

유비쿼터스 서비스를 위한 BcN의 개방형 서비스 인터페이스

최영일

◆ 목 차 ◆

1. 서 론
2. 통신 서비스 개발 구조의 진화
3. 개방형 서비스 구조
4. Open API 유형
5. 웹 서비스 기반의 Open API
6. 개방형 서비스 사례
7. 개발동향
8. 결 론

1. 서 론

컴퓨터와 통신을 융합하여 정보를 교환하게 된 제1의 정보 혁명, 인터넷을 통해 콘텐츠의 창출과 유통을 촉진한 제2의 정보 혁명을 지나, 일상생활 환경에 센서, 컴퓨터, 네트워크 장비 등을 보이지 않게 내장하고, 사람들이 인지하지 못한 상태에서 각종 편리한 서비스를 제공하는 제3의 정보 혁명이 오늘날 정보 기술 산업의 화두가 되고 있다. 이에 대해 우리나라에서는 유비쿼터스 코리아 (u Korea) 라는 패러다임을 차세대 IT 전략으로 하여 새로운 산업에 대한 성장 모멘텀의 형성을 추구하고 있으며, 이를 국가 발전의 원동력으로 승화하기 위한 유비쿼터스 IT 환경 구축이 IT839 정책으로 추진되고 있다.

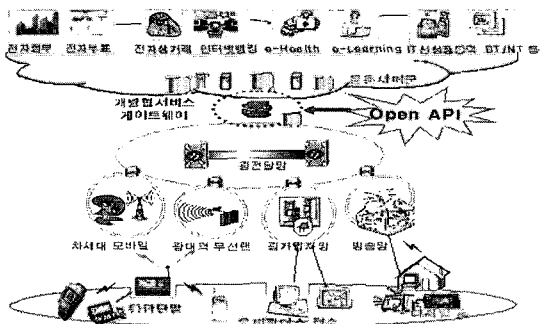
유비쿼터스 IT 환경은 유비쿼터스 컴퓨팅, 유비쿼터스 네트워크, 유비쿼터스 서비스라는 세가지 기반을 통해 실현된다. 유비쿼터스 컴퓨팅은 사물들에 칩, 센서, RFID 태그 등을 심거나 부착함으로써 컴퓨팅 능력과 통신 능력을 부여하는 것이며, 유비쿼터스 네트워크는 이러한 유비쿼터스 컴퓨팅 능력을 갖는 개체들을 브로드밴드, 위성, 모바일,

무선랜 등의 통신망을 통해 연결하는 것이다. 유비쿼터스 서비스는 유비쿼터스 컴퓨팅과 유비쿼터스 네트워크를 기반으로 정보통신 시스템의 기능이나 콘텐츠들을 서로 이음새 없이 (Seamless) 연계하고 통합함으로써 시너지를 창출하는 것이다. 이러한 유비쿼터스 컴퓨팅과 유비쿼터스 네트워크를 위한 인프라로서 IT839 를 통해 유비쿼터스 센서 네트워크 (USN)와 광대역 통합망 (BcN)의 구축이 본격적으로 추진되고 있다. 유비쿼터스 서비스는 RFID 서비스, 홈네트워크 서비스, 텔레매틱스 서비스 같은 다양한 서비스들이 융합됨으로써 가능해지며, 이와 같은 서로 다른 다양한 서비스들을 표준방식을 통해 연계하고 융합하는 기술로서 웹서비스 기술이 주목 받고 있다.

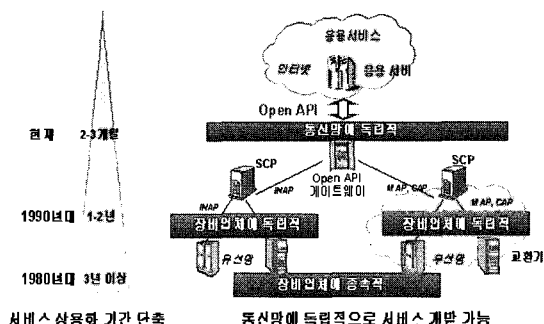
광대역 통합망 (Broadband convergence Network) 에서도 통신망 사업자나 서비스 사업자들이 유무선 통신망 등의 네트워크 하부구조에 독립적으로 통신서비스를 정의하고, 구현하기 위해 웹서비스 기술을 도입하고 있다. 본 논문에서는 유비쿼터스 서비스의 실현을 위해 광대역 통합망 (BcN)에서 차세대 통신 서비스의 개발 패러다임으로 추진되고 있는 웹서비스 기반의 개방형 서비스 인터페이스 (Open API)에 대하여 살펴본다.

본 논문의 구성은 다음과 같다. 2장에서는 통신

* 한국전자통신연구원



(그림 1) 광대역 통합망 (BcN) 구조



(그림 2) 통신 서비스 개발 구조의 진화

서비스 개발 구조의 진화 방향에 대하여, 3장에서는 개방형 서비스 구조에 대하여, 4장에서는 Open API 유형에 대하여, 5장에서는 웹서비스 기반의 Open API에 대하여, 6장에서는 개방형 통신 서비스의 사례에 대하여 설명하고, 7장에서는 국내외 개발 현황에 대하여 살펴본다.

2. 통신 서비스 개발 구조의 진화

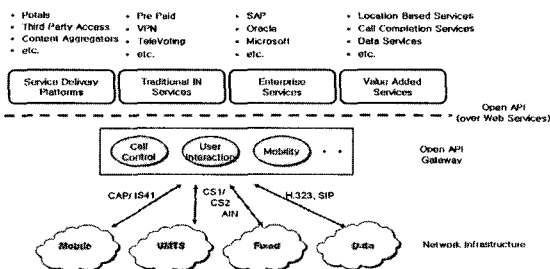
통신 서비스의 개발 구조는 그림 2와 같이 통신망에 대한 전문 지식 없이도 다양하고 풍부한 새로운 서비스를 개발 할 수 있도록 진화되고 있다.

1980년대 모든 통신 서비스가 각 단국 교환기에서 제공되었던 경우에는 이들 서비스의 개발을 전적으로 교환기 제조업체에 의존할 수 밖에 없어 새로운 서비스 개발에 대한 제한이 있었다. 이러한 종속 관계를 해결하기 위해 1990년대에 통신 서비스 로직을 교환기로부터 분리하는 지능망 개념이 도입

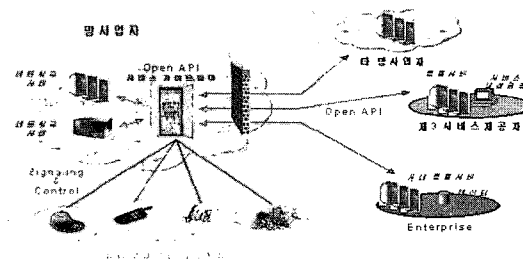
되었으며, 이로 인해 비로서 통신 사업자 중심의 서비스 개발이 이루어지게 되었다. 그러나, 지능망에서의 통신 서비스 개발도 지능망 프로토콜을 잘 아는 통신 전문가 집단으로 제한되게 되며, 또한 동일한 내용의 서비스이더라도 사업자별로 중복하여 개발되는 문제점을 가지게 된다. 이러한 문제점을 해결하기 위해 통신망의 서비스 계층을 통신망의 제어 계층으로부터 분리하고, 이들 간에 표준화된 인터페이스를 적용하는 개방형 서비스 구조가 새로운 통신 서비스의 개발 패러다임으로 제시되고 있다.

3. 개방형 서비스 구조

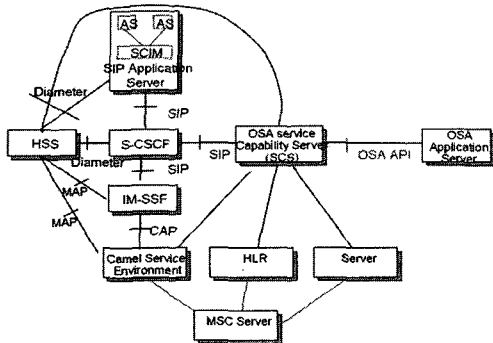
개방형 서비스 구조란 통신망의 기능들을 표준화된 인터페이스로 정의하고, 응용 프로그램들이 사용할 수 있도록 개방하여, 통신망의 구조나 프로토콜에 독립적으로 다양한 부가 서비스가 개발될 수 있도록 하는 서비스 개발 패러다임이다.



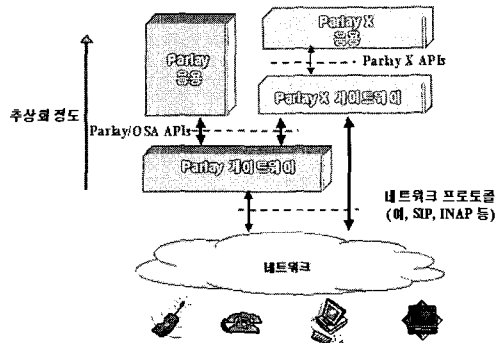
(그림 3) 개방형 서비스 구조



(그림 4) Open API 서비스 제공구조



(그림 5) IMS 구조



(그림 6) 2가지 유형의 Open API

응용 프로그램들이 사용할 수 있는 이러한 인터페이스를 Open API (Application Programming Interface)라고 한다. 호 제어 (call control), 이동성 (mobility), 과금 (charge), 프레즌스 관리 (presence management) 등의 통신망 기능들이 Open API로 개방됨으로써 통신망에 대한 전문 지식 없이도 통신 서비스들이 쉽게 개발될 수 있게 되며, 더 나아가 기업 응용에서 개발된 웹서비스들과 조합하여 다양한 신규 서비스의 개발이 가능하게 된다.

개방형 서비스 구조를 통신망에 적용하기 위해서는 응용 프로그램들이 통신망 자원에 접근할 수 있도록 Open API를 제공하는 Gateway가 필요하다. 이러한 개방형 서비스 Gateway는 응용 프로그램들이 이용하는 Open API를 통신망의 프로토콜로 변환하고 제어하는 기능을 수행한다.

개방형 서비스 구조를 통신망에 도입하면, 지능망 기반의 서비스 개발 시간에 비해 20% 수준으로 통신 서비스의 개발 시간을 단축할 수 있게 되며, IT 개발 인력을 서비스 개발 인력으로 활용할 수 있어 정보 기술과 통신 기술이 융합된 창의적인 서비스의 개발이 가능하게 된다. 또한 통신망 사업자의 고유 영역이었던 통신 서비스 영역을 규격화된 인터페이스를 통해 외부로 개방함으로써, 물리적인 통신 인프라를 갖고 있지 않은 제3의 서비스 사업자들을 VNO (Virtual Network Operator) 또는 VMNO (Virtual Mobile Network Operator) 등의 형태로 수용하고, 이들 서비스 사업자들을 통해 인터넷 기반의 창의적인 멀티미디어 서비스를 발굴하

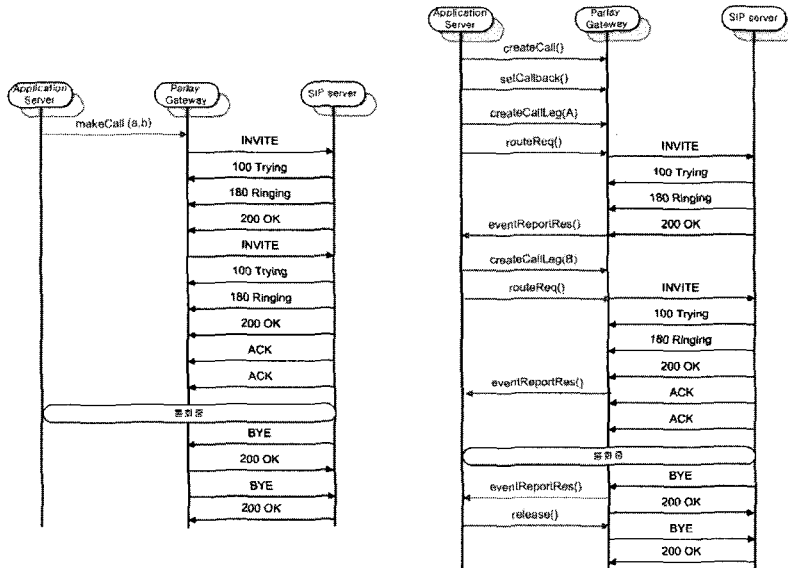
고 상용화할 수 있게 된다.

이러한 사업 모델에서 통신망 사업자의 역할은 서비스를 개발하여 제공하는 서비스 사업자가 아니라, 제 3의 서비스 사업자들이 다양한 서비스들을 개발할 수 있는 환경을 제공하는 것으로 그 역할이 재정의 될 수 있다. 이를 통해 신규 서비스 개발 투자에 대한 위험 부담을 줄일 수 있으며, 다양한 요구 사항에 맞는 서비스들을 적기에 제공하여 사용자의 요구를 충족시킬 수 있게 된다. 또한 기업에 구축된 다양한 데이터베이스, 응용 소프트웨어, IT 839의 다양한 서비스들과 통신 기능들이 결합된 유비쿼터스 서비스들을 발굴하여 통신 산업계의 활성화를 촉진할 수 있게 된다.

개방형 서비스 구조는 기존의 유무선 통신망 뿐만 아니라, 3GPP, ETSI, OMA, MSF, WWRF 등의 표준화 기관에서 제시하고 있는 차세대 통신망의 서비스 개발 방법으로서 채택되고 있다. 예를 들어, 유무선 통합망의 제어 구조로서 주목 받고 있는 3GPP의 IMS (IP Multimedia Subsystem) 구조에서도 그림 5와 같이 Open API 기반의 OSA (Open Service Access) 인터페이스가 필수 요구사항으로 제시되고 있다.

4. Open API 유형

개방에 있어 중요한 요소는 누구나 쉽고 일관성 있는 방법으로 이용할 수 있도록 해야 한다는 점이다. 이를 위해 Parlay Group, ETSI, 3GPP 등이



(그림 7) Parlay X API와 Parlay API의 제어 구조 비교

공동으로 Open API에 대한 단일 표준화를 추진하고 있으며, 통신망의 기능을 추상화하는 정도에 따라 CORBA 기반의 API (Parlay/OSA API)와 웹서비스 기반의 API (Parlay X API)로 분리되어 표준화가 진행되고 있다.

Parlay/OSA API는 통신망 프로토콜을 추상화하여 발전시킨 API로서 CORBA 기반의 IDL (Interface Definition Language)로 정의되어 있다. 통신망의 기능에 대한 비교적 풍부한 API 들을 제공하는 반면, 프로토콜에 대한 이해가 필요하여 그 사용법이 다소 복잡하다.

웹서비스 기반의 Parlay X API는 통신망 기능에 익숙하지 않은 서비스 개발자들이 쉽게 사용할 수 있도록 통신망 기능을 최대한 단순화시켜 XML 기반의 WDSL로 정의한 인터페이스이다. IT 분야의 새로운 기술 패러다임으로 부상하고 있는 웹 서비스 기술을 통신망에 적용하여, 통신망의 기능에 대해 단순화 된 API를 제공함으로써, 통신망에 대한 지식이 거의 없는 사람이라도 Parlay X API를 사용하여 통신망의 기능이 융합된 유비쿼터스 서비스를 쉽게 개발할 수 있도록 한다.

통신망의 가입자들 간의 통화 연결을 응용 서비스에서 제어하는 서비스를 개발하는 경우에 대하여, CORBA기반의 Parlay/OSA API를 이용하는 방법과 웹서비스 기반의 Parlay X API를 이용하는 방법에 대한 비교는 그림 7과 같다.

Open API는 응용 서버와 개방형 서비스 게이트웨이 간에 적용된다. CORBA방식의 Parlay/OSA API로 구현하는 경우는 callback 메커니즘을 기반으로 한 비동기 방식이며, 기존 프로토콜 레벨과 비슷한 수준의 API로 추상화된 반면, 웹서비스 기반의 Parlay X API는 통신망의 기능을 추상화하는 정도와 제어할 수 있는 granularity면에서 매우 단순해진 것을 알 수 있다.

5. 웹서비스 기반의 Open API

Parlay X API는 응용 서비스 개발에 필요한 통신망의 기능을 호출하여 사용할 수 있도록 통신망의 기능들을 웹서비스 형태로 제공한다. 2005년 6월 현재 V2.0 규격이 표준화되었으며 13가지의 통신 기능들에 대한 인터페이스들이 표 1과 같이 정

(표 1) Parlay X API V2.0 규격

SCF	APIs
Third Party Call	MakeCall(), GetCallInformation(), EndCall(), CancelCall
Call Notification	HandleBusy(), HandleNotReachable(), HandleNoAnswer(), HandleCalledNumber(), NotifyBusy(), NotifyNotReachable(), NotifyNoAnswer(), NotifyCalledNumber()
Short Messaging	SendSms(), SendSmsLogo(), SendSmsRingtone(), GetDeliveryStatus(), NotifySmsReception(), GetReceivedSms()
Multimedia Messaging	SendMessage(), GetMessageDeliveryStatus(), GetReceivedMessages(), GetMessageURIs(), GetMessage(), NotifyMessageReception()
Payment	ChargeAmount(), RefundAmount(), ChargeVolume(), GetAmount(), RefundVolume(), ReserveAmount(), ReserveAdditionalAmount(), ChargeReservation(), ReleaseReservation(), GetAmount(), ReserveVolume(), ReserveAdditionalVolume(), ChargeReservation(), ReleaseReservation()
Account management	GetBalance(), GetCreditExpiryData(), BalanceUpdate(), VoucherUpdate(), GetHistory()
Terminal Status	getStatus(), getStatusForGroup(), startNotification(), endNotification(), statusNotification(), statusError(), statusEnd()
Terminal Location	getLocation(), getTerminalDistance(), getLocationForGroup(), startGeographicalNotification(), startPeriodicNotification(), endNotification(), locationNotification(), locationError(), locationEnd()
Call Handling	SetRules(), SetRulesForGroup(), GetRules(), ClearRules()
Audio Call	playTextMessage(), playAudioMessage(), playVoiceXmlMessage(), getMessageStatus(), endMessage()
Multimedia Conference	createConference(), getConferenceInfo(), endConference(), inviteParticipant(), disconnectParticipant(), getParticipantInfo(), getParticipant(), addMediaForParticipant(), deleteMediaForParticipant()
Address List Management	createGroup(), deleteGroup(), queryGroups(), setAccess(), queryAccess(), addMember(), addMembers(), deleteMember(), deleteMembers(), queryMembers(), addGroupAttribute(), deleteGroupAttribute(), queryGroupAttributes(), addGroupMemberAttribute(), deleteGroupMemberAttribute(), queryGroupMemberAttributes(), addMemberAttribute(), queryMemberAttributes(), deleteMemberAttribute()
Presence	subscribePresence(), getUserPresence(), startPresenceNotification(), endPresenceNotification(), statusChanged(), statusEnd(), notifySubscription(), subscriptionEnded(), publish(), getOpenSubscriptions(), updateSubscriptionAuthorization(), getMyWatchers(), getSubscribedAttributes(), blockSubscription

의되어 있다.

5.1 Call Control 관련

응용 프로그램에서 통신망 가입자에 대한 호를 제어할 수 있도록 다음과 같은 4가지 그룹들의 다양한 호제어 기능들이 웹서비스로 제공된다.

5.1.1 Third Party Call 기능

응용 프로그램에서 통신망 가입자 간의 호를 설정할 수 있게 하는 기능으로서 다음과 같은 API를 제공한다.

- **MakeCall** (callingParty, calledParty, charging): 발신 및 착신 가입자 간의 호 연결
- **GetCallInformation** (callID): 호의 상태, 연결 시각, 통화 시간 등의 정보 획득
- **EndCall** (callID): 호 종료

5.1.2 Call Notification 기능

지정된 가입자의 호 상태가 통화중 이거나 무응답 일 경우, 또는 특정 서비스 번호에 대한 호가 발생하는 경우에, 이를 응용 프로그램이 통보 받아

호를 다른 착신지로 전환하거나 종료할 수 있게 하는 기능으로서 다음과 같은 API를 제공한다.

- **handleBusy** (calling, called): 착신 가입자가 통화 중일 때 호 전환, 종료 등 처리 제어
- **handleNotReachable** (calling, called): 호가 착신 측에 연결되지 못할 경우의 처리 제어
- **handleNoAnswer** (calling, called): 착신 가입자가 무응답 일 때 처리 제어
- **handleCalledNumber** (calling, called): 지정한 착신번호에 대한 호가 발생했을 때 처리 제어

5.1.3 Call Handling 기능

응용 프로그램에서 통신망의 특정 가입자에 대하여 착신되는 호에 대한 처리 방법을 지정할 수 있도록 하는 기능으로서 다음과 같은 API를 제공한다.

- **setRules** (address, rules): 지정한 가입자의 호에 대하여 호 차단 또는 허용, 조건부 전환, 무조건 전환, 안내방송 송출 등의 처리 규칙을 설정
- **setRulesForGroup** (address list, rules): 지정한 가입자들의 호에 대한 처리 규칙 설정

- `getRules (address)`: 지정한 가입자의 규칙 검색
- `clearRules (address list)`: 지정한 가입자들에 대한 규칙 삭제

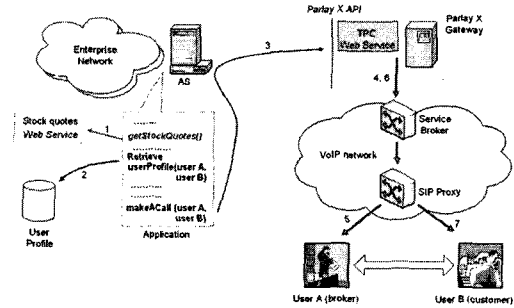
5.1.4 Multimedia Conference

응용 프로그램에서 통신망의 특정 가입자들을 대상으로 멀티미디어 회의를 생성하고 참가자를 제어할 수 있도록 하는 기능으로서 다음과 같은 API를 제공한다.

- `createConference (type, description, duration, participants max #, owner)`: 회의 생성
- `getConferenceInfo (confID)`: 생성된 회의 정보 검색
- `endConference (confID)`: 생성된 회의 종료
- `inviteParticipant (confID, participant)`: 지정된 참석자를 회의에 초대
- `disconnectParticipant (confID, participant)`: 지정된 참석자에 대한 연결 해제
- `addMediaForParticipant (confID, participant, media)`: 특정 참석자에게 미디어 스트림 추가
- `deleteMediaFroParticipant(confID, participant, media)`: 특정 참석자의 미디어 스트림 제거

Call Control 관련 API는 인터넷 서비스 개발자들이 응용 프로그램을 개발할 때, 통신망의 호 제어 메커니즘을 상세히 알고 있지 않더라도 호를 제어하는 프로그램 개발을 용이하게 한다. 걸려오는 전화를 선택적으로 받거나, 차단하고, 또한 전환하는 등의 서비스를 받기 위해 통신 사업자 별도로 서비스를 신청하는 번거로움이 없이 Open API를 이용한 개방형 서비스를 통해 개인 맞춤형의 호 제어가 가능하게 되는 것이다.

예를 들어, 증권 회사의 고객 관리 응용 서비스의 한 예로, 고객이 관심을 갖고 있는 종목의 주가가 사전에 등록된 임계치를 넘었을 경우에, 그 처리를 어떻게 할 것인가를 고객과 논의하기 위해, 고객과 증권 회사 직원 간의 전화 통화를 연결시키는 기능이 필요한 경우, 응용 프로그램에서 `makeCall` (고객 전화번호, 직원 전화 번호) API를 호출함으로써 고객과 증권회사 직원 간에 통신망을 통



(그림 8) Third Party Call Control 시나리오

한 전화 연결이 되도록 할 수 있다.

5.2 Messaging 관련

응용 프로그램에서 통신망의 가입자에게 메시지를 전송하거나, 망에서 전송된 메시지를 수신하는 제어를 할 수 있도록 다음과 같은 3가지 그룹의 기능들이 웹서비스로 제공된다.

5.2.1 Short Messaging

응용 프로그램에서 통신망의 Short Message Service를 제어할 수 있는 기능으로서 다음과 같은 API를 제공한다.

- `SendSms (Address, SenderName, Message)`: 지정한 단말에 메시지를 전송
- `GetSmsDeliveryStatus (requestID)`: 메시지 전송 상태 검색
- `NotifySmsReception (RegistrationIdentifier)`: 메시지 수신 통보
- `GetReceivedSms (RegistrationIdentifier)`: 통신망에 저장된 메시지를 가져오기

5.2.2 Multimedia Messaging

응용 프로그램에서 통신망의 Multimedia Message Service를 제어할 수 있게 하는 기능으로서 다음과 같은 API를 제공한다.

- `SendMessage (Address, SenderAddress, Subject, priority)`: 지정한 단말에 메시지를 전송
- `GetMessageDeliveryStatus (DeliveryStatus)`:

메시지 전송 상태 검색

- **GetReceivedMessages** (RegistrationIdentifier, Priority): 통신망에 저장된 메시지를 가져오기
- **NotifyMessageReception** (MessageRefIdentifier, Message): 메시지 수신 통보

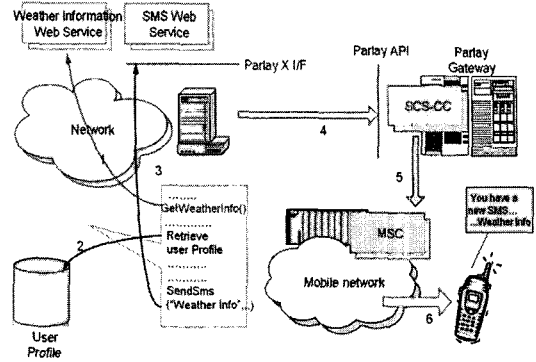
5.2.3 Audio Call 기능

응용 프로그램에서 통신망 가입자의 단말에 음성 메시지 등을 전송할 수 있도록 하는 기능으로서 다음과 같은 API를 제공한다.

- **playTextMessage** (Address, Text, Language): 지정한 단말에 텍스트 메시지를 음성으로 변환하여 전송
- **playAudioMessage** (Address, AudioUrl): 지정한 단말에 오디오 파일을 전송
- **playVoiceXmlMessage** (Address, VoiceXmlUrl): 지정한 VoiceXml 콘텐츠를 전송
- **getMessageStatus** (Correlator): 메시지 상태 정보 획득
- **endMessage** (Correlator): 메시지 전송 종료

SMS는 기본적으로 text 기반의 단문을 전송하는 서비스인 반면 MMS는 text는 물론, 그래픽, 이미지, 오디오, 비디오를 포함한 멀티미디어 메시지를 전송할 수 있다. 통신망에 대해 전혀 지식이 없고, 메시지가 어떻게 망에서 전달되는지 상세 기술을 전혀 몰라도 어플리케이션과 결합한 다양한 서비스를 개발할 수 있다. 예를 들어 뉴스 콘텐츠를 web을 통해 제공하는 서비스 포털의 경우, 가입자 정보 중 단말 주소를 안다면 MMS API를 통하여 아주 쉽게 가입자에게 뉴스 콘텐츠를 보낼 수가 있다.

Audio Call API는 통신망 가입자의 단말에 음성 메시지나 음악 등의 정보를 보낼 수 있는 점이 차별화 되며, 연결된 호를 통해 정보의 수집도 가능하다. 예를 들어 가족이나 친구의 생일을 일정 관리 프로그램에 등록해 두고, **PlayAudioMessage** ()를 호출하여 축하 음악을 통신 단말로 전송할 수 있으며, e-mail을 확인할 수 없는 상황에서 중요한 메일 수신 시 **PlayTextMessage** ()를 이용하여 메일의 내용을 음성으로 변환하여 단말로 보내줄 수



(그림 9) Send SMS 시나리오

있다. 설문조사 기관 등에서 수행하던 전화 설문이나 전화 투표 서비스도 설문 내용을 VoiceXML 콘텐츠로 제공하면 오퍼레이터의 도움 없이 전화 설문이나 전화 투표 서비스가 가능하다.

5.3 Mobility 관련

응용 프로그램에서 통신망 가입자 단말의 이동성을 제어할 수 있도록 다음과 같은 2가지 그룹의 기능들이 웹서비스로 제공된다.

5.3.1 Terminal Status

응용 프로그램에서 통신망의 단말 상태를 알 수 있게 하는 기능으로서 다음과 같은 API를 제공한다.

- **getStatus** (Addresses): 지정한 단말의 상태 정보 요청 (이때 제공되는 이동단말의 상태 정보는 Reachable, Not Reachable, Busy 등으로 표현된다.)
- **getStatusForGroup** (Addresses): 지정한 단말들의 상태 정보들 요청
- **startNotification** (Reference, Addresses, Criteria): 지정한 단말의 상태가 변경되는 경우에 이 사실을 응용 프로그램으로 통보할 것을 요청
- **endNotification** (Correlator): 통보 요청 해지
- **statusNotification** (Correlator, Address, CurrentStatus): 지정된 단말의 상태 변경 통보

5.3.2 Terminal Location

응용 프로그램에서 통신망의 단말 위치 정보를 제어할 수 있게 하는 기능으로서 다음과 같은 API를 제공한다.

- getLocation (Address): 지정한 단말의 위치 및 정보를 요청 (이때 제공되는 이동단말의 위치 정보는 위도, 경도, 고도 등으로 표현된다)
- getTerminalDistance (Address, Latitude, Longitude): 특정 위치에 대한 거리 정보 요청
- getLocationForGroup (Addresses): 지정한 단말들의 위치 정보를 요청
- startGeographicalNotification (Reference, Addresses, Latitude, Radius, Criteria): 지정한 단말이 특정 영역을 벗어나거나 진입하는 경우, 이를 통보해주시기를 요청
- startperiodicNotification (Reference,Addresses): 주기적인 단말 위치 정보 요청
- endNotification (Correlator): 단말 위치 정보 요청 해지
- locationNotification (Correlator, Data, Criteria): 요청된 단말의 위치 정보 통보

응용서비스에서는 이러한 Mobility 관련 API 들을 사용하여 이동통신망에 대한 상세 지식 없이도 단말의 정보를 얻기 위한 여러 동작을 요청할 수 있다. 예를 들어, 사전에 지정한 대상이 서비스 요청자의 위치 정보를 기반으로 하여 일정 거리 영역 내에 들어오거나 나갈 때, 그 사실을 서비스 신청자에게 통보해 주며, 사용자 근처의 교통 사항, 할인 행사 정보, 이벤트 정보 등을 알려주는 서비스도 가능해진다. 또한, 학교를 다니는 자녀가 지정된 반경 영역을 벗어나게 되면 이를 보호자에게 알려 주어 사고 유무를 확인할 수 있도록 할 수 있다.

5.4 Charging 관련

응용 프로그램에서 통신망 사업자를 통한 지불 및 계좌 관리 기능을 제어할 수 있도록 하기 위해 다음과 같은 2가지 그룹의 기능들이 웹서비스로 제공된다.

5.4.1 Payment

신뢰성 있는 통신망 사업자가 지불을 대행하게 하는 기능으로서 다음과 같은 API를 제공한다.

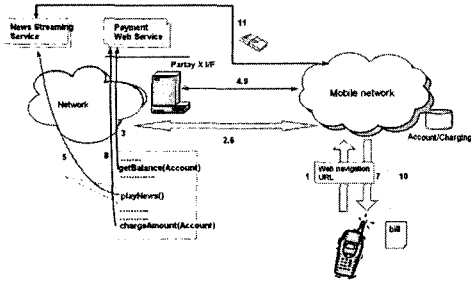
- ChargeAmount (endUserIdentifier, Amount): 지정한 계정에서 지정 금액을 지불
- RefundAmount (endUserIdentifier, Amount): 지정한 계정에 지정 금액을 환불
- ReserveAmount (endUserIdentifier, Amount): 지정한 계정에서 지정 금액을 가지불
- ChargeReservation (reservationIdentifier, Amount): 지정한 계정에서 가지불된 금액을 정산
- ReleaseReservation (reservationIdentifier): 지정한 계정에 가지불된 금액을 환불

5.4.2 Account Management

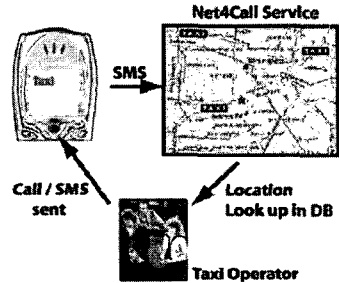
통신망 사업자가 관리하고 있는 선불 카드 등의 계좌에 대한 제어를 할 수 있게 하는 기능으로서 다음과 같은 API를 제공한다.

- GetBalance (EndUserIdentifier, EndUserPin): 지정한 계정의 현재 상태 요청
- GetCreditExpiryDate (EndUserIdentifier, EndUserPin): 유효 기간 조회
- BalanceUpdate (EndUserIdentifier, EndUserPin, Amount): 계정에 지정한 금액 입금
- VoucherUpdate (EndUserIdentifier,EndUserPin, VoucherIdentifier,VoucherPin): 지정한 계정에 상품권 상당의 금액 입금
- GetHistory (EndUserIdentifier, EndUserPin, Date, MaxEntries): 거래 내역 조회

소규모 전자 상거래를 하는 인터넷 몰을 운영하는 사업자의 경우, 결제 방식을 자체적으로 구축하지 않고 결제 대행업체에 의뢰하여 결제 기능을 수행하고 있음으로 인해, 거래상의 불안함과 개인 정보의 노출에 대한 위험 등이 전자 상거래 활성화의 저해 요인으로 작용하고 있다. Payment API 및 Account Management API는 신뢰성 있는 통신 사업자가 전자상거래의 결제 기능을 대행할 수 있도록 하여, 인터넷 상거래 관련 사업자들이 별도의



(그림 10) Payment 시나리오



(그림 11) Open API 응용 서비스 - Call by Name

인증 채널을 가지는 신용카드 결제 업체나 휴대폰 결제 대행 업체의 협력 없이도 통신 사업자를 통한 안전한 과금 회수 대행을 할 수 있도록 한다.

예를 들어, 통신 사업자가 판매하는 선불카드를 구입하여, 각 포털 사이트에서 운영하는 개별적인 화폐 단위를 구매할 필요 없이, 선불카드 하나로 물건을 구매하거나 영화, 게임 등을 다운로드 받을 수 있게 된다. 포털 사이트의 입장에서는 자체적인 신용카드 결제 등 과금 기능을 보유하지 않더라도 신뢰성 있는 통신망 사업자를 통하여 안전하게 결제를 할 수 있는 장점이 있게 된다.

5.5 Presence 관련

응용 프로그램에서 통신망 가입자의 context에 기반한 정보를 제어할 수 있도록 다음과 같은 기능이 웹서비스로 제공된다.

5.5.1 Presence

유무선 통합망에 대한 사용자의 다양한 상태 정보를 관리할 수 있게 하는 기능으로서 다음과 같은 API를 제공한다.

- **subscribePresence (Presentity, Attributes, Application):** 지정한 가입자의 프레즌스 정보 이용 요청 (프레즌스 정보로는 활동 상태, 장소, 프라이버시, 단말, 기타 정보 등이 있음)
- **notifySubscription (Presentity, Decisions):** 등록 요청을 받은 상대방의 승인 여부를 통보
- **publish (Presence):** 사용자(presentity)가 자신의 프레즌스 정보를 등록

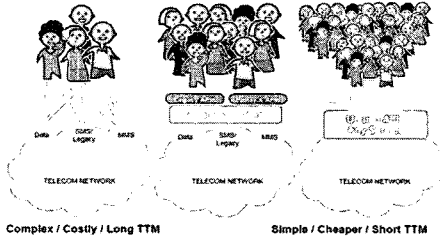
- **getUserPresence (Presentity, Attributes):** 지정한 가입자의 프레즌스 정보 조회
- **startPresenceNotification (Presentity, Attributes):** 프레즌스 정보의 주기적 통보 요청
- **statusChanged (Correlator, Presentity, ChangedAttributes):** 변경된 프레즌스 정보 통보
- **endPresenceNotification (Correlator):** 프레즌스 정보의 주기적 통보 요청을 해지

Presence API는 통신망 가입자의 상황 및 Preference를 제어할 수 있는 기능을 제공한다. 어느 곳에서 무엇을 어떤 상태로 하고 있는지, 어떤 통신 기기의 사용이 가능하지 등을 제어 할 수 있으며, 향후 context-aware 통신 환경의 발전과 더불어 유비쿼터스 서비스의 기반을 제공할 수 있을 것이라 예측되고 있다.

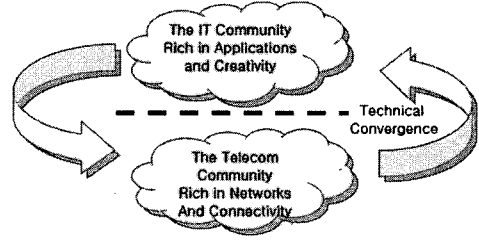
6. 개방형 서비스 사례

통신망의 기능들이 웹서비스 기반의 Open API로 개방됨으로 인해 통신 서비스 개발은 이러한 API를 조합하는 것으로 단순화 될 수 있으며, 엔터프라이즈 도메인에서 개발된 웹 서비스들과 조합하여 다양한 신규 서비스의 개발이 가능하게 된다.

더 나아가 IT839의 센서네트워크, 홈네트워크, 텔레매틱스 등의 기능들이 웹서비스로 공개되면, 이 서비스들의 연동과 융합을 통해 새로운 복합 애플리케이션의 창출을 가능하게 할 수 있다. 레고 블럭을 이용하여 자신만의 고유한 작품을 만들 수 있듯이, 누구든 아이디어만 있으면 새로운 유비쿼터스 서비스의 개발이 가능하게 되는 것이다.



(그림 12) Open API 발전 방향



(그림 13) 정보 기술과 통신 기술의 융합

Open API를 이용한 응용 서비스의 개발 사례로서 그림 11의 Call-by-Name 서비스는 통신망의 메시지 기능 및 호 제어 기능을 사업자의 데이터베이스와 결합한 서비스 유형을 제시한다.

예를 들어 택시가 필요한데 근처에 택시가 보이지 않는 경우, 서비스 사업자로 “택시”라는 SMS 문자를 보내면, 서비스 사업자의 응용 서버는 Terminal Location API의 GetLocation()을 통해서 발신자의 위치를 파악하고, 사업자의 데이터베이스에 등록된 택시 사업자들 중 요청자의 위치에 가까운 택시 사업자를 찾아내어 발신자의 전화번호와 위치 정보를 MMS를 통해 알려주어 택시를 보낼 수 있도록 하거나, 또는 직접 통화할 수 있도록 TPC 기능을 이용하여 서로 간에 호를 연결해 줄 수도 있다. 이러한 서비스는 택시뿐 아니라 일반적인 상호 검색 등의 디렉토리 서비스로 확장이 가능하며, 지역 정보 검색 등 텔레매틱스 서비스 개발에도 적용이 가능하다.

7. 개발 동향

터키의 이동통신망 사업자 TELSİM에 의해 처음으로 Open API 기술을 적용한 Pre-paid 서비스가 2001년 6월 상용화 된 이래, 유럽의 이동통신망 사업자들을 중심으로 Open API가 도입되고 있다. 기존의 사업자들은 지능망 서비스의 보완 및 진화 단계로서, 신규 사업자들은 다양한 서비스의 신속한 도입을 위해 개방형 서비스 구조를 채택하고 있다. 미국의 SPRINT는 IBM과 협력하여 기업 고객을 목표로 개방형 서비스 플랫폼을 구축하고 있으며, 영국의 BT는 미래의 통신망에서의 수익은

개인 중심의 인터넷 기반 멀티미디어 서비스로 인해 발생할 것이라고 전망하고 Open API 기술 개발에 투자를 하고 있다.

국내에서는 KT, 데이콤, LGT, SKT 등 유무선 통신망 사업자를 중심으로 새로운 형태의 비즈니스 모델을 창출하기 위해 Open API 기반의 통신망 구축이 추진되고 있으며, 유엔젤, 인프라밸리, 헤리트 등을 중심으로 Open API 기반의 게이트웨이 및 응용 서버 개발이 진행되고 있다. 국책 과제로서 2004년부터 ETRI를 중심으로 6개 통신망 사업자 및 4개 장비 개발업체와 협력하여 광대역 통합망에 대한 웹서비스 기반의 Open API의 발굴 및 국내의 표준화를 추진하고 있으며, 유무선 통합 서비스 및 통신·방송 융합 서비스를 위한 웹서비스 기반의 개방형 서비스 게이트웨이를 개발하고 있다. 또한 개방형 서비스 기술 인력 양성 및 새로운 서비스 발굴을 위해 정부 주관으로 개방형 서비스 센터 구축이 추진되고 있다.

8. 결 론

통신망의 하부 구조에 독립적으로 통신 서비스를 정의하고, 개발할 수 있도록 유무선 통신망의 기능들이 표준화된 인터페이스를 통해 외부로 공개되고 있다. 이러한 Open API는 정보 기술과 통신 기술을 통합하는 가교 역할을 함으로써 새로운 융합형 통신 서비스의 개발을 가능하게 하는 기술로 대두되고 있으며, 이는 서비스 지향 구조를 구현하는 웹서비스 기술로 현실화되고 있다.

BcN에서 추진되고 있는 이러한 웹서비스 기반

의 개방형 서비스 기술은 IT 839 서비스들과 융합되어 유비쿼터스 서비스의 기반을 조성하게 되며, 신규 비즈니스 모델을 발굴 및 새로운 시장을 창출하는 기회를 제공할 수 있을 것으로 기대 된다. 이를 위해서는 개방형 서비스 기반의 차세대 통신망 인프라 구축에 통신 사업자들의 관심과 투자가 절대적으로 필요하다. 인터넷의 본질이 정보와 지식의 공유에 있다면, 유비쿼터스의 본질은 u헬스, u환경, u교통 등 유비쿼터스 서비스의 실제적인 구현을 통한 인간의 삶의 질 향상에 있기 때문이다.

참 고 문 헌

- [1] 김은주, “다시 주목받는 웹서비스”, 디지털 타임즈, 2005.5.23.
- [2] 김인, “유비쿼터스 시대와 IT 서비스”, 전자신문, 2005.5.23.
- [3] ETSI, “Open Service Access (OSA): Parlay X Web Services”, ETSI 202 391-14 V1, Mar. 2005.
- [4] 3GPP, “3rd Generation Partnership Project; Technical Specification Group Core Network; Open Service Access (OSA); Parlay X Web Services; Release 6”, 3GPP TS 29.199 V6.2, June 2005.
- [5] Parlay Group. (<http://www.parlay.org>), White Papers and Meeting Presentations.
- [6] Ard-Jan Moderdijk, “Opening the Networks with Parlay/OSA API: Standards and Aspects behind the APIs”, IEEE Network, pp. 58 ~64, May/June 2003.
- [7] Zygmunt Lozinski, “Parlay/OSA - a New Way to Create Wireless Services”, IEC Mobile Wireless Data, pp.61-70. June 2003
- [8] 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.
- [9] 정기숙, 이용주, 임종철, 임선환, 박유미, 김형환, “융합형 통신 서비스를 위한 Open API 기술 - 특집 연재”, 온더넷, 2005년2월호 ~ 2005년 7월호.
- [10] 박유미, 최영일, 이병선, “웹 서비스 기반의 개방형 서비스 기술”, 통신학회, vol. 22, No. 5, pp. 28-42, May, 2005.

◎ 저 자 소 개 ◎



최 영 일

1983년 서울대학교 전자공학과 졸업 (공학 학사)
 1998년 충남대학교 컴퓨터과학과 졸업 (이학 석사)
 2002년 충남대학교 컴퓨터과학과 졸업 (이학 박사)
 1996년 정보통신 기술사
 1984년~1986년 AT&T Bell 연구소 객원연구원
 1983년~현재 한국전자통신연구원, 개방형서비스플랫폼팀 팀장
 2002년~현재 표준화 전문위원 (MSF, Parlay Group)
 관심분야 : 차세대 네트워크, Softswitch, 개방형 서비스 기술