

주 제

웹서비스 기반의 개방형 서비스 기술

한국전자통신연구원 박유미, 최영일, 이병선

차례

- I. 서 론
- II. 개방형 서비스 구조 개요
- III. Open API 표준화 동향
- IV. 웹서비스 기술의 통신망 적용
- V. 웹서비스 기반의 개방형 서비스 제어 구조
- VI. 국내외 기술 동향
- VII. 국제 상호운용성 시험
- VIII. 결 론

I. 서 론

최근의 정보통신 산업은 글로벌화의 진전으로 말미암아 최고 기술을 선점한 국가만이 생존하는, '국가간 무한경쟁'으로 치닫고 있다. 이에 대해 우리나라에서는 차세대 IT전략인 '유비쿼터스 코리아 (u Korea)'라는 패러다임을 화두로 하여 새로운 산업에 대한 성장 모멘텀의 형성을 추구하고 있으며, 이를 국가 발전의 원동력으로 승화하기 위해 'IT 839'라는 전략이 추진되고 있다.

IT839는 IT산업의 가치사슬에 따라 8대 신규 정보통신 서비스(와이브로, DMB, 텔레매틱스 등)를 활성화하여, 3대 인프라(BcN, IPv6, USN/RFID 등)에 대해 투자를 유발하고, 이를 바탕으로 9개 신성장 동력 산업(차세대이동통신, 디지털TV/방송 등)을

동반 육성하기 위한 IT산업의 발전 전략이다. 네트워크 된 칩이 살아 숨쉬는 유비쿼터스 환경을 만들어내는 실질적인 주체로서 IT839의 3대 인프라에 대한 구축이 추진되고 있으며, 이들 중 광대역 통합망(BcN)은 유비쿼터스 코리아의 기간망으로 주목 받으며 차세대 통신 인프라로서 상용망 구축이 진행되고 있다(그림 1).

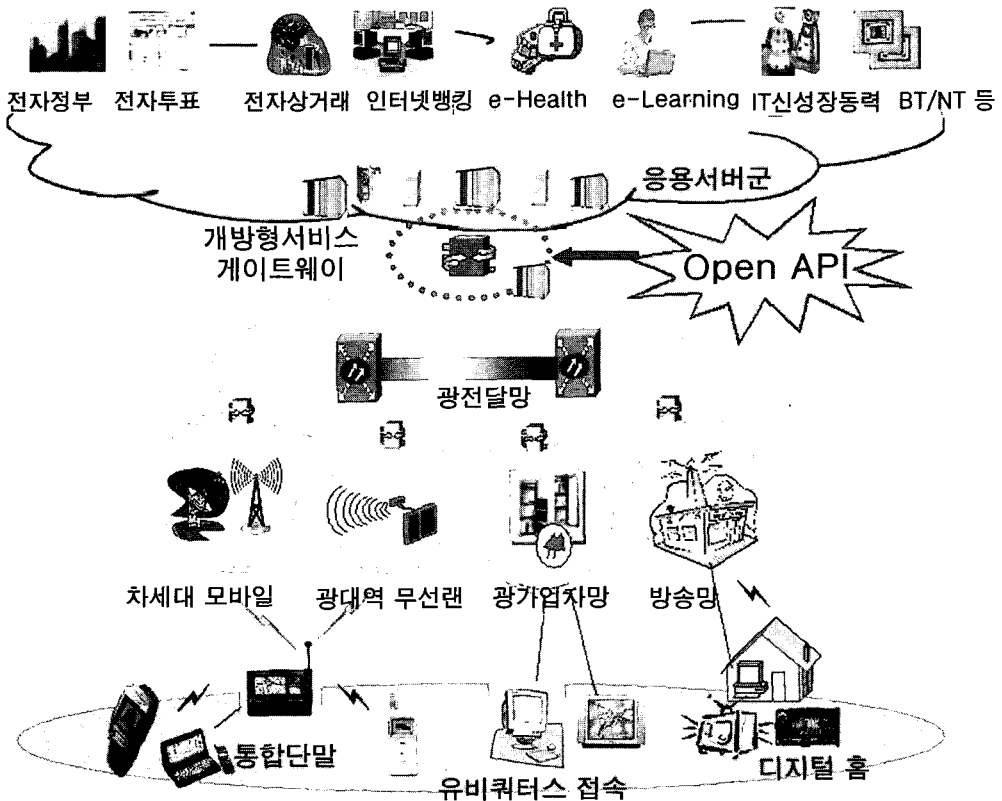
BcN의 가장 큰 특징은 컨버전스와 개방으로 요약될 수 있다. 통신, 방송, 인터넷 등 개별 네트워크로 존재하던 여러 종류의 통신망이 IP를 기반으로 하는 단일 망으로 묶이고 있으며, 제3의 어플리케이션 개발자들도 통신, 방송, 인터넷이 융합된 다양한 부가 서비스를 쉽게 개발할 수 있도록 하기 위해 통신망의 기능들이 표준화된 방법을 통해 외부로 공개되고 있다.

이러한 개방형 서비스 구조를 통해, 통신망의 기능을 이용하는 기업용 응용 소프트웨어에 의한 새로운 수익 사업의 발굴, 제3의 서비스 사업자를 수용하는 사업 모델의 구축, 통신과 방송 기능이 결합된 융합형 서비스의 개발 등 통신사업자들의 BcN의 사업 모델을 확대할 수 있게 된다[1].

개방에 있어 중요한 요소는 누구나 쉽고 일관성 있는 방법으로 이용할 수 있도록 해야 한다는 점이며, 이를 위해 통신망의 기능들에 대한 개방형 서비스 인터페이스의 표준화가 추진되고 있다. 개방형 서비스 인터페이스 (Open API)는 Parlay 그룹을 중심으로

CORBA 기반의 Parlay/OSA (Open Service Access) API와 웹서비스 (Web Service) 기반의 Parlay X API로 이원화 되어 있다. Parlay/OSA API는 통신망의 기능에 대한 비교적 풍부한 API 군을 제공하는 반면, 그 사용법이 다소 복잡하고, Parlay X API는 통신망의 기능에 대해 단순화 된 API를 제공함으로써, 통신망에 대한 지식이 거의 없는 사람이라도 API를 사용하여 새로운 서비스를 쉽게 개발할 수 있도록 한다[2-5].

본 논문에서는 차세대 통신 서비스 개발 패러다임으로 부상하고 있는 웹서비스 기반의 개방형 서비스



(그림 1) 광대역 통합망 (BcN)

기술에 대하여 살펴본다. 본 논문의 구성은 다음과 같다. II장에서는 개방형 서비스 구조를 설명하고, III장에서는 Open API 표준화 동향과 IV장에서는 웹서비스 기술의 통신망 적용에 대하여 설명한다. V장에서는 웹서비스 기반의 개방형 서비스 제어 구조를 비교 분석하고, VI장에서는 국내외 기술 동향과 VII장에서는 Open API 관련 국제상호운용성 시험에 관하여 설명하고, VIII장에서 결론을 맺는다.

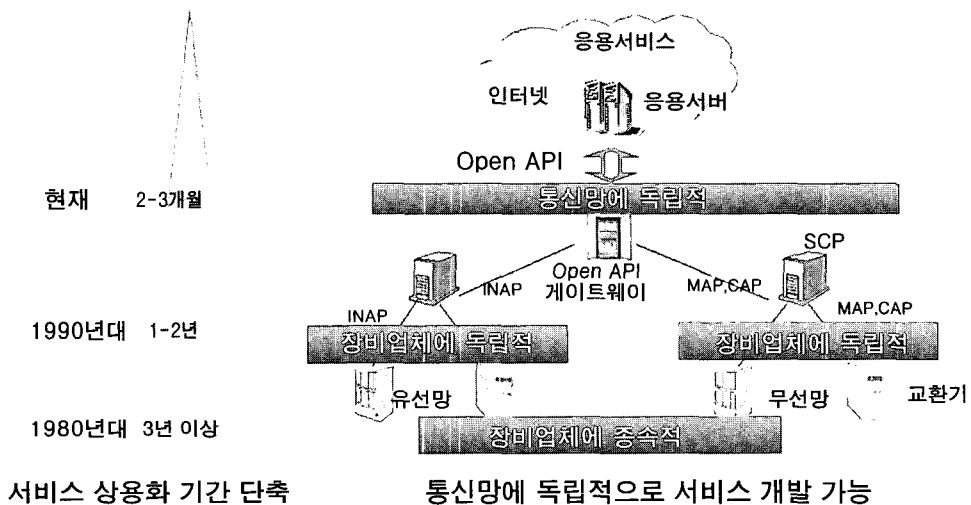
II. 개방형 서비스 구조 개요

통신 서비스의 구조는 (그림 2)와 같이 프로토콜 기반의 지능망 서비스 형태에서 Open API 기반의 개방형 서비스 구조로 진화되고 있다. 1980년대 모든 통신 서비스가 교환기에서 제공됨으로 인해 서비스의 개발 및 유지 보수를 교환기 벤더에 의존하던 방식에서, 1990년대 통신망 서비스 로직을 교환기로

부터 분리하는 지능망 개념의 도입으로 인해 통신 사업자 중심의 서비스 개발이 이루어지게 되었다. 그러나, 각 사업자 별로 구축된 지능망에서의 통신 서비스 개발자는 지능망 프로토콜 전문가로 제한되게 되며, 동일한 내용의 서비스가더라도 사업자 별로 중복하여 개발되고 관리되는 문제점을 가지게 된다. 이러한 문제점을 해결하기 위해 표준화된 인터페이스를 도입하여 통신망의 서비스 계층을 통신망의 제어 및 전송 계층으로부터 분리하는 개방형 서비스 구조가 차세대 통신망의 요구사항으로 제시되고 있다.

개방형 서비스 구조란 통신망의 기능들을 표준화된 인터페이스로 정의하고, 응용 프로그램들이 사용할 수 있도록 개방하여, 통신망의 구조에 독립적으로 다양한 부가 서비스가 개발될 수 있도록 하는 서비스 개발 패러다임이다.

응용 서비스 계층과 전달망 계층 사이의 표준화된 인터페이스를 Open API (Application Programming Interface)라고 하며, 호 제어 (call control), 이



(그림 2) 통신 서비스 구조의 진화

동성 제어 (mobility), 과금 제어 (charge), 프레즌스 관리 (presence management) 등의 통신망 기능들이 추상화 되어 API로 규격화 되고 있다. 이렇게 통신망의 기능들이 Open API로 개방됨으로써 통신망 (유선망, 무선망, IP망, 방송망 등)의 구조 또는 프로토콜 기술에 독립적으로 차세대 통신망에서의 통신 서비스들이 쉽게 개발될 수 있게 되며, 더 나아가 기업 응용에서 개발된 웹서비스들과 조합하여 다양한 신규 서비스의 개발이 가능하게 된다[2-3].

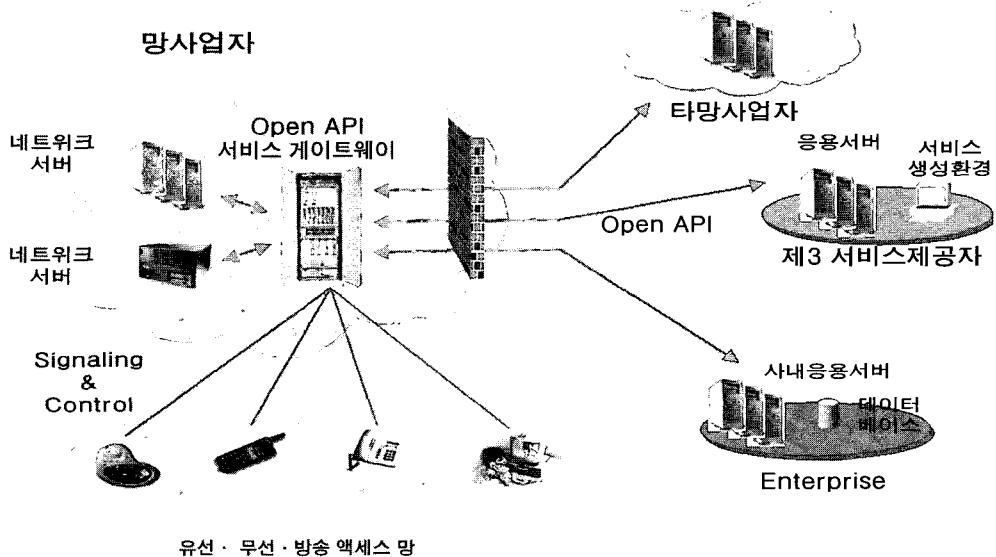
(그림 3)과 같이 통신망 사업자는 고유의 통신 서비스 제공자에서 제 3의 서비스 제공자에게 통신 웹 서비스를 판매하는 사업자로 사업 영역을 확대할 수 있게 된다. 이러한 사업 모델에서 통신망 사업자의 역할은 킬러 서비스를 개발하여 제공하는 서비스 사업자가 아니라, 제 3의 서비스 사업자들이 다양한 킬

러 서비스들을 개발할 수 있도록 환경 (Killer Enabling Environment)을 제공하는 것으로 그 역할이 재정립 될 수 있다.

통신망 사업자가 판매한 통신 웹서비스를 이용하면 일반 서비스 사업자 또는 개인들까지도 창의적인 서비스의 개발이 가능하게 된다. 보안과 품질이 보장된 통신 웹서비스를 이용하게 되므로, 신규 서비스 개발 시 투자에 대한 부담을 줄여 다양한 서비스 개발을 시도할 수 있고, 서비스들을 적기에 제공하여 사용자의 요구를 충족시킬 수 있게 된다.

또한 기업에 구축된 다양한 데이터베이스와 응용 소프트웨어들과 통신 기능들을 결합한 창조적인 서비스들을 발굴하여 통신 산업계의 활성화를 촉진할 수 있게 된다.

광대역 통합망에서 Open API를 통해 제공되는 서



(그림 3) Open API 서비스 제공 구조

비스는 각 영역의 인프라 내에서만 제한적으로 제공되었던 기존 서비스 간의 벽을 허물 것으로 예상됨에 따라 일반 서비스 사용자는 고정 및 휴대 통신 단말(이동/유선 전화, 휴대 인터넷 단말, 기타 통신 장비)에서 통신, 방송, 인터넷의 구분 없이 다양한 콘텐츠를 기반으로 하는 개인 중심의, 이동형, 멀티미디어 서비스를 제공받을 수 있게 될 것이다.

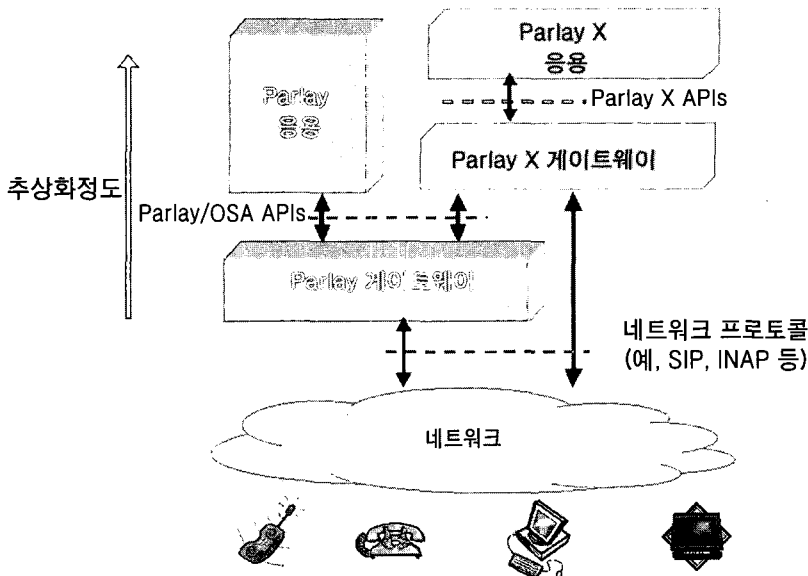
개방형 서비스 구조를 통신망에 적용하기 위해서는 서비스 게이트웨이와 서비스를 수행하는 응용 서버가 필요하다. 게이트웨이는 서비스 개발자들이 통신망 자원에 접근할 수 있도록 표준화된 API를 제공하며, API를 통신망의 프로토콜로 변환하고 제어하는 기능을 수행한다.

응용 서버는 Parlay X API를 이용하여 응용 서비스를 개발하고 운용할 수 있도록 서비스 생성 및 실행 기능을 수행한다.

III. Open API 표준화 동향

Open API의 표준화는 여러 단체에서 진행되고 있으나, Parlay 그룹을 중심으로 단일화 된 표준안이 추진되고 있다[2]. Parlay 그룹에서 표준화가 진행 중인 Open API는 통신망의 기능을 추상화하는 정도에 따라 CORBA 기반의 Parlay/OSA API와 웹서비스 기반의 Parlay X API로 나뉘어진다(그림 4).

Parlay/OSA API는 지능망 프로토콜을 추상화하여 발전시킨 API로서 CORBA 기반의 IDL (Interface Definition Language)로 정의되었다. Parlay/OSA API 관련 최신 규격은 2004년 9월 Parlay/OSA API 5.0 (ETSI OSA Release 3, 3GPP OSA Release 6)로서 14개의 규격으로 이루어지며, 12개의 통신망 기능들이 UML (Unified Modeling Language) 기반의 인터페이스 클래스와 메소드



(그림 4) Parlay/OSA API와 Parlay X API

(Method) 형태로 정의되어 있다.

Parlay X API는 웹서비스 기반의 인터넷 응용에서 통신망의 기능을 쉽게 사용할 수 있도록 하기 위한 목적으로, 통신망의 기능에 익숙하지 않은 IT 기반의 개발자들을 위해 통신망의 기능들을 최대한 단순화시켜 XML 기반의 WSDL (Web Service Description Language)로 정의한 인터페이스이다. 2003년 4월 V1.0을 시작으로 2004년 6월 V1.0.1 규격이 표준화되었으며, V2.0은 3GPP R6로 공표되었고,

ETSI에서는 승인 단계에 있다. Parlay X API 2.0 규격에는 V1.0.1의 8가지 통신 기능들에 5가지 기능이 추가되어 13가지의 통신 기능들에 대한 인터페이스들이 <표 1>과 같이 정의되어 있다[2].

IV. 웹서비스 기술의 통신망 적용

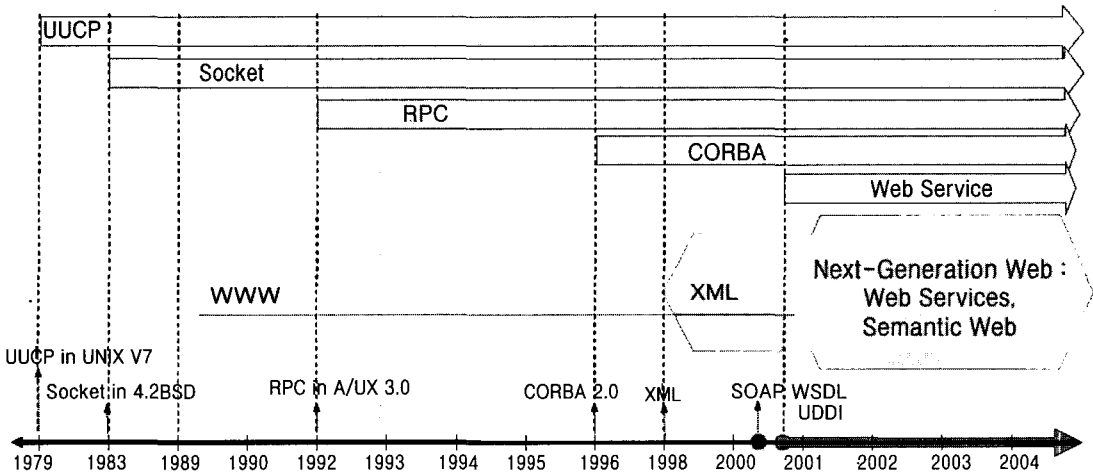
Parlay X API의 기반은 IT 의 웹서비스 기술이다.

<표 1> Parlay X API V2.0 규격

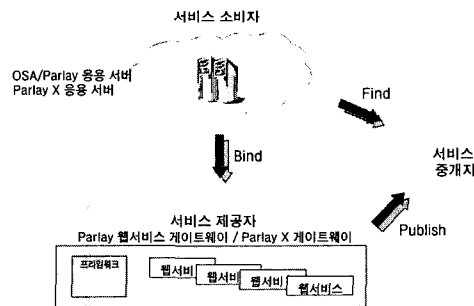
기능 (SCF)	내 용	APIs
Third Party Call	제 3자가 양자간의 호를 제어하는 기능	MakeACall(), GetCallInformation(), EndCall(), CancelCall()
Call Notification	통신망의 단말에서부터 발생한 호를 응용 서버 측으로 알려주는 기능	HandleBusy(), HandleNotReachable(), HandleNoAnswer(), HandleCalledNumber(), HandleBusy(), NotifyNotReachable(), NotifyNoAnswer(), NotifyCalledNumber()
Short Messaging	단문 문자 메시지 (SMS) 송수신 기능	SendSms(), SendSmsLogo(), SendSmsRingtone(), GetSmsDeliveryStatus(), NotifySmsReception(), GetReceivedSms()
Multimedia Messaging	멀티미디어 메시지 (MMS) 송수신 기능	SendMessage(), GetMessageDeliveryStatus(), GetReceivedMessages(), GetMessageURIs(), GetMessage(), NotifyMessageReception()
Payment	데이터의 볼륨 또는 금액별 과금 기능 (예약, 부과 기능)	ChargeAmount(), RefundAmount(), ChargeVolume(), GetAmount(), RefundVolume(), ReserveAmount(), ReserveAdditionalAmount(), ChargeReservation(), ReleaseReservation(), GetAmount(), ReserveVolume(), ReserveAdditionalVolume(), ChargeReservation(), ReleaseReservation()
Account Management	과금과 관련된 계정 관리 기능	GetBalance(), GetCreditExpiryDate(), BalanceUpdate(), VoucherUpdate(), GetHistory()
Terminal Status	단말의 상태를 파악하기 위한 기능	GetStatus(), GetStatusForGroup(), StartNotification(), EndNotification(), StatusNotification(), StatusError(), StatusEnd()
Terminal Location	단말의 위치를 파악하기 위한 기능	GetLocation(), GetTerminalDistance(), GetLocationForGroup(), StartGeographicalNotification(), StartPeriodicNotification(), EndNotification(), LocationNotification(), LocationError(), LocationEnd()
Call Handling	호 수용 및 차단 기능	SetRules(), SetRulesForGroup(), GetRules(), ClearRules()
Audio Call	TTS 및 VoiceXML을 이용한 음성 호 처리 기능	PlayTextMessage(), PlayAudioMessage(), PlayVoiceXmlMessage(), GetMessageStatusEndMessage()
Multimedia Conference	멀티미디어 회의 생성 및 관리 기능	createConference(), getConferenceInfo(), endConference(), inviteParticipant(), disconnectParticipant(), getParticipantInfo(), getParticipants(), addMediaForParticipant(), deleteMediaForParticipant()
Address List Management	주소 개별 및 Group 관리 기능	createGroup(), queryGroups(), setAccess(), queryAccess(), addMember(), addMembers(), deleteMember(), deleteMembers(), queryMembers(), addGroupAttribute(), deleteGroupAttribute(), queryGroupAttributes(), addGroupMemberAttribute(), deleteGroupMemberAttribute(), queryGroupMemberAttributes(), addMemberAttribute(), queryMemberAttributes(), deleteMemberAttribute()
Presence	프레즌스 정보 관리 기능	subscribePresence(), getUserPresence(), startPresenceNotification(), endPresenceNotification(), statusChanged(), statusEnd(), notifySubscription(), subscriptionEnded(), publish(), getOpenSubscriptions(), updateSubscriptionAuthorization(), getMyWatchers(), getSubscribedAttributes(), blockSubscription()

웹서비스는 인터넷상에서 기업간에 존재하는 응용 프로그램들을 XML (eXtensible Markup Language) 기반으로 상호 연동시키는 표준화된 소프트웨어 기술로 (그림 5)와 같이 분산 컴퓨팅 기술이란 이름으로 발전되어 온 기존 여러 기술들의 문제점을 극복하여 최종단계에 도달한 기술로 평가되고 있다[6]. 구조적 관점에서 웹서비스는 서비스 지향적 구조 (Service-Oriented Architecture)를 실현하는

기술로서, 서비스를 제공하는 서비스 제공자 (service provider), 서비스를 제공받는 서비스 소비자 (service consumer 혹은 service requester), 그리고 서비스의 공급자와 소비자를 중개해 주는 서비스 중개자 (service broker)로 구성된다. 서비스 소비자는 서비스 중개자를 통해 찾고자 하는 웹서비스의 제공자를 찾은 후 (FIND), 그 제공자가 제공하는 (PUBLISH) 서비스에 바인딩 (BIND)하여 호출함으



(그림 5) 분산 처리 기술의 발전



(그림 6) Parlay X 웹서비스 구조

로써 서비스를 제공받는다.

통신 서비스 산업에서도 프로그램의 용이성으로 다수의 개발자를 확보하기 위해 웹서비스 기술을 도입하기 시작했으며, 개방형 서비스 구조에 웹서비스 기술을 적용하여 Parlay X API가 정의되었다.

웹서비스 기술을 적용한 Parlay X API 실현 구조는 (그림 6)과 같다. Parlay X 서비스 게이트웨이가 서비스 제공자로서 추상화 된 망 기능을 웹서비스 형태로 공표하고 (Publish), 응용은 필요한 망 기능을 서비스 브로커를 통해 찾아 (FIND) 연결한 후 (BIND) 서비스 소비자로서 이를 호출하는 방식의 트라이앵글 웹서비스 구조에 따라 서비스를 제공할 수 있다.

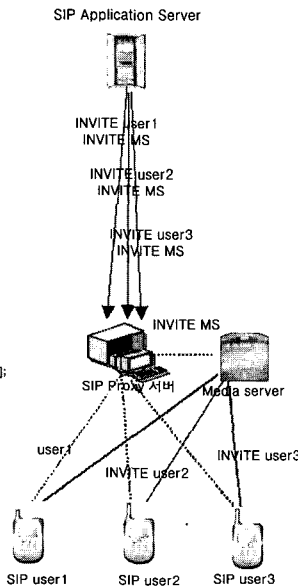
V. 웹서비스 기반의 개방형 서비스 제어 구조

차세대 통신망을 위한 서비스 제어 구조는 API 기반의 제어 구조와 프로토콜 기반의 제어 구조로 대별된다. API 기반의 제어 구조로 하부망에 독립적인 Open API를 정의하여 서비스 개발에 사용하며, 프로토콜 기반의 서비스 제어 구조에서는 All IP망을 목표로 하는 BcN에서의 제어 프로토콜인 SIP을 이용한다.

API 기반의 서비스 제어 구조에서는 규격화된 API를 사용함으로써 프로그램이 용이하여 다수의 개발자를 확보할 수 있고, 개발 기간이 짧은 반면, 프로토

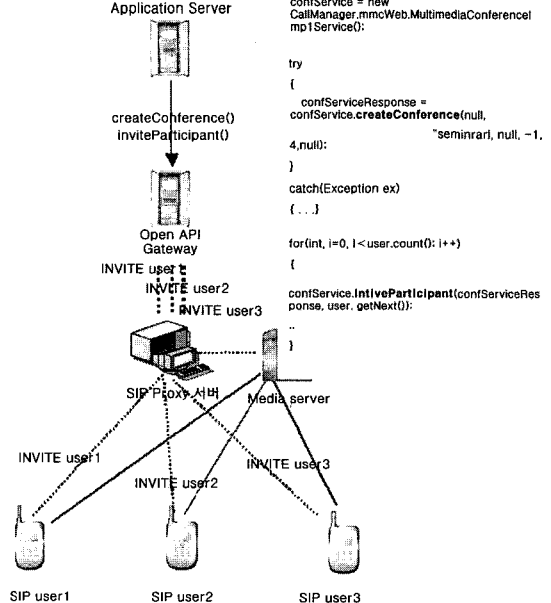
```
createConference(userList)
{
    i:=: userURL = getURL[userList, i++];
    //invite user
    for[userURL!=null]{
        msg=encode[INVITE, userURL];
        sendMsg(msg);
        do {
            rcvMsg[msg];
            rcvdMsg = decode[msg];
        } while[rcvdMsg!=ProvResp];
        if[rcvdMsg==Final Resp]
            userSDP = getSDP[Final Resp];
            encode[INVITE, MSURL, userSDP];
            sendMsg[INVITE, MS_URI];
            do {
                rcvMsg[&msg];
                rcvdMsg = decode[msg];
            } while[rcvdMsg!=ProvResp];
            if[rcvdMsg == FinalResp]
                msSDP = getSDP[FinalResp];
            else ...
                msg=encode[ACK, userURL, msSDP];
                sendMsg[MSG];
            }
        else.
            userURL = getURL[userList, i++];
    } //for
} //createConference
```

SIP 응용



(가) SIP 프로토콜 이용

Open API 응용



(나) 웹서비스 기반의 Parlay X API 이용

(그림 7) 프로토콜 기반과 API 기반의 제어 구조 비교

콜 기반의 서비스 제어 구조에서는 프로토콜의 이행을 바탕으로 서비스를 개발해야 하므로 API 기반의 개발 방법보다 복잡하여 서비스 개발 기간이 길어지나, 망 연동에는 용이한 장점이 있다.

본 장에서는 멀티미디어 회의 통화와 3자 연결 통화 서비스를 대상으로, API 기반의 제어 구조와 프로토콜 기반의 제어 구조를 비교 분석한다. 멀티미디어 회의 통화 서비스는 3명 이상의 사용자가 음성/영상 회의 통화를 수행하는 서비스이고, 3자 연결 통화 서비스는 제 3자(사람 또는 응용)가 2명의 사용자 사이에 호를 연결해 주는 서비스로서 모두 SIP 단말을 이용한다.

(그림 7)은 멀티미디어 회의 통화 서비스 개발 시, (가) SIP 프로토콜을 이용하는 방법과 (나) Open API를 이용하는 개발 방법에 대해 네트워크 구조와 프로그램을 비교하여 도시하였다. (가)의 경우는 Open API 플랫폼을 사용하지 않고, SIP 응용 서버에서 직접 멀티미디어 회의 호를 제어하는 경우로, SIP 응용 서버는 멀티미디어 회의 호 제어 로직과 SIP 프로토콜 처리부를 포함한다.

따라서 멀티미디어 회의 통화를 제공하는 SIP 응용 서버는 회의와 사용자 들간 세션 설정 및 해제, 미디어 믹싱을 위한 미디어 서버와 사용자간 세션 설정 및 해제 기능을 SIP 프로토콜에 기반하여 구현되어야 한다. (나)와 같이 웹서비스 기반의 Parlay X API를 이용하는 경우에는, 응용 서버와 Open API 게이트웨이 간에 Parlay X API인 createConference(컨퍼런스_정보)와 inviteParticipant(초대자_주소)를 초대하는 사람 수만큼 차례로 호출하면 게이트웨이의 MMC (Multimedia Conference) 웹서비스가 하부 SIP 망을 제어하여 회의 통화가 연결되도록 한다. 즉, (가)의 SIP 응용 서버가 가지고 있는 서비스 로직을 Open API 게이트웨이가 대신 수행함으로써 멀티미디어 회의 호 연결을 위해 관련 API를 호출하

는 것으로 간단히 구현할 수 있게 된다.

(그림 8)은 3자 연결 통화 서비스 개발 시, (가) CORBA기반의 Parlay/OSA API를 이용하는 방법과 (나) 웹서비스 기반의 Parlay X API를 이용하는 개발 방법에 대해 네트워크 구조와 프로그램을 비교하여 도시하였다. 오퍼레이션의 회수로 비교해 볼 경우, (가)의 경우 응용과 게이트웨이 간에 10개의 Parlay/OSA API로 호를 연결할 수 있는 반면 (나)의 경우에는 단 1개의 Parlay X API 만으로 호 연결이 가능하다.

프로그램 개발의 편의성 관점에서, (나)의 경우는 '전화 걸기'라는 일반적 개념의 makeCall()이라는 API를 호출하는 반면, (가)의 경우 지능망 프로토콜에 기반한 기능을 수행하는 상세 API들을 호출하게 되므로 지능망에 관련 지식이 없는 개발자라면, 개발 뿐 아니라 프로그램 유지 보수에 좀더 많은 비용이 소요될 것이다.

그러나 보다 섬세하고 복잡한 망의 기능을 다루어야 하는 응용이라면 Parlay X API보다는 Parlay/OSA API를 사용하는 것이 프로그램의 목적을 달성하기 유리하다.

VI. 국내의 기술 동향

2000년 까지만 하더라도 회의적으로 여겨졌던 Open API 기술은 해가 갈수록 다양한 제품들로 상용화되고 있으며, 2004년 11월 기준으로 <표 2>와 같이 Parlay/OSA API 관련 238 개의 제품이 개발되고 있고, 해마다 약 38% 씩 제품이 증가하고 있다고 보고 되고 있다²⁾.

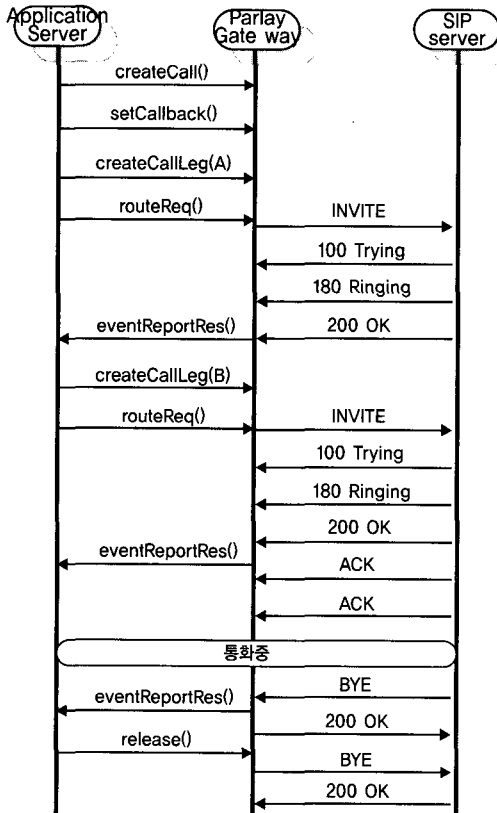
<표 3>은 웹서비스 기반의 Open API인 Parlay X 게이트웨이 개발 제품으로 현재 통신망에 적용되었거나 상용화된 제품 현황을 나타낸다. Argela (터어

〈표 2〉 Parlay/OSA API 관련 제품 개발 현황

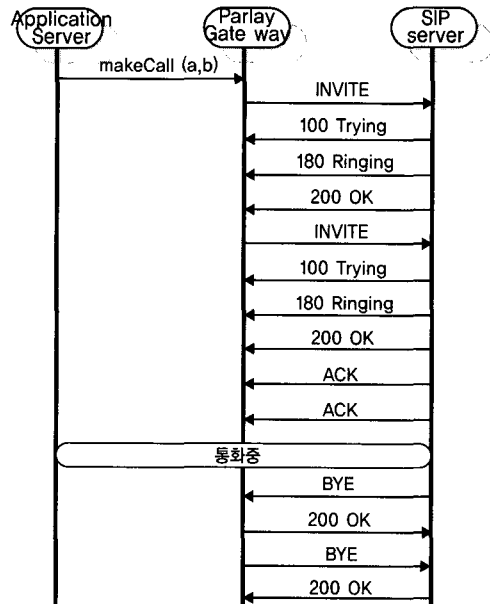
Product Type	Number
Parlay/OSA Gateway	27
Parlay/OSA Application Now 42% of all products	101
Parlay-X Platforms	11
Application Server	20
Development Environment / SCE	12
Simulators and test tools	8
Specialised SCS	3
Developer Progra	9
Analyst Reports	4
Other	22
Courses and Events	19
Total	238

〈표 3〉 웹서비스 기반의 Open API 게이트웨이 제품

회사	제품명	특징
AePONA	Cause Way	NGN에서의 서비스 플랫폼 제공
Appium	EWay XWay	Parlay/OSA 게이트웨이 Parlay X 게이트웨이
Argela	Argela IN	Parlay/OSA 게이트웨이 Parlay X 게이트웨이
Ericsson	NRG (Network Resource GW)	Parlay X API와 Parlay/OSA API 지원
IBM	WEST	Parlay X 1.0 지원 게이트웨이와 Parlay 응용 서버 결합
Lucent	MLife ISG	Parlay/OSA V4.1, Parlay X API 지원
인프라벨리	Parlay 게이트웨이	Parlay 4.1 APIs Frameworks Parlay X 1.0 GW(예정)
유엔셀	Parlay 게이트웨이	OSA/Parlay/OSA API 4.1 and 3.1 Parlay X 1.0 GW(예정)



(가) CORBA기반의 Parlay/OSA API 이용

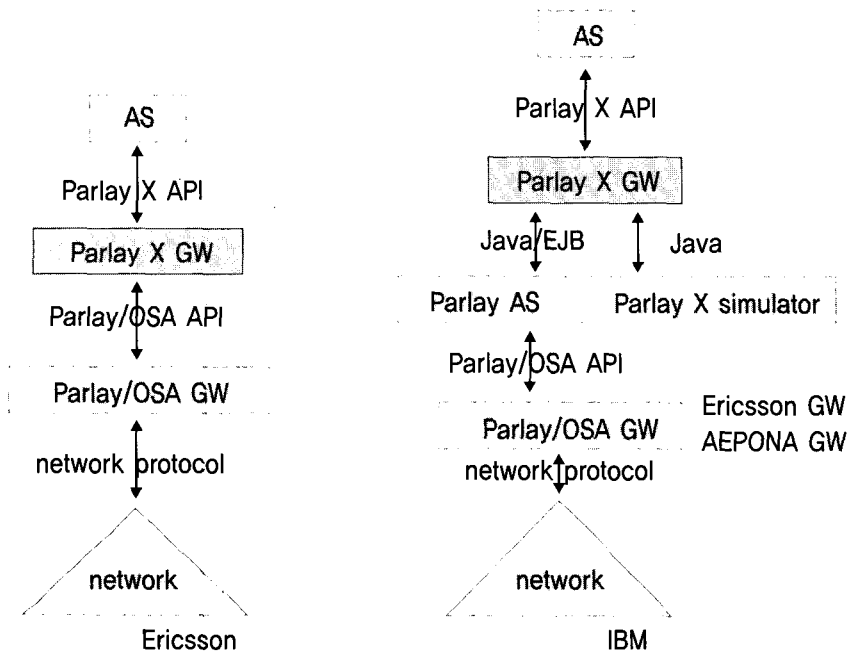


(나) 웹서비스 기반의 Parlay X API 이용

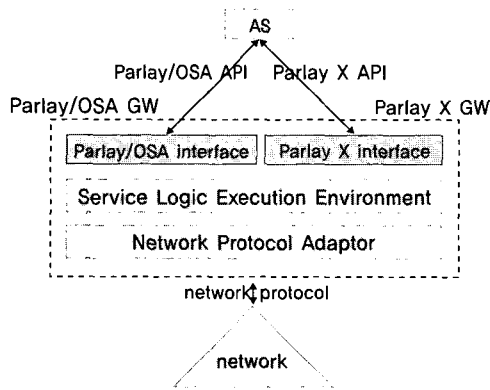
(그림 8) Parlay/OSA API와 Parlay X API 간의 제어 구조 비교

키), IBM, Ericsson, Telcordia 등이 Open API 게이트웨이에서 Parlay/OSA API와 Parlay X 웹서비스를 함께 제공하고 있다[3].

(그림 9)는 <표 3>에 나타난 Parlay X 웹서비스를 제공하는 게이트웨이 개발 업체 중 Ericsson, IBM의 상용 제품과 국내 유엔젤과 인프라밸리에서 개발이



(가) 사례 1



(나) 사례 2

(그림 9) Open API 게이트웨이 실현 구조 사례

진행되고 있는 개방형 서비스 게이트웨이의 시스템 구조를 비교한 그림이다.

(그림 9)의 사례 1로서 Ericsson과 IBM의 Parlay X 게이트웨이는 CORBA기반의 Parlay/OSA 게이트웨이와 연동하여 네트워크의 기능을 이용하고 있다. 이는 Parlay 게이트웨이가 이미 망에 적용되어 있을 경우 Parlay X 게이트웨이를 Parlay 응용 서버로 구축하여 Parlay 게이트웨이에서 제공하는 망의 기능을 간접적으로 이용할 수 있게 하는 구조이다.

사례 2의 개방형 서비스 게이트웨이는 Parlay X API와 Parlay/OSA API가 공동의 서비스 로직 실행 환경을 통하여 개별적으로 망과 연동하는 구조를 갖는다. 따라서 사례 2의 구조에서 Parlay X 웹서비스는 Parlay/OSA 게이트웨이 구축과 무관하게 독립적으로 제공될 수 있다. 기존에 Parlay 게이트웨이가 적용된 망에서는 대부분 (그림 9)의 사례 1과 같이 Parlay X 게이트웨이를 적용하고 있으며, 개방형 서비스 기술을 새로이 도입하는 사업자의 경우에는 사례 2와 같이 적용하여 구축하고 있다.

〈표 4〉는 웹서비스 기반의 Open API를 지원하는 응용 서버 제품 현황이고, 〈표 5〉는 Open API 응용 서비스 현황이다[2].

〈표 4〉 웹서비스 기반의 Open API 응용 서버 제품

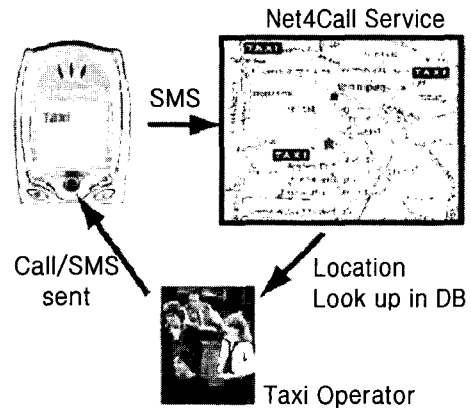
회사	제품명	특징
IBM	WS Server	Parlay X, Parlay/OSA 인터페이스 제공

Open API를 이용한 응용 서비스의 개발 사례는 표 5와 같으며, Call-by-Name, Phone Pages, iLocator 등을 대표적인 사례로 꼽을 수 있다.

(그림 10)의 Call-by-Name 서비스는 Net4Call 회사에서 제시한 서비스로서 통신망의 메시지 기능 및 호 제어 기능을 사업자의 데이터베이스와 결합한 서비스 유형을 제시한다. 예를 들어 택시가 필요한데

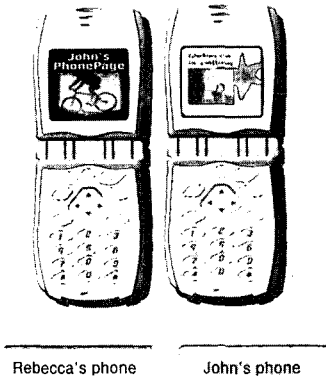
〈표 5〉 Open API 응용 서비스

회사	Application	특징
Ecrio	Rich Instant Messaging	무선망과 유선망을 이용한 Instant Messaging 서비스로 Yahoo, MSN, ICQ 등과 상호 연동
IBM	Agent Notification Service	User의 location에 기반하여 Voice, Email, Message, Web 등을 통합한 다양한 Notification 서비스를 제공
Lucenti	Locator	가족 또는 친구들이 특정 영역에 들어오거나 나갈 때 이동단말에 그들의 위치에 대한 그래픽 지도 정보를 제공함
Solomio	Smart Call Service	In-Call, Connect Call, Missed Call, Home Call, Office Call
Telenity	Smart Alert	부재중 도착한 통화에 대해 발신번호를 관리, 가입자가 전화할 수 있을 때 통화할 수 있도록 통보해주는 서비스
Phonepages	Phonepages	착신자와의 호 연결 시 착신자의 홈 페이지가 발신자에게 자동으로 제공되는 서비스



(그림 10) Open API 응용 서비스 - Call by Name

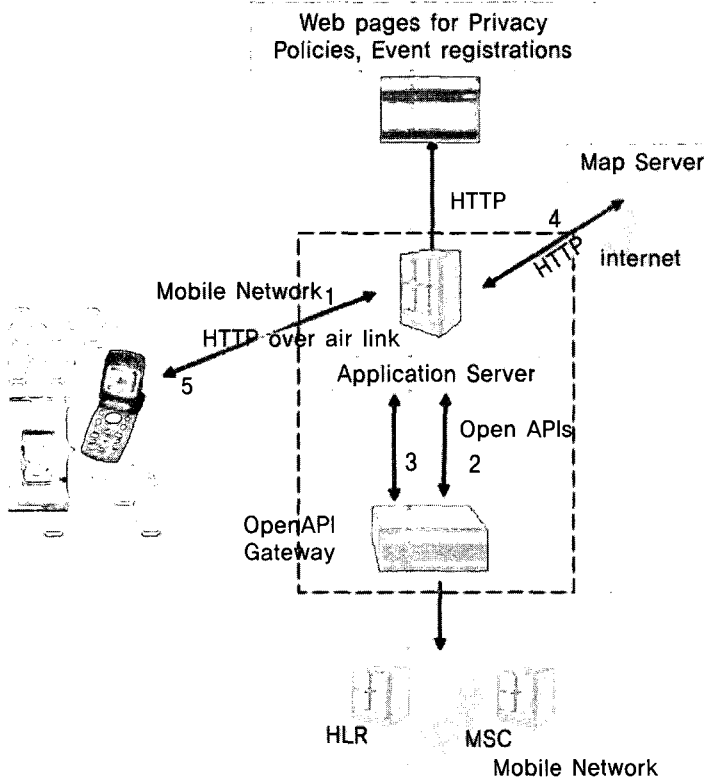
근처에 택시가 보이지 않는 경우, 서비스 사업자로 “택시”라는 SMS를 보내면, 서비스 사업자는 통신망의 “Terminal Location” 기능을 이용하여 발신자의 현재 위치를 파악하고, 데이터베이스에 등록된 택시 사업자들 중 발신자의 위치에 가까운 택시 사업자를 찾아내고 발신자의 전화 번호를 알려주어 택시를 보낼 수 있도록 한다. 이 서비스는 택시뿐 아니라 일반



(그림 11) Open API 응용 서비스 - Phone Pages

사업장의 위치를 지도를 통해 알려주거나, 사업장에서 제공하는 서비스 내용을 MMS나 WAP을 통해 알려줄 수 있으며, 또는 직접 통화할 수 있도록 서로 간에 호를 연결할 수도 있다.

Phonepages사의 응용 서비스는 (그림 11)에서와 같이 전화를 할 때, 단말기를 통해 상대방에 대한 홈페이지(home page)를 서로 검색할 수 있게 해준다. 통화가 연결되기 이전에 자신의 상태에 대한 정보를 홈페이지를 통해 알려 줄 수도 있으며, 통화가 끝난 후에도 상대방이 준비해둔 홈페이지를 계속하여 탐색할 수 있다. Phone Pages 서비스는 개인들 간의 정보 아니라, 일반 사업장에서도 사업 및 상품에 대한 소개를 홈페이지를 통해 할 수 있게 된다. 예를 들어 피자 가게의 경우, 전화를 한 고객에게 오늘의 특별 피자 메뉴는 무엇인지, 가격은 얼마인지, 오늘의 특별한 이벤트는 무엇인지, 가게의 좌석 상태는 어떠한지, 예약은 가능한지 등을 통화가 이루어지기 전에 홈페이지를 통해 알려줄 수 있다.



(그림 12) Open API 응용 서비스 - iLocator

Lucent에서 제시한 (그림 12)의 iLocator 서비스는 가입자의 위치를 기반으로 하여 사전에 등록해둔 사람이 일정 거리 영역 내에 들어오거나 나갈 때 그 사실을 통보해 주며, 사용자가 요구하는 근처의 교통 사항, 사업장 정보, 이벤트 정보 등을 지도를 통해 알려주는 서비스이다. 학교를 다니는 자녀가 지정된 반경 영역을 벗어나게 되면 이를 보호자에게 알려 주어 사고 유무를 확인할 수 있도록 할 수 있다.

VII. 국제 상호운용성 시험

Parlay X 웹서비스 국제 상호운용성 시험 행사는 ETSI와 MSF(Multi Service Forum)에 의해 시행되고 있다.

다양한 분야의 국제 상호운용성 시험 행사를 주관하고 있는 ETSI Plugtests에서 개방형 서비스 인터페이스 분야로는 2003년 1차 OSA/Parlay API 시험 행사를 시작하여, 2004년 2차 시험 행사를 개최하였으며, 2005년 3월 3차 시험 행사에서 Parlay X 웹서비스 상호운용성 시험을 주최하였다. 표 6과 같이 3차 시험 행사에는 ETRI, 프랑스 IBM, Ericsson 등 8개사가 참가하여 Parlay X API V2.0 (WS-I BP1.0 기준) 기반의 게이트웨이, 응용, 시뮬레이터 등을 대상으로 서비스와 장비 간 상호운용성 시험을 수행하였으며, 2005년 9월 4차 Parlay/OSA API 상호운용성 시험이 계획되어 있다.

차세대 VoIP 통신망에 대한 상호운용성 시험을 시행하고 있는 MSF는 MSF 구현 협약을 기반으로 차세대 통신망을 구축하였을 때의 상호운용성을 확보하기 위해 2002년부터 2년 주기로 유럽/아시아/미국을 연결하는 글로벌 네트워크를 구성하여 차세대 통신

망에 대한 가능성을 점검하는 GMI (Global MSF Interoperability) 상호운용성 시험 행사를 추진하고 있다 [8]. 2004년 시행된 GMI에서는 MSF 구현 협약을 기반으로 하여 Parlay/OSA API 및 Parlay X API를 적용한 click-to-connect 서비스, click-to-conference 서비스, GETS (Government emergency tele-phone service) 등의 응용 서비스를 VoIP망의 장비들과 연동하는 상호운용성 시험을 수행하였다.

VIII. 결론

IT를 중심으로 한 컨버전스, 유비쿼터스 시대의 도래는 개인화 된 다양한 통신 서비스의 개발에 대한 새로운 패러다임을 요구하고 있다. 이를 위해 차세대 통신 인프라의 기능들이 표준화 된 인터페이스를 통해 개방되고 있으며, 이는 서비스 지향 구조를 구현하는 웹서비스 기술로 현실화되고 있다.

이러한 웹서비스 기반의 개방형 서비스 기술은 IT 839에서 추진되고 있는 모든 융합형 서비스의 개발 패러다임으로 적용되어, 신규 비즈니스 모델을 발굴

〈표 6〉 ETSI 3차 OSA/Parlay 상호운영 시험 행사

참가장비	참가사	참가 SCF
Parlay X Gateway & Network Simulator	ETRI	TPC, Call Notification, Payment, Account Management, Terminal Location, Terminal Status, Multi-Media Conference
	Ericsson	SMS, MMS
	TILAB	TPC, SMS, Terminal Location, multi-Media Conference
	IBM	TPC, SMS, Payment, Terminal Location, Terminal Status
	ARGELA	TPC, SMS
Application	AePONA	TPC, SMS, MMS
	ETRI	TPC, SMS, Payment, Account Management, Terminal Location, Terminal Status Multi-Media Conference
	IBM	TPC, SMS, Payment, Account Management, Terminal Location, Terminal Status
	TILAB	Third party call, SMS, MMC
	eBIZ Mobility	Payment

하고 시장 수요를 창출하는 기회를 제공할 수 있게 될 것이다.

이를 위해서는 차세대 통신망 인프라에 웹서비스 기반의 개방형 서비스 구조 구축에 통신 사업자들의 관심과 투자의 집중이 필요하다 하겠다.

[참고 문헌]

- [1] Jean-Yves Cochenec, "Activities on next-generation networks under global information infrastructure in ITU-T," IEEE Communications Magazine, vol. 40, no. 7, pp. 98-101, Jul, 2002.
- [2] <http://www.parlay.org> 발표 자료 및 white papers.
- [3] Zygmunt Lozinski, "Parlay/OSA - a New Way to Create Wireless Services," IEC Mobile Wireless Data, pp.61-70. Jun. 2003.
- [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, pp. 58~64, May/June 2004.
- [5] Ard-Jan Moderdijk, "Open Service Architecture: Concepts and Standards," Ericsson Technical Report, 2001.
- [6] 한국정보산업연합회, "뉴패러다임 & IT컨버전스 : 웹서비스(Web Service)," 2004년 민간백서, pp.123-129.
- [7] ETSI, "OSA/Parlay Interop Event 14-17 April 2003; Report," Apr. 2003.
- [8] MSF, "MSF GMI 2004 Technical Description," MSF, Dec. 2003.



박유미

1991년 숙명여자대학교 전산학과 (학사)
 1997년 충남대학교 컴퓨터과학과 (석사)
 1991년 ~ 현재 한국전자통신연구원 선임 연구원
 관심분야 : 차세대 네트워크, 개방형 서비스 기술,
 차세대 인터넷 프로토콜, SIP



최영일

1983년 서울대학교 전자공학과 (학사)
 1998년 충남대학교 컴퓨터과학과 (석사)
 2002년 충남대학교 컴퓨터과학과 (박사)
 1996년 정보통신 기술사
 1985년 ~ 1986년 Bell 연구소 객원연구원
 1983년 ~ 현재 한국전자통신연구원 (현 개방형서

비스플랫폼팀장, 표준전문위원)

2002년 ~ 정통부 국제표준전문가 (MSF, Parlay)

관심분야 : 차세대 네트워크, Softswitch, 개방형 서비스 기술



이병선

1980년 성균관대학교 수학과 (학사)
 1982년 동국대학교 전산학과 (석사)
 2003년 한국과학기술원 전산학과 (박사)
 1982년 ~ 현재 한국전자통신연구원 (현 BcN서
 비스연구그룹 그룹장)
 2000년 ~ MSF 이사, BcN포럼 운영위원, 정통부

국제표준전문가

관심분야 : NGN/BcN 구조, 개방형 서비스 기술, 소프트웨어 생성 자동화
 기술