

# Parlay Web Service를 이용한 VoIP서비스의 구현

\*정의현 \*김화성 \*\*최영일 \*\*이병선

\*광운대학교 전자통신공학과, \*\*한국전자통신연구소, 소프트웨어팀

\*bbohal@hanmail.net, \*hwkim@daisy.gwu.ac.kr, \*\*{yichoi, bslee}@etri.re.kr

## Implementation of VoIP using Parlay Web Service

\*Yeuhun Jung \*Hwasung Kim \*\*Youngil Choi \*\*Byungsun Lee

\*Dept. Electronic and Communications Engineering, Kwangju Univ. \*\*Softswitch team, ETRI .

### 요 약

개방형 네트워크로의 진화는 third-party업체들의 참여를 가능하게 하였고, third-party애플리케이션의 대표적인 예로 VoIP를 들 수 있다. 최근 플랫폼에 중립적이며, 언제 어디서나 손쉽게 접근할 수 있는 Web 서비스의 등장은 많은 개발자들의 흥미를 유발하고 있다. 이에 Parlay Group에서는 개방형 네트워크에서 Web서비스를 지원하기 위해 Parlay X API를 제정하였다. 본고에서는 개방형 네트워크에서의 Web서비스 제공을 위한 방법으로 VoIP Web서비스를 구현하여 Parlay X API를 활용하여 보았다.

### 1. 서 론

통신환경의 변화는 개방형 네트워크로 새로운 통신네트워크의 변화를 가져오게 되면서 third-party업체들이 애플리케이션을 제공할 수 있게 되었다. 이러한 애플리케이션 중 가장 대표적인 예로 VoIP(Voice over Internet protocol) 서비스를 들 수 있다. 즉 개방형 구조의 차세대 네트워크에서 Voice 및 멀티미디어 서비스 제공에 대한 연구 개발이 이루어지고 있다.

이러한 서비스를 제공하기 위한 개방형 인터페이스의 대표적인 예로서 Parlay API(Application Programming Interface)가 있다. Parlay group은 AT&T, BT, Cegetel, Cisco Systems, Ericsson, IBM, Lucent Technologies, Microsoft, Nortel Networks, Siemens and Uticom와 같은 여러 통신장비 회사와 컴퓨터 플랫폼 회사들을 포함하고 있다. 그리고 새로운 방식의 Open Programmable Network API규격을 제정하고 동시에 이 규격이 상용제품의 구현에 채택될 수 있도록 촉진하는 것을 목표로 하고 있다. 또한 Parlay API는 음성과 영상 그리고 데이터를 네트워크에서 제어하는 통합된 API를 추구한다.

최근 Parlay 그룹은 Java API Realization Working Group과 Web Service Working Group을 만들면서 Parlay 웹 서비스 API에 대한 관심이 높아지고 있다. Web서비스는 최근 가장 각광 받는 기술이며 향후 많은 서비스들이 웹 서비스로 변할 것이라고 판단한다. Web서비스는 기업에서 애플리케이션의 통합과 동일한 개념이며 애플리케이션에 구매 받지 않고 비즈니스로직의 변화에 맞게, 필요 시 조합하여 유연한 시스템을 구현할 수 있는 장점을 가지는 기술이다. 즉 이는 웹 서비스를 통하여 기존의 분산 컴퓨팅을 구현할 뿐만 아니라 기존의 분산 컴퓨팅이 해결할 수 없는 범위 밖의 것들을 웹 서비스를 통하여 할 수 있게 됨에 따라 기존 CORBA, RMI, COM을 사용하는 Parlay API도 웹 서비스를 통하여 구현이 가능하게 되었다. 이러한 흐름에 Parlay 그룹도 따라감에 따라 새로 제공되는 서비스 애플리케이션들이나 기존의 애플리케이션들이 사용자들이 사용하기 편리한 Web서비스로 개발되거나 변화할 것이라고

기대 된다. 마찬가지로 대표적인 서비스인 VoIP서비스도 Web 서비스를 이용한 서비스로 발전할 것이라 본다[1].

### 2. VoIP

VoIP란 지금까지 PSTN(Public Switched telephone network)을 통해 이루어졌던 음성 서비스를 Internet Protocol을 이용해 여러 가지 다양한 서비스를 제공하는 기술을 말한다. 이렇게 IP망을 이용함으로써 기존의 전화망에서 하지 못했던 많은 서비스들이 이루어지고 있으며, 대표적인 응용들로서는 웹콜센터, Instance message, CTI(computer Telephony integration), UMS(Unified Messaging System)등을 들 수 있으며, 현재도 다양한 응용들을 서비스 하려고 하고 있다.

현재 VoIP 구조는 gateway와 단말 그리고 call connection agent로 구성되어 있다. gateway는 PSTN망과 IP망을 연결해서 통신이 가능하게 해주며, 이런 gateway를 call connection agent가 MGCP(Media Gateway Control Protocol)란 프로토콜을 사용해서 제어하게 된다. 앞으로 call connection agent는 SoftSwitch로 대체 될 것이며, 현재 활발한 연구가 진행 중에 있다. 그리고 IP망에서는 기본적으로 H.323과 SIP(Session Initiation Protocol)가 시그널링 프로토콜로 사용되고 있다.

### 3. Web 서비스

Web 서비스는 애플리케이션 개발에서는 최신 기술이며, 다양한 플랫폼에서 개발하는 개발자들의 흥미를 유발하고 있다. 기본적인 개념은 간단한데, Web 서비스라는 기술을 통해 인터넷이나 네트워크로 다른 객체에 RPC(Remote Procedure Calls)를 수행한다는 것이다. 웹 서비스가 이런 기능을 지원하는 최초의 기술은 아니지만, 플랫폼 중립적 표준인 HTTP나 XML을 사용함으로써 클라이언트에게 전체 시스템 구현을 송길 수 있다는 점에서 기존 기술과 차별화된다. 클라이언트는 서비스의 URL과 메소드 호출에 사용될 데이터형은 알아야 하지만 어떤 프로그래밍 언어를 사용해서 구현되어 있는지는 알 수 없다 [4].

본 연구는 ETRI 위탁과제로 수행되었음.

### 3.1. Web 서비스의 모델과 Parlay Web서비스

Web 서비스에는 크게 세가지 역할이 존재한다. 그것은 서비스 제공자, 서비스 요청자, 서비스 중개자이다. Parlay Web 서비스 역시 일반적인 Web 서비스 모델과 그 모양이 비슷하다 [4].

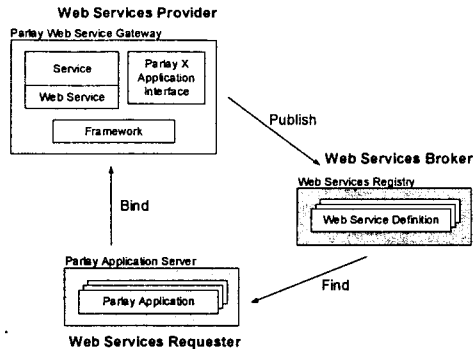


그림 1. Web 서비스 모델

- 서비스 제공자(Service provider): 서비스 제공자는 Web 서비스를 제작하는 이를 말한다. 서비스 제공자는 Web서비스 형태로 사용자에게 비즈니스 기능을 제공한다. Parlay Web 서비스 제공자는 Parlay Web 서비스에 필요한 Interface를 Web서비스 애플리케이션의 서버 역할을 하는 Gateway를 구현하고, 기능을 기술한 WSDL(Web Service Description Language)문서를 서비스 중개자에게 등록하여 공개하게 된다.
- 서비스 요청자(Service Requester): 서비스 요청자는 Web 서비스를 실제로 사용하는 애플리케이션을 의미한다. Web 서비스 요청자인 Parlay 애플리케이션은 서비스 제공자가 WSDL문서를 통해 서비스의 기능을 알아낼 수 있다. 먼저 서비스가 공개된 레지스트리를 검색 하여 서비스의 세부사항을 알아낸다. 이후에 원하는 서비스에 연결하게 되면 Gateway의 Framework를 통해 Interface에 접근, 사용이 가능하게 된다.
- 서비스 중개자(Service Broker): 서비스 중개자는 서비스 레지스트리에 서비스 제공자가 공개한 WSDL문서를 올려두고 서비스 요청자가 원하는 서비스를 발견할 수 있도록 도와주는 역할을 한다.

### 3.2 WSDL

WSDL은 XML포맷을 이용하여 Web서비스를 기술하는 데이터 형식을 말한다. 기본적으로 Web서비스의 메소드, 메소드의 인자형, 데이터형, 하부의 전송 프로토콜, Web서비스의 종점 URL을 명시하게 된다.

Web서비스 요청자는 Web서비스가 어떤 동작을 하는지 알아야 한다. 그러므로 어떤 서비스가 제공되는지 Web서비스 제공자에게 그 명세를 요청하게 된다. 이때 서비스 제공자는 다른 과정을 거치지 않고 단지 WSDL파일만을 공개함으로써 요청자에게 필요한 모든 정보를 알려줄 수 있다.

### 3.3 Parlay X API

Parlay X는 현재의 Parlay API의 제공 능력을 불복화 하여 최상위로 추상한 API형태를 제공하며, 추상화된 메소드들을 통하

여 애플리케이션을 보다 손쉽게 개발할 수 있도록 해준다. 따라서 전문가가 아니더라도 Parlay X API는 추상화 단계가 매우 높기 때문에 그만큼 손쉽게 이용 가능 하므로 많은 개발자들이 참여할 수 있다. 이러한 이유로 인해서 지금보다 많은 Third Party 업체들의 참여를 촉진 시킬 것이다.

Parlay X Web 서비스와 Parlay X API, Parlay API와는 다음과 같은 관계를 가지고 있다. Parlay X 애플리케이션은 Parlay X API Interface를 통해서 Parlay X Web서비스를 이용할 수 있다. Parlay X Web서비스는 기존의 Parlay API Interface를 이용하여 Parlay Gateway를 거쳐 네트워크에 접속하는 것도 가능하도록 되어 있고, 아니면 직접 네트워크에 직접 접속할 수 있게 되어 있다[2][3].

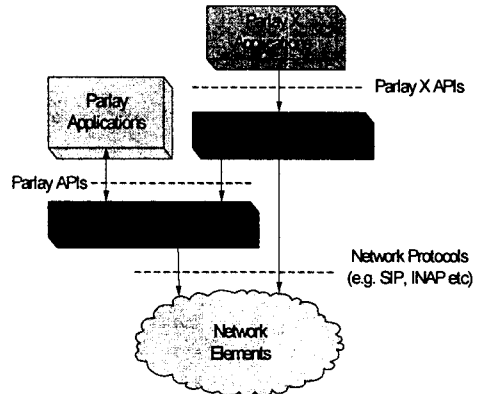


그림 2. Parlay X APIs and Parlay APIs

## 4. 구현

구현은 Parlay X API에서 제공하는 Third Party Call을 이용하였다. 우선 Parlay X API에서 제공하는 WSDL을 이용해 서버와 클라이언트의 java파일을 생성하고, 그 뒤에 필요한 코드를 추가하여 구현하였다. java파일의 생성을 위해서 apache에서 제공하는 Axis의 WSDL2Java를 이용하였다. WSDL2Java는 클라이언트 파일을 생성하고, 서버를 위해 Skeleton, Impl, Stub 파일을 생성한다[5].

### 4.1. 서버

WSDL에서 자동으로 서버파일을 생성하면 Impl로 끝나는 파일을 코딩하게 된다. Parlay X API에서 제공하는 third party call에는 makeACall, getCallInformation, endCall, cancelCallRequest의 메소드가 정의되어 있다. 자동 생성된 파일에는 이부분에 대한 구현이 되어있지 않기 때문에 개발자가 코드를 추가하여 동작하게 한다. 실제 call을 연결하기 위해서 SIP packet을 관리하는 ThirdPartyCallSIP, SIPPacketManager, SIPPacket, SIPParser등의 java파일을 추가하여 SIP에서 정의하는 기능을 구현하였다[6].

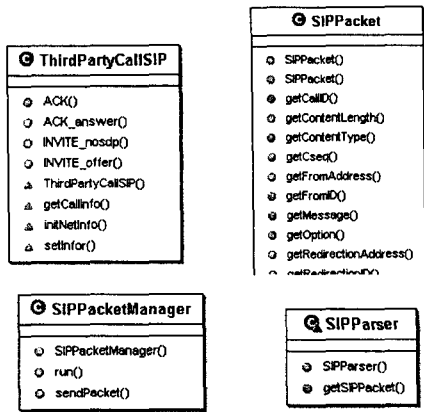


그림 3. Class Diagram

4.2. 클라이언트

클라이언트는 서버로부터 WSDL파일을 받아와 java파일을 생성하고, 그 파일에 정의되어 있는 서버의 4가지 메소드를 이용하여 구현한다. 클라이언트에서는 그 4가지 메소드의 내부구현은 알 수 없고, 넘겨줘야하는 인자값과 리턴값을 가지고 구현하게 된다. Web페이지를 구성하기 위해 서블릿을 사용하였고, 자동 생성된 파일에 다음의 4가지, ThirdPartyWeb.html, ThirdPartyWebInfo.Java, ThirdPartyWeb.Java, make.java를 추가하였다. make파일에는 클라이언트 메소드를 구성하였고, Web 페이지에서 호출하도록 하였다[5].

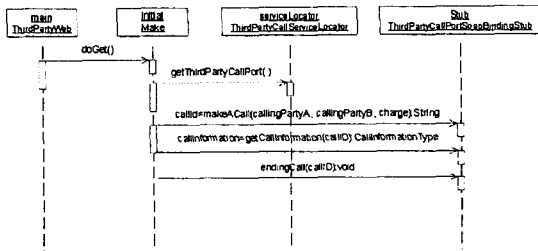


그림 4. 클라이언트

4.3 동작과정

클라이언트는 다음과 같이 동작한다.

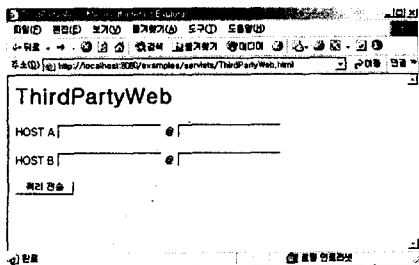


그림 5. 처음 접속하는 Web페이지

사용자가 Web페이지에 접속하여 값을 입력하고 전송하면 makeAcall을 통해 그 값을 서버에게 전달하게 된다. 서버에서는 값을 전달 받아 실제 call을 연결하고 그 뒤에 call ID를 돌려주게 된다. 다시 클라이언트에서는 call ID를 이용하여 call의 정보를 받아오거나 call을 마치고나 할 수 있다.

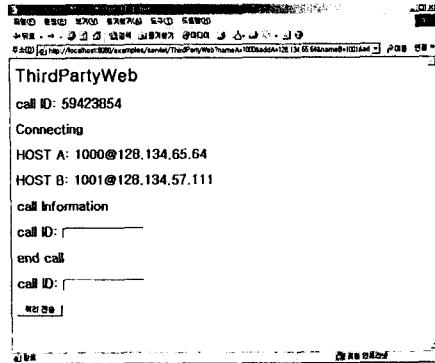


그림 6. call설정이 맺어진 화면

4. 결론

본 논문에서는 Parlay X API를 이용하여 Web서비스를 구현하였다. Parlay X API는 간단하게 추상화 되어있는 몇 개의 메소드만을 제공함으로써 프로그램을 단순화를 도모하였다. 또한 WSDL문서를 통해 서비스를 정의함으로써 어떤 플랫폼에서라도 Parlay X를 통해 손쉽게 Web서비스를 구현할 수 있도록 하였다. Parlay X API에서 메소드의 단순화는 많은 third-party 업체들의 참여를 촉진하여 더 많은 애플리케이션의 개발로 이어지도록 할 것이다. 또한 Parlay X가 Web서비스를 지원함으로써 Web서비스의 이점을 통해 사용자의 편의를 도모하고, 네트워크 사업자 역시 많은 애플리케이션의 적용을 통하여 보다 넓은 시장을 확보하게 될 것이다.

5. 참고 문헌

- [1] Parlay X Web Services Specification V1.0. - <http://www.parlay.org/>
- [2] Parlay X Web Services White Paper. - <http://www.parlay.org/>
- [3] Parlay Web Services - White Papers, Version 1.0. - <http://www.parlay.org/>
- [4] S.Jeelani Basha and Scott Cable Professional Java Web Service.
- [5] S.Jeelani Basha and Romin Irani AIXS: The Next Generation of JAVA SAOP.
- [6] J.Rosenberg, H.Schulzrinne and G.Camarillo "SIP:Session Initiation Protocol, IETF RFC 3261", March 2002.