

RTP 전송 모듈을 포함한 Java 기반의 SIP User Agent의 구현

조현규⁰ 김영학 장춘서
금오공과대학교 컴퓨터공학과
(blackjo⁰ yhkim)@cesp.c1.kumoh.ac.kr, csjang@kumoh.ac.kr

Implementation of Java based SIP User Agent Including RTP transmission module

Hyun Gyu Jo⁰ Young Hak Kim Choon Seo Jang
Dept. of Computer Engineering, Kumoh National Institute of Technology

요 약

VoIP(Voice over IP) 시스템을 구현함에 있어서 호설정을 처리하는 여러 프로토콜이 제안되고 있는 가운데 IETF(Internet Engineering Task Force)에서 제안한 SIP(Session Initiation Protocol)는 텍스트 기반의 프로토콜로서 구현과 파싱이 쉬운 등 많은 장점을 가지고 있어 차세대 VoIP의 표준으로 자리잡고 있다. 또한, 뛰어난 확장성을 가지고 있어 다양한 서비스에 적용할 수 있는 호설정 프로토콜이다. 본 논문에서는 SIP를 이용한 VoIP 시스템을 구현함에 있어 주요 구성요소 중 하나인 UA(User Agent)를 2002년 6월에 발표된 새로운 SIP 버전에 맞추어 개발하였다. 본 UA는 플랫폼에 독립적으로 기능을 할 수 있도록 자바(Java)를 사용하여 GUI(Graphical User Interface)환경으로 구현하였다. 그리고 RTP(Real-time Transport Protocol) 전송 모듈을 통하여 호설정이 이루어진 후 실제 음성과 화상통신이 이루어지는 부분을 포함하였다.

1. 서 론

VoIP 시스템을 구현함에 있어 호설정을 위해 H.323, SIP, MGCP, Megaco/H.248, RTSP 등 여러 프로토콜이 제시되고 있으며 그 중에서 IETF에서 제안한 SIP는 HTTP(Hyper Text Transfer Protocol)처럼 텍스트 기반의 프로토콜로서 구현이 쉽고 응용성, 확장성 등의 측면에서 많은 장점을 가지고 있어 차세대 VoIP의 표준으로 자리잡고 있다. 또한, SIP는 이동성 지원 서비스, 인스턴스 메시징 서비스, 전자상거래 등 다른 응용분야로의 확장이 용이한 프로토콜이다.

본 논문에서는 SIP를 이용한 VoIP 시스템을 구현함에 있어서 구성요소중의 하나인 UA를 자바를 이용하여 2002년 6월에 발표된 새로운 SIP 버전에 맞추어 개발하였다. 또한, RTP 전송모듈을 함께 구현하여 호설정이 이루어진 후에 실제 통신이 가능하도록 하였다. 본 논문의 구성은 다음과 같다. 2장은 SIP의 구성과 SDP(Session Description Protocol), RTP에 대해 언급하고 3장은 구현된 시스템에 대한 설명과 이를 사용하여 호설정을 하는 과정 및 음성과 화상을 통해 실제 통신이 이루어지는 부분을 다루며 4장에서 결론을 맺는다.

2. SIP 구성 및 SDP와 RTP

2.1 SIP 구성

SIP는 VoIP 시스템에서 사용자간에 호설정과 제어를 담당하는 프로토콜의 하나로 본 논문은 2002년 6월에 RFC 3261 표준 문서로 확정된 새로운 버전을 기반으로 하였다.

SIP는 하위계층의 트랜스포트 프로토콜과 독립적으로 동작하는 응용계층의 프로토콜로서 호처리는 UTF-8 포맷의 텍스트 형태인 메시지를 사용한다. 메시지는 클라이언트에서 서버로 보내지는 요청(request)메시지와 서버에서 클라이언트로 보내지는 응답(response) 메시지로 나뉘어진다.

요청메시지의 종류로는 INVITE, ACK, CANCEL, BYE, REGISTER, OPTION 등의 메소드가 있고 응답메시지는 숫자로 된 상태코드를 사용하며 종류는 호의 진행상태를 나타내는 1xx, 요청이 성공적으로 처리 되었음을 나타내는 2xx, 요청에 대한 위치정보에 관해 응답하는 3xx, 잘못된 형식의 요청 등에 응답하는 클라이언트 에러인 4xx, 서버 에러를 나타내는 5xx, 요청에 대한 지원이 불가능한 경우에 응답하는 6xx계열이 있다[1][2].

메시지는 시작라인(start-line)과 헤더필드, 헤더필드의 마지막을 가리키는 공백라인 그리고 옵션부분인 바디(Body)부분으로 구성된다. 여기서, 시작라인의 형태에 따라 요청과 응답메시지가 구분이 되고 헤더필드를 포함한 나머지 부분은 두 메시지 모두 동일한 형태를 갖는다. 헤더필드에 필수적으로 포함되는 항목은 요청메시지 전송 후 되돌아올 응답을 위한 위치 정보인 Via 필드, caller와 callee의 주소정보를 나타내는 From과 To 필드, 하나의 다이얼로그(dialog) 식별을 위한 Call-ID, 트랜잭션 식별자로 사용되는 CSeq, 그리고 새로운 버전에 필수적으로 포함된 메시지가 목적지까지 전달되는 동안의 홉(hop)의 수를 제한하는 기능을 하는 Max-Forwards 이다. 이들 필드 중에서 Via 필드에는 요청에 의해 생성된 트랜잭션을 확인하기 위한 branch 파라미터가 필수적

로 포함되고 From과 To 필드에는 Tag 파라미터가 포함되는데 용도는 Call-ID와의 조합으로 다이얼로그 식별에 사용된다. 바디는 실제통신을 위한 미디어의 협약에 필요한 세션정보를 기술하는 SDP 형식으로 표현된다.

SIP의 구성요소에서 서버의 종류로는 받은 메시지를 파싱하고 처리하며 적절한 응답메시지를 보내거나 또는 다른 서버나 UA로 메시지를 포워딩하는 프록시(proxy) 서버, 받은 메시지에 대해 해당하는 주소를 전달하는 리다이렉트(redirect)서버, 사용자의 위치정보를 포함하는 로케이션(location)서버, 구성요소들을 등록, 구성, 삭제하는 역할을 담당하는 레지스트라(registrar)가 있다[3]. UA는 호설정을 위한 메시지를 송/수신하여 SIP 트랜잭션을 개시 및 종료하는 논리적 실체로서 터미널, 인터넷 폰 등을 예로 들 수 있다. UA는 UAC(User Agent Client)와 UAS(User Agent Server)로 구분되고[2] 보통은 하나로 통합되어 구현되어진다.

UA 간에 주고받는 메시지는 서버를 통하거나 또는 직접적으로 송/수신이 되며 송신측에서 보내는 요청메시지를 수신측에서 수신하면 적절한 응답메시지를 보냄으로써 호처리가 진행된다. 그림 1은 두 UA간에 프록시 서버를 통하여 호처리가 이루어지는 예이다.

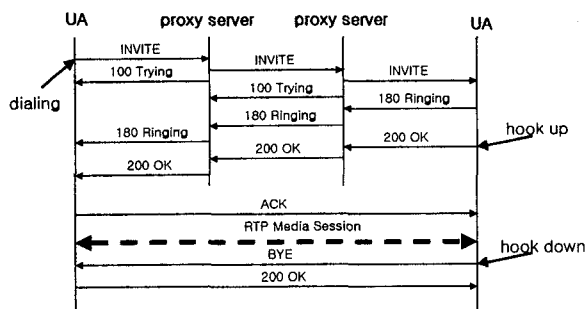


그림 1: SIP 호처리의 흐름도

2.2 SDP와 RTP

SDP는 멀티미디어 세션을 기술하는 프로토콜로서 SIP에서는 메시지의 바디부분에 내장되어 RTP상에서 통신에 사용되는 미디어의 종류와 포트번호 코덱(codec)능력 등을 협상하는 역할을 한다. SDP에 기술되는 내용은 크게 세가지 범주로 나뉘어진다. 첫째, 프로토콜 버전과 세션이름 등 전반적인 세션정보를 나타내는 세션 기술(Session Description) 둘째, 통신의 시작, 중단, 반복 등의 시간정보를 담고있는 시간 기술(Time Descriptions) 셋째, 미디어의 타입, 트랜스포트 프로토콜, 미디어의 형식 등의 사항을 담고 있는 미디어 기술(Media Descriptions)부분이다[4].

SDP의 기술형식은 <type>=<value>의 텍스트 형태로 구성되고 그림 2는 SIP 메시지의 바디에 기술되는 SDP의 예이다.

```

v=0
o=johg IN IP4 sip2.kumoh.ac.kr
c=IN IP4 202.31.130.29
t=0 0
m=audio 42050 RTP/AVP 0
    
```

그림 2: SDP의 예

RFC 1889로 표준화된 RTP는 오디오, 비디오 및 시뮬레이션 데이터와 같은 실시간 데이터를 멀티캐스트 또는 유니캐스트 네트워크를 이용해서 전송하는 응용 서비스에 적합한 종단간 트랜스포트 기능을 제공하는 프로토콜이다. RTP는 데이터를 페이로드(payload) 타입을 이용하여 전송할 수 있을 뿐만 아니라 모니터링(monitoring), 타임스탬프(timestamp), 시퀀스번호(sequence number)를 이용하여 데이터의 실시간 전송이 가능한 프로토콜이다[5]. RTP 데이터 전송기능은 제어 프로토콜에 의해 확장되는데, RTCP(Real-time Transport Control Protocol)는 데이터의 전달상황을 감시하며, 최소한의 제어기능과 매체 식별 기능을 제공한다.

3. 시스템구현

본 연구에서는 호처리를 위한 SIP의 구현을 위하여 RFC 3261 표준문서를 따랐다. 구현을 위한 시스템의 구성은 그림 3과 같으며 UA 사이의 SIP 메시지 전달을 위해 프록시 서버를 채택하여 linux OS 상에서 운영하였고 등록이나 위치 정보 서비스는 MySQL을 사용하여 관리하였다. UA는 자바를 이용하여 PC 윈도우즈 상에서 GUI 형태로 작성하였다. 더불어 RTP 전송 모듈은 자바 기반 멀티미디어 프로그래밍을 위한 API인 JMF(Java Media Framework)를 사용하여 구현하였다.

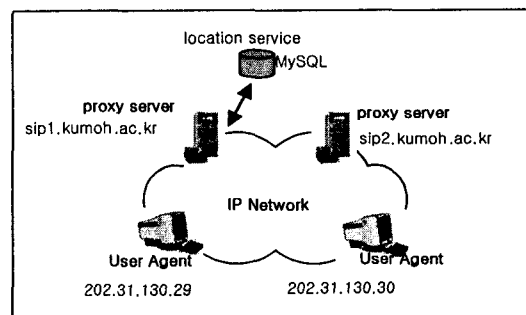


그림 3: 시스템 구성도

UA의 메인화면은 그림 4와 같으며 주요 기능들은 다음과 같다. 호설정을 위한 요청메시지 생성, 생성된 요청메시지 전송, 수신 받은 메시지 파싱, 요청메시지의 형태 및 상태에 따라 대응하는 응답메시지 생성, 생성된 응답메시지를 전송하는 부분 등이며 이러한 기능을 사용하기 위한 GUI 화면과, 호처리시에 주고받는 메시지의 내용을 보여주는 부분 등을 포함시켰다.

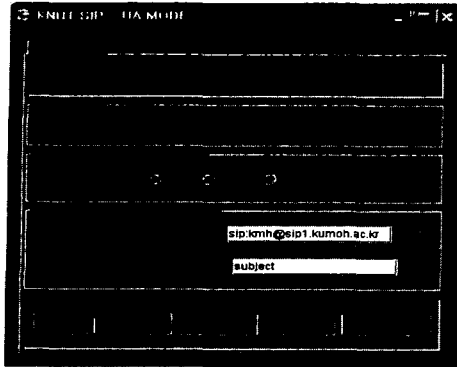


그림 4: 구현된 UA의 메인화면

구현된 UA를 사용하여 호설정을 하는부분 및 실제 통신과 세션이 종료되기까지의 과정은 다음과 같다. 하나의 세션을 생성하기 위해서 caller가 'Dial' 버튼을 눌러 INVITE 요청메시지를 전송하면, 서버를 통하여 요청메시지를 수신한 callee는 'Answer' 버튼을 눌러 caller에게 성공적으로 메시지를 받았음을 알리는 200 OK 응답메시지를 보낸다. 이 메시지를 받은 caller의 UAS는 ACK 메시지를 발송하여 응답메시지를 바르게 받았음을 callee에게 알리고 호가 개설된다. 그 다음 호설정시에 사용된 SDP에서 협의한 미디어를 통하여 실제적인 통신이 RTP 상에서 이루어진다. RTP 세션은 쓰레드(thread)를 이용하여 처리하여 통신 중에 있는 UA는 또다른 UA로부터 오는 요청메시지를 수신하고 처리할 수 있도록 하였다. 통신이 끝나고 하나의 세션이 종료되는 과정은 'Hangup' 버튼을 눌러 BYE 요청메시지를 보내고 200 OK 응답메시지를 받음으로써 처리가 된다. 이때 BYE 메시지는 구 SIP 표준문서인 RFC 2543 에서는 언제라도 전송될 수 있었으나 새로운 SIP 표준문서에서는 존재하는 다이얼로그 내에서만 보내질 수 있도록 변경되었으므로 본 논문에서도 이에 따라 구현하였다. caller가 INVITE 요청메시지를 보낸 후 'Cancel' 버튼을 누르는 경우에는 CANCEL 요청메시지가 전송이 되고 이에 따라 호설정이 취소된다. 그림 5는 UA가 주고 받는 메시지의 실제 내용이다.

실제 통신을 지원하는 RTP 모듈은 JMF를 사용하여 프로그래밍 하였고 호설정을 처리하는 과정에서 결정된 미디어의 종류, IP주소, 포트번호 그리고 압축방식 등의 정보를 이용하여 처리된다. 구현된 UA에서는 텍스트, 음성, 화상의 세가지 형태의 실제 통신을 제공하였으며 음성통신은 별도의 화면 없이 송/수신자간에 마이크와 스피커를 통하여 전송이 되도록 구현하였고 화상통신에는 음성의 송/수신 기능을 포함시켰다. 이러한 통신을 위해 RTP 모듈들은 음성과 영상을 캡처하는 부분, 캡처된 데이터의 소스를 생성하는 부분, 음성과 영상을 전송하기 위한 프로세서의 생성과 포맷지정 부분, RTP 전송처리를 하는 부분, 전송을 종료하는 부분, 데이터의 수신에 따라 재생하는 부분으로 구현되었다. 그림 6은 구현된 시스템에서 화상통신을 하는 실제 화면이다.

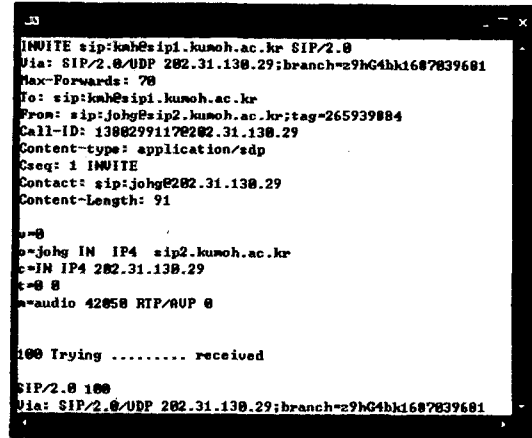


그림 5: 호설정을 위한 송/수신 메시지



그림 6: 구현된 시스템에서의 실제 화상통신 화면

4. 결론

본 논문에서는 SIP를 이용한 VoIP 시스템의 주요 구성요소인 UA 부분을 플랫폼에 독립적으로 기능을 할 수 있도록 자바를 사용하여 GUI형태로 구현하였고 RTP 전송 모듈을 함께 구현하여 호설정이 처리된 후 실제 음성과 화상통신이 이루어질 수 있도록 하였다. 이때, SIP 메시지의 바디부분에는 SDP를 사용하여 RTP상에서 통신에 사용되는 미디어의 종류와 전송에 사용되는 포트번호, 코덱능력 등을 다양하게 협상할 수 있도록 하였다. 이와 같이 SIP는 기존의 VoIP 서비스뿐만 아니라 다양한 서비스의 호설정 프로토콜로서 이용될 수 있으며 여러 응용분야에 적용할 수 있는 프로토콜이므로 본 논문에서 구현된 시스템을 기반으로 SIP 확장(Extention) 기능을 추가하여 다양한 응용부분에 적용하는 연구가 추가로 필요하다고 보겠다.

5. 참고문헌

- [1] IETF SIP Working Group, <http://www.ietf.org/html.charters/sip-charter.html>
- [2] Session Initiation Protocol, RFC 3261
- [3] Fredrik Fingal, Patrik Gustavsson. "A SIP of IP-telephony", <http://www.cs.columbia.edu/~hgs/sip/papers.html>
- [4] Session Description Protocol, RFC 2327
- [5] A Transport Protocol for Real-Time Applications, RFC 1889