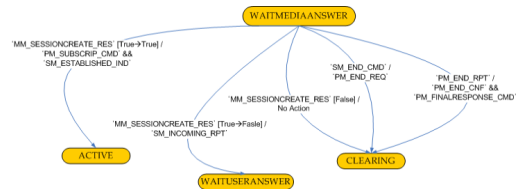
MO가 상대방에게 전화를 걸어 연결의사를 묻고 있는 상태를 말한다. MO의 미디어 정보를 SDP에 실어 SIP Signal과 함께 보내면 MT로부터 연결 가능여부를 알려준다. MO는 ESTABLISHING 상태에서 MT의 결과값을 해석하여 다른 State로 전이한다.



(그림6. ESTABLISHING state의 전이과정)

(4) WAITMEDIAANSWER

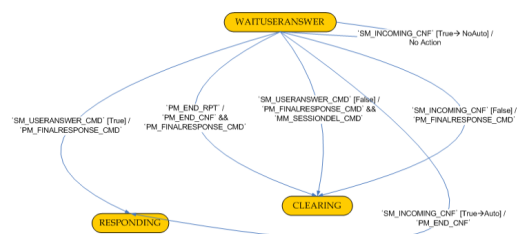
상대단말에서 요구하는 미디어 채널의 capacity나 코덱 정보 및 performance로 내 단말과 통신가능한지를 판단하는 상태를 말한다. MO는 ‘200 OK’ response message와 함께 전달되고, MT는 ‘INVITE’ request message에 실패 온다. 따라서 MO는 ESTABLISHING 상태에서 ‘200 OK’가 오면 WAITMEDIAANSWER로 전이하고 그렇지 않은 경우 CLEARING으로 전이한다.



(그림7. WAITMEDIAANSWER state의 전이과정)

(5) WAITUSERANSWER

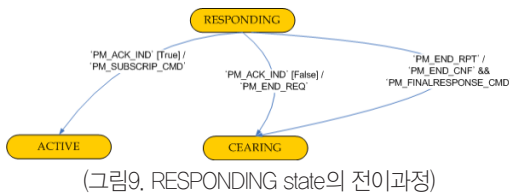
MT는 상대방에서 요구하는 미디어 정보로 미디어 채널 연결이 가능하더라도 최종사용자 (End-User)에게 전화연결을 할 것인지 의사결정을 요구해야 한다. 최종사용자가 상대방의 전화를 받을 의사가 없다면 MT는 ‘200 OK’가 아닌 다른 Failure message를 생성하여 상대방에게 보내야 한다.



(그림8. WAITUSERANSWER state의 전이과정)

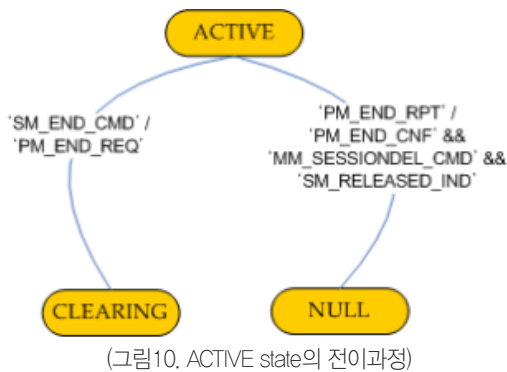
(6) RESPONDING

MT가 '200 OK'를 보내고 나서 MO로부터 'ACK'를 받아야지 미디어 채널을 열 수 있다. '200 OK'를 보냈지만 네트워크나 서버의 error로 signal데이터가 손실 되어 MO가 받지 못하였을 경우 MT는 몇 차례 재전송 한 후 timeout된다. 이때 CLEARING으로 전이한다.



(7) ACTIVE

MO와 MT가 음성이나 영상과 같은 미디어 데이터를 주고 받을 수 있는 상태이다. 또한, SIP signal중 'BYE' request를 이용하여 전화 연결을 해제할 수 있는 상태이다.



3.4 각 State별 이벤트처리 방법

각 State는 FSM(Finite State Machine)으로 관리한다. FSM은 유한한 개수의 상태들을

가진 하나의 기계이고, 그 상태들 중 하나가 현재 상태인 것이다. FSM은 입력을 받고 어떠한 상태전이함수에 기반해서 현재상태로부터 출력상태로의 상태전이를 일으킨다. 그리고 출력상태는 새로운 현재 상태가 된다. FSM table을 두어 현재 상태에서 유효한 이벤트를 정의해 두어 해당 이벤트가 입력되면 조건에 맞는 상태로 전이한다. FSM에서는 이전의 상태를 읽어 올 수도 있고, Preprocessing Table을 두어 현재 상태에서 우선적으로 처리되어야 하는 이벤트를 실시간으로 받아 들여 시스템의 치명적인 오류를 막을 수 있다. 즉, User Termination이거나 Network Close에 관련된 이벤트는 모든 상태에서 미리 정의되지 않았더라도 우선적으로 시스템에서 처리 및 안정적인 복구를 필요로 한다. 따라서 FSM의 Preprocessing Table에서 처리되도록 하는 것이 바람직하다.

실시간으로 들어오는 이벤트 처리에 있어 CLEARING state는 복잡한 SIP signal을 효율적으로 관리할 수 있도록 한다. MO가 전화를 걸어 MT가 OK응답을 보냈더라도 SIP signal상으로 MO로부터 ACK가 오지 않으면 Call은 성공적으로 연결된 것이 아니므로 CLEARING state로 전이하여 Call Fail에 대한 signal처리를 마무리한다. MO나 MT에서 갑작스런 시스템 오류로 인하여 Call-setup중에 END를 요청하는 경우도 CLEARING에서 처리하도록 한다.

3.5 7 State를 기반한 SIP신호 처리

MO와 MT가 Proxy Server에 REGISTER

과정을 끝내고 전화연결의 유효한 상태가 되면 NULL로 초기화 된다. MO가 원하는 상대를 선택하여 MT에 전화를 걸면 'INVITE' request를 보내고 ESTABLISHING State가 된다. MT는 이를 받고 요구되는 미디어 정보를 해석하여 MO와 미디어 채널 연결 가능 여부를 판단한다. 이때 MT는 WAITMEDIAANSWER state가 된다. MT가 해석할 수 없는 미디어 채널 혹은 미디어 코덱을 요구할 경우 MT는 Failure message를 보내야 하므로 CLEARING state로 간다. NULL상태로 바로 가지 않고 CLEARING state에서 SIP signal특성상 주고 받아야 하는 message를 걸러주어 시스템에 예외상황을 막는 역할을 한다. MT가 해석 가능한 미디어 채널을 요구할 경우 자신의 시스템 정보를 SDP message에 실어서 response message를 보내고 MT는 RESPONDING state로 전이한다. 'INVITE'에 대해 '200 OK' response를 보냈더라도 ACK message를 받아야지 Call 연결이 성공하기 때문에 RESPONDING state에서 MO로부터의 ACK message를 기다린다. 한편 MT로부터 response를 받은 MO는 ACK message를 보낸 후 '200 OK'일 경우 미디어 채널을 생성 및 해제 관리하는 MM모듈에 세션 생성 이벤트를 보내어 데이터를 받을 준비를 한다.

IV. 시스템 검증

4.1 구현 검증

SIP 프로토콜을 이용하여 전화연결 후 응용

서비스를 하기 위하여 시스템은 안정적으로 설계되어야 하며 어떠한 예외상황에서도 치명적인 오류뿐만 아니라 작은 오류를 일으켜서는 안 된다. 이를 검증하기 위하여 본 논문에서 제시하는 이벤트 기반의 Seven States구조는 WCDMA향 IP기반의 핸드폰 단말과 LG전자에서 자체 개발한 SIP Proxy서버를 인터넷에 물려서 실험하였다. SIP Proxy서버에 단말의 등록을 위한 Registration서버를 함께 돌아가도록 하였다. 단말에서 구현된 서비스로는 OMA(Open Mobile Appliance) Working Group으로 활동하고 있는 PoC(Push to Talk Over Cellular)를 구현하여 OMA-EICS-PoC-Client-V1_0-20050729-D에서 제시하는 Test Items를 실험하여 표 1과 같은 결과를 보인다[15].

OMA-EICS-PoC-Client-V1_0-20050729-D는 OMA에서 제시하는 시스템 안정성 및 기능시험을 할 수 있는 자료로써 industry에서 자사 개발제품을 검증하는 KPI(Key Performance Indicator)로 많이 사용된다.

표 1은 Control Plan에서 검증되어야 하는 Item list이다. OMA-EICS-PoC-Client-V1_0-20050729-D의 Control Plan은 PoC Client Originating Procedures(PoCCPSpec-COP), POC Service Settings procedure(PoCCPSpec-CSP), POC Session Initiation(PoCCPSpec-CSI), POC Session Modification(PoCCPSpec-CSM), PoC Client Re-joining a PoC

Session(PoCCPSpec-CRS), PoC Client Leaving a PoC Session(PoCCPSpec-CLS), PoC Session unrelated operations for PoC Client Initiating Procedures(PoCCPSpec-CUO), PoC Client Terminating Procedures(PoCCPSpec-CTP)로 구성한다.

표1에서 Test Results는 OMA PoC 표준스펙에 맞추어 구현후 실험한 결과를 보인다.

〈표1. OMA Test Results〉

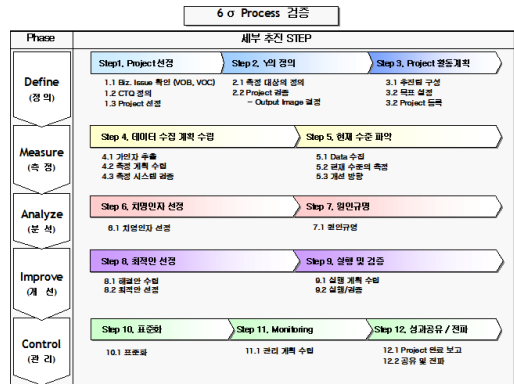
OMA Function	Results
General registration to SP/IP Core	Passed
PoC Service Setting Procedure initiation of SIP PUBLISH request	Passed
Indication of setting results after receiving SIP 200OK response	Passed
General Initiation SIP INVITE request	Passed
SIP Session Timer	Passed

4.2 설계 검증

7 State를 이용한 SIP signal처리 설계는 6σ Process 검증방법을 사용하여 신호처리 구조 및 Seven States를 증명한다. σ는 프로세스의 산포를 나타내는 척도로써 통계적인 용어로 표준편차를 의미한다. 즉 데이터들이 중심으로 부터 전형적으로 떨어진 거리를 나타내며, 6시그마는 규격상한과 규격하한이 있는 경우 단기간적으로 분포의 중심과 규격한계 사이의 거리가 표준편차의 6배나 될 정도로 불량률이 아주 낮은 상태이다. 6σ Process를 통하여 최적합 설계를 찾아내서 통신설정시 불량률을 낮출수 있는 방법을 검증 및 제시한다.

6σ Process에서도 DMAIC기법을 통하여 설계검증을 하면 그림 11과 같은 STEP으로 구

체화 할 수 있다.



(그림 11. 6σ Process 검증 Step)

4.2.1 Define

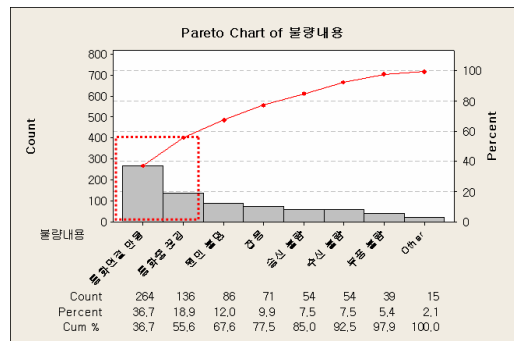


그림 12. 통신 시장 불량현황 분석(Pareto분석)

Define 단계는 사용자 및 개발자 관점에서 중요하고, System설계상의 Impact가 가장 큰 Project를 도출하고, 이를 성공적으로 추진하기 위하여 Project를 명확하게 정의하는 단계로써, 개선의 가능성이 있는 대상을 명확히 정의할 수 있도록 하는 단계이다. 따라서 통신 시장의 불량 현황을 분석하면 그림 12와 같은데, RF Module시장 불량 점수 Data분석결과 휴대장치 불량중 SW적인 문제가 98%를 점유하고 있고,

통화 연결 안됨, 통화중에 끊김 불량이 약 58%를 점유하고 있는 실정이다.

따라서, 통화가 안정적으로 연결 될 수 있는 핵심 기술이 필요하다. RFC3261 표준 프로토콜로는 시장의 복잡한 환경 대응이 불가능하여 Project Y의 값을 ‘상태 기반 SIP프로토콜 지원 필요’로 설정한다.

4.2.2 Measure

Measure 단계는 선정된 CTQ의 현재 수준을 측정하여 Base line을 설정하고, CTQ에 대한 개선 목표 및 방향을 설정하는 단계이다. 기존의 상태기반으로 통화설정 되는 시스템을 측정하여 프로세스의 현재 수준을 파악하여 개선의

방향과 정도를 결정한다.

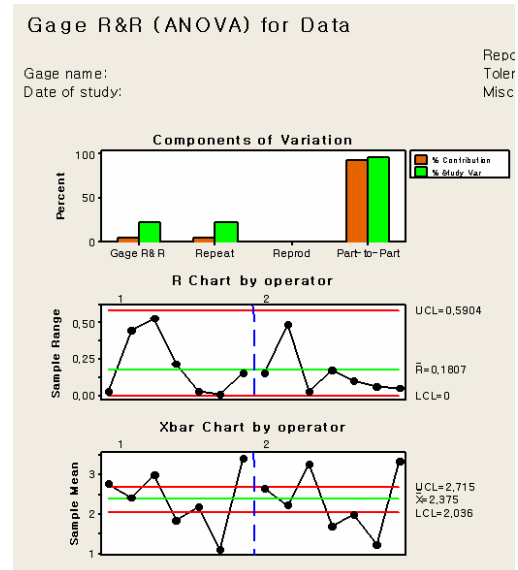
표 2에서 Parts는 통화설계모델, Operator는 시험환경, Data는 Signal 불량률을 나타낸다.

표 2의 실험 Data로 Gage R&R을 통하여 나온 Graph는 그림 13와 같다. 그림 13에서 Components of Variation은 고려된 항목별 변동 비중을 표현하고 있는데 Gage R&R의 비중이 Part-to-Part에 비해 작으므로 바람직하다. R Char by operator는 개별 시험 환경에서 측정값의 범위가

Parts	operator	Data
1	1	2.75
1	1	2.78
2	1	2.65
2	1	2.19
3	1	2.75
3	1	3.29
4	1	1.97
4	1	1.75
5	1	2.19
5	1	2.16
1	2	2.75
1	2	2.59
2	2	2
2	2	2.5
3	2	3.25
3	2	3.28
4	2	1.77
4	2	1.59
5	2	1.96
5	2	2.06
6	1	1.12
6	1	1.13
7	1	3.53
7	1	3.37
6	2	1.19
6	2	1.25
7	2	3.37
7	2	3.32

〈표2. 실험 Data Resume〉

모두 관리한계선 내에 있고, Xbar Chart by operator는 개별 시험 환경에서 측정자의 측정평균값중 60%이상이 관리 한계선 밖에 있으므로 바람직하다고 말할 수 있다.

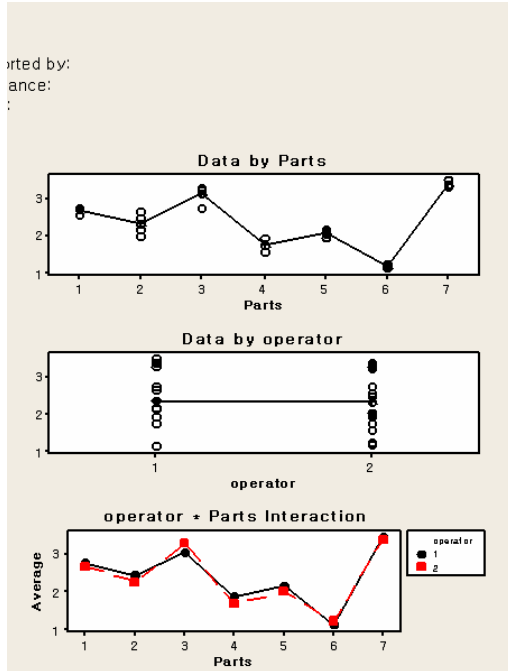


(그림 13. Gage R&R – Diagnostics with Graphs)

계속해서, 그림 14의 Data by Parts는 각 모델별 측정값의 산포가 적고 각 모델별 평균값은 차이가 크다. 따라서 바람직하다고 한다. Data by operator는 모델별로 구별하지 않고, 시험 환경들 평균 연결선이 수평으로 바람직하고, 마지막 그래프결과인 ‘operator*parts Interaction’은 각 모델별로 측정자의 평균값들이 2군에서 교차함을 확인할 수 있어 교호작용이 있다고 판단 할 수 있다.

Gage R&R을 통하여 시스템 설계에 따른 개선가능이 있는 것으로 판단되어 정의된 Project Y에 대한 현재 수준을 정확히 파악하고, 이에

따라 개선 목표와 방향을 설정하여야 한다.



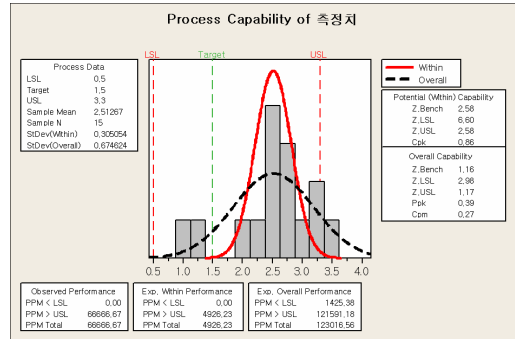
(그림 14. Gate R&R - 결과분석)

그림 15와 16은 단말의 통화 시스템의 현재 수준을 Graph로 나타낸 것이다. Process는 통계적으로 안정되어 있으며, 수집된 데이터는 정규분포를 따른다고 할 수 있다. 현재 시스템의 정확한 수준을 파악하기 위하여 ZShift값을 구한다.

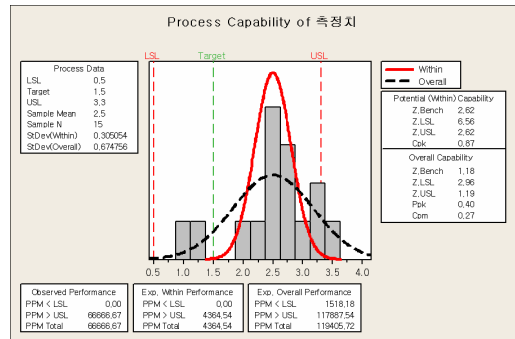
$$ZShift = ZST - ZLT$$

그림 15와 그림 16에서 ZShift값은 다음과 같다.

$$ZShift = 2.62 - 1.16 = 1.46$$

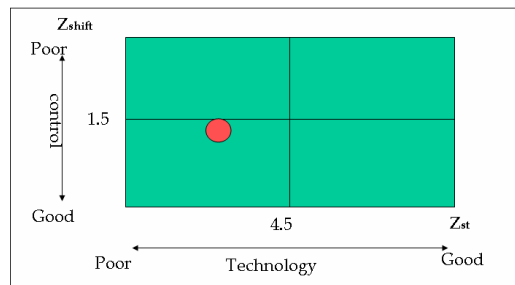


(그림 15. Graph 분석 I - ZLT)



(그림 16. Graph 분석 II - ZST)

따라서 그림 17의 4Block Diagram으로 결과를 분석하면 현재 통화시스템은 ZShift는 1.46으로 관리가 대체로 적절하지 못하고 ZST는 2.62로 통화 SW를 설계하는 기술력 향상이 필요하다.



(그림 17. 4 Block Diagram)

4.2.3 Analysis

Analyze 단계는 Measure 단계에서 도출된 CTQ의 현 수준의 원인을 분석하기 위해, 도

General Linear Model: Response versus Architecture, States, ...						
Factor	Type	Levels	Values			
Architecture	fixed	2	SM/PM/MM, Single Manager			
States	fixed	3	1, 4, 7			
Memory Opti.	fixed	2	On, Off			
Event Driven	fixed	2	On, Off			
Call Protocol	fixed	2	SIP, H.324			

Analysis of Variance for Response, using Adjusted SS for Tests						
Source	DF	Seq SS	Adj SS	Adj MS	F	P
Architecture	1	2.4098	2.4098	2.4098	6.98	0.012
States	2	3.7320	3.7320	1.8660	5.33	0.008
Memory Opti.	1	2.6901	2.6901	2.6901	7.68	0.008
Event Driven	1	0.0566	0.0566	0.0566	0.16	0.690
Call Protocol	1	0.7615	0.7615	0.7615	2.17	0.147
Architecture*States	2	4.6728	4.6728	2.3364	6.67	0.003
Architecture*Memory Opti.	1	0.2893	0.2893	0.2893	0.83	0.368
Architecture*Event Driven	1	0.0010	0.0010	0.0010	0.00	0.959
Architecture*Call Protocol	1	1.1073	1.1073	1.1073	3.16	0.082
States*Memory Opti.	2	2.1586	2.1586	1.0793	3.08	0.055
States*Event Driven	2	1.5196	1.5196	0.7598	2.17	0.125
States*Call Protocol	2	0.1498	0.1498	0.0749	0.21	0.808
Memory Opti.*Event Driven	1	0.3516	0.3516	0.3516	1.00	0.321
Memory Opti.*Call Protocol	1	0.2370	0.2370	0.2370	0.68	0.415
Event Driven*Call Protocol	1	0.1846	0.1846	0.1846	0.53	0.471
Architecture*States*Memory Opti.	2	0.6764	0.6764	0.3382	0.97	0.388
Architecture*States*Event Driven	2	0.2101	0.2101	0.1051	0.30	0.742
Architecture*Memory Opti.*Event Driven	1	0.1197	0.1197	0.1197	0.34	0.562
Architecture*States*Call Protocol	2	0.0474	0.0474	0.0237	0.07	0.935
Architecture*Memory Opti.*Call Protocol	1	0.1269	0.1269	0.1269	0.36	0.550
Architecture*Event Driven*Call Protocol	1	0.1087	0.1087	0.1087	0.31	0.580
States*Memory Opti.*Event Driven	2	0.0144	0.0144	0.0072	0.02	0.980
States*Memory Opti.*Call Protocol	2	0.6805	0.6805	0.3402	0.97	0.386
States*Event Driven*Call Protocol	2	1.3476	1.3476	0.6738	1.92	0.157
Memory Opti.*Event Driven*Call Protocol	1	0.8607	0.8607	0.8607	2.46	0.124
Architecture*States*Memory Opti.*Event Driven	2	0.1902	0.1902	0.0951	0.27	0.763
Architecture*States*Memory Opti.*Call Protocol	2	1.0520	1.0520	0.5260	1.50	0.233
Architecture*States*Event Driven*Call Protocol	2	0.3669	0.3669	0.1834	0.52	0.596
Architecture*Memory Opti.*Event Driven*Call Protocol	1	0.0464	0.0464	0.0464	0.13	0.718
States*Memory Opti.*Event Driven*Call Protocol	2	0.3162	0.3162	0.1581	0.45	0.639
Architecture*States*Memory Opti.*Event Driven*Call Protocol	2	0.3231	0.3231	0.1615	0.46	0.633
Error	48	16.8151	16.8151	0.3503		
Total	95	43.6237				

S = 0.591873 R-Sq = 61.45% R-Sq(adj) = 23.71%

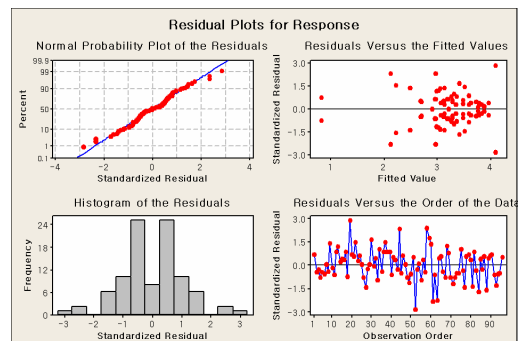
Unusual Observations for Response

Obs	Response	Fit	SE Fit	Residual	St Resid
19	5.30000	4.10000	0.41852	1.20000	2.87 R
44	3.09000	2.11500	0.41852	0.97500	2.33 R
52	2.90000	4.10000	0.41852	-1.20000	-2.87 R
58	3.96000	2.97500	0.41852	0.98500	2.36 R
61	1.99000	2.97500	0.41852	-0.98500	-2.35 R
63	1.14000	2.11500	0.41852	-0.97500	-2.33 R

R denotes an observation with a large standardized residual.

(그림 18. General Full Factorial Design)

출된 가인자를 검증하여 치명인자를 선정하며, 치명인자의 세부 의미를 분석하여 근본원인을 선정한다. 본 논문의 증명을 위하여 가인자는 Architecture, State사용, Memory Optimizing, Event Driven SW구조, 통신 Protocol로 선정한다. Architecture는 본 논문의 제시 구조인 SM/PM/MM인지 Single Manager를 사용하는지로 실험하고, State는 3 level로 1 state, 4 states, 7 states로 한다. Memory Optimizing과 Event Driven SW구조는 On/Off로 하고, 통신 설정 Protocol은 SIP와 H.324를 사용하여 실험 Data를 구한다.



(그림 19. 실험 결과에 대한 Residual Plots)

5개의 가인자 조건을 주어 General Full Factorial Design으로 실험한 후 결과를 분석하면 그림 18과 같다. 유의 수준 5%하에서 2인자 교호작용이 Architecture*State를 제외하고는 유의하다고 분석된다. Architecture*State를 고정인자로 놓고 RSM을 수행하면 그림 20과 같은 결과를 확인 할 수 있다. 그림 20에서는 Architecture는 SM/PM/MM 방식을 사용하고, State는 7 States를 두어 시스템을 설계하는 것

이 가장 좋은 결과를 가지는 것을 증명하고 있다.

Least Squares Means for Response			
Architecture*States		Mean	SE Mean
SM/PM/MM	1	3.228	0.1480
SM/PM/MM	4	3.431	0.1480
SM/PM/MM	7	2.465	0.1480
Single Manager	1	3.400	0.1480
Single Manager	4	3.295	0.1480
Single Manager	7	3.380	0.1480

Residual Plots for Response			
-----------------------------	--	--	--

(그림 20. 실험 결과 분석)

V. 결론

최근에는 핸드폰과 같은 소형 단말에서뿐만 아니라 Digital TV나 정보가전에서 SIP를 이용한 다양한 응용시나리오를 적용하고 있다. 본 논문은 단말에서 응용서비스를 효율적으로 개발하기 위하여 Seven States를 설정하고 발생 할 수 있는 이벤트를 정의하여 SIP프로토콜로 Call-Setup을 하는 모든 응용시나리오에 적용할 수 있다. 응용서비스에 따라 다양한 예외상황이 발생하고 단말의 치명적인 오류에 대해 안정적으로 처리 할 수 있는 메커니즘을 보인다. PoC라는 응용서비스를 구현하여 OMA에서 제시하는 EICS의 모든 Item들을 pass하였다. OMA에서 제시하는 Test case가 PoC에 의존적이라는 한계는 있겠지만 Seven States구조

로 설계된 SIP 인터넷 전화연결 시스템은 효율적이고 확장 가능한 signal처리를 할 수 있다. 표 2의 실험 결과와 같이 SIP 전화연결 시 다양한 예외상황을 볼 수 있는데 각각의 경우에서도 상대방과의 세션을 유지하면서 안정적인 미디어 전송이 가능하였다. 새로운 응용서비스가 추가되면 시나리오에 따른 이벤트를 추가로 정의하여 FSM에 적용하여 개발자가 쉽게 확장 가능한 구조이다. 마지막으로, 본 논문에서 제시하는 7 State구조가 통신 시스템 설계에 효율적임을 6 Sigma검증 방법으로 보였다.

향후 연구과제로는 새로운 응용서비스에 따른 추가 이벤트를 정형화 할 수 있는 구조설계를 하여 이벤트 추가 없이 User Interface API만 변경하는 구조를 설계할 것이다.

>> 참고문헌

- [1] J. Rosenberg, H.Schulzrinne, G. Camarillo, A. Johnston, J. Peterson, R. Sparks, M/Handley, E/ Schooler, "Session Initiation Protocol," IETF, RFC3261, June 2002.
- [2] B. Campbell, R. Mahy and C. Jennings, "MSRP: The Message Session Relay Protocol," IETF DRAFT: draft-ietf-simple-message-sessions-10, Feb. 2005.
- [3] M. Handley and V. Jacobson, "SDP: Session Description Protocol," IETF, RFC 2327, April 1998.
- [4] A. Johnston and O. Levin, "Session initiation

- protocol call control Conferencing for User agents,” IETF DRAFT: draft-ietf-sipping-cc-conferencing-06, 2004.
- [5] J. Rosenberg, “The Session Initiation Protocol (SIP) UPDATE Method,” IETF, RFC 3311, Sep. 2002.
- [6] W. Marshall, “Private Session Initiation Protocol (SIP) Extensions for Media Authorization,” IETF, RFC 3313, Jan. 2003.
- [7] Arkko, J. Torvinen, V. Camarillo, G., Niemi A. and Haukka T., “Security Mechanism Agreement for the Session Initiation Protocol,” IETF, RFC 3329, 2003.
- [8] T. Evers and H. Schulzrinne, “Predicting internet telephony call setup delay,” in Proc. IP Telephony Workshop, Berlin, Germany, Apr. 2000.
- [9] C. Huang, C. Lee and C. Zheng, “A Novel SIP-Based Route Optimization for Network Mobility,” IEEE Commun., vol. 24, no. 9, pp. 1682-1691, Sept. 2006.
- [10] R. V. Prasad, R. Hurni, and H. S. Jamadagni, “A scalable distributed VoIP conferencing using SIP,” in Proc. 8th IEEE Int. Symp. Comput. Commun., Jun./Jul. 2003, pp. 608-613.
- [11] Salsano, S. Veltri, L. and Papalillo D., “SIP Security Issues: The SIP authentication procedure and its processing load,” IEEE Network, Vol. 16, no 6, pp. 38-44, Nov.-Dec. 2002.
- [12] 허미영, 한재천, 현욱, 박선옥, 강신각, “SIP 기반 인터넷전화 서비스를 위한 사용자 에이전트의 설계 및 구현”, 한국정보과학회논문지 정보통신 제32권 제 3호, 2005년 6월
- [13] Telecommunication Standardization Sector of ITU, “ITU-T Recommendation H.323 Packet based multimedia communications systems,” Orlando, FL, Feb. 1998.
- [14] vovida software, <http://www.vovida.org/protocols/downloads/sip/>
- [15] Open Mobile Alliance, <http://www.openmobilealliance.org/>.

>> 저자소개



신 용 경(Yongkyoung Shin)

· Email : lemonsage@lge.com
· Tel : +82-2-2033-7843
· Fax : +82-2-2005-3850

- 1999. 2 : 경북대학교 컴퓨터과학 학사
- 2001. 2 : 경북대학교 컴퓨터과학 석사
- 2003. 2 : 경북대학교 컴퓨터과학 박사수료
- 2003. 2~현재 : LG전자 소프트웨어센터 선임연구원
- 주관심분야 : 이동 멀티미디어 스트리밍, 네트워크 프로토콜, MPEG-4 콘텐츠, 실시간 스트리밍