

# 이동통신 자료 동기화를 이용한 외근 사원 일정관리

## Schedule Management for Outside Duty Personnels Using Mobile Communication Data Synchronization

장 대 진 (Dae Jin Jang)	계명대학교 컴퓨터공학과 박사과정
박 기 현 (Kee Hyun Park)	계명대학교 컴퓨터공학부 교수
주 흥 택 (Hong Taek Ju)	계명대학교 컴퓨터공학부 조교수
유 상 진 (Sang Jin Yoo)	계명대학교 경영정보학과 교수

### 요 약

외근 사원들은 사무실 보다는 밖에서 주로 개인적으로 근무하는 경우가 많기 때문에, 개인 및 부서 전체의 일정관리가 대단히 어렵다. 개인이나 부서 전체의 일정을 변경할 필요가 있을 때마다, 일일이 모두에게 전화를 해서 이를 알려야 한다. 그런데, 외근 사원들이 가지고 다니는 PDA 등과 같은 이동통신 기기를 이용해서 필요할 때마다 가장 최근의 부서 일정을 다운로드하거나 본인의 일정을 업로드 할 수 있다면, 외근 사원들의 개인 및 부서 전체의 일정관리 문제를 쉽게 해결할 수 있다.

본 논문에서는 이동통신 환경에서의 자료동기화 기법을 이용하여 외근 사원들을 위한 일정관리 시스템을 설계하고 구현하였다. 이동통신 환경에서의 자료동기화란 이동통신 단말기에 저장된 자료와 중앙 관리 서버의 자료를 일치시키는 작업을 일컫는다. 즉, OMA(Open Mobile Alliance)에서 표준으로 제안하고 있는 SyncML 자료동기화 방법을 이용하여 언제 어디서나 외근 사원들이 부서 전체의 일정을 점검하고 본인의 일정을 부서 전체에 알릴 수 있는 일정관리 시스템을 설계하고 구현하였다.

**키워드 :** 일정관리, 자료동기화, SyncML, OMA

## I. 서 론

휴대형 단말기에 대한 정의는 조사 기관 및 조사하는 사람에 따라 다양하게 나타나고 있으나, 제한된 메모리와 디스플레이를 가지면서 손에 들고 다니며 이용할 수 있는 단말기를 통칭한다. 휴대형 단말기의 예로는 핸드폰, PDA 등

을 들 수 있으며, 좀 더 광의의 개념으로 확장하면 태블릿(Tablet) PC, 핸드헬드 PC 등도 포함될 수 있다. 지금까지 휴대형 단말기로 가장 많이 사용되어 왔던 것은 휴대폰이었다. 현재 휴대폰은 가장 편리한 통신기기로써 우리의 생활양식을 변화시키면서 일상생활의 필수품으로 자리 잡고 있다(권지인, 2004년; 위피(WIFI); 장대진 등, 2005).

또한, 무선 네트워크 기반 기술의 발달과 서비스의 다양화로 인해, 휴대형 단말기는 일정관

† 본 연구는 산업자원부의 지역혁신 인력양성사업의 연구결과로 수행되었음.

리나 전자메일 교환과 같은 개인적인 용도에서 기업 운영이나 업무의 효율성을 향상시키는 비즈니스 영역에 대한 새로운 수단으로 그 역할이 확대되고 있다. 이와 더불어, 휴대형 단말기는 기업의 생산성을 높이기 위한 도구로 발전하면서 금융, 제조, 물류 및 교육 분야 등 다양한 사업 분야에 적용됨으로써, 포스트 PC 시장을 선도할 품목으로 많은 관심이 집중되고 있다(한국 전자통신 연구원기술평가팀, 2005).

그런데 이동무선통신에서는 기존의 유선 인터넷 기반과 달리, 연결 비용에 따른 효율성 및 무선통신환경의 서비스 범위 한계로 인하여 중앙 관리 서버의 자료를 항상 이용할 수 없는 실정이지만, 사용자는 장소나 시간에 구애받지 않고 최신 자료의 이용을 원한다. 휴대형 단말기의 사용자가 인터넷에 연결된 서버의 자료를 이용하기 위해서는 우선적으로 무선 인터넷 통신을 통하여, 해당 자료를 검색한 후, 이를 대상 단말기에 저장하는 방식으로 동작하고 있다. 단말기에 저장된 자료를 사용하는 동안에 중앙 관리 서버와의 연결이 해제되기 때문에, 연결 해제 시간동안 변경된 자료를 다시 무선 인터넷 통신을 이용하여 중앙 관리 서버에 전송하여야 한다. 또한, 중앙 관리 서버에서 해당 자료를 추가, 변경 및 삭제와 같은 동작이 발생할 수 있으므로, 단말기의 저장된 자료가 항상 최신성을 가진다는 점에 신뢰성을 보장할 수 없다. 즉, 휴대형 단말기의 사용자가 항상 최신의 자료를 이용하기 위해서는, 중앙 관리 서버의 자료와 단말기의 자료를 일치시키는 작업이 필요하며, 이를 이동무선 통신에서의 자료 동기화(Data Synchronization)라고 한다(Ren and Song, 2002; Butrico et al., 2000; Agarwal et al., 2002).

유 · 무선 네트워크에서 동일한 자료가 중앙 관리 서버 및 다수의 휴대형 단말기에 각각 저장되어 있는 경우에, 한 지점에서 동일한 자료에 대한 변경 작업을 수행한다면 해당 서버 및 각각의 단말기의 자료도 신속하게 변경된 사실이

통보되어야 하는 요구가 최근에 급격히 늘어나고 있다. 예를 들면, 특정 회사의 한 부서 직원들 모두가 동일한 일정 자료를 자신의 휴대형 단말기에 저장한 후, 사무실 바깥에서 근무를 시작한 후, 회사 내부 사정에 의해 소속된 부서의 일정이 변경되거나, 해당 고객에 대한 정보가 변경될 경우, 서로 다른 일정 자료를 자신의 단말기에 저장할 우려가 있다. 이러한 경우 해당 부서의 직원들에 대한 전체적인 일정 자료에 대해서 일치성 문제가 발생하여 혼란이 생길 수 있다. 따라서 각자의 일정 자료에 대한 변경이 일어날 때 마다, 다른 서버 혹은 단말기들에 각각 중복 저장되어 있는 동일한 자료에 대한 동기화 작업이 필요하다. 재택근무자의 증가 및 직원들의 영업 분야에서 휴대형 단말기의 사용이 급증하는 추세로 볼 때, 자료의 일치성 보장에 대한 문제 해결 방안을 고려해 볼 수 있다. <그림 1>은 자료 동기화 방식을 이용한 일정 관리의 예를 보여주고 있다(Ren and Song, 2002; Butrico et al., 2000).

다음 <그림 1>에서 알 수 있듯이, 한 기업의 외근 직원들이 오전에 회사에 출근한 뒤, 각자의 오늘 업무 및 부서 전체의 일정을 각 개인의 단말기에 저장한 후, 외근업무를 위해 회사 밖으로 나가게 된다. 그 후, 회사 내에서 부서 전체의 일정이 변경된다. 변경된 부서 업무 일정은 기업 중앙 서버에 저장되고, 각 외근 직원들의 단말기들과 동기화 과정을 수행한 후, 각 외근 직원들의 단말기에는 변경된 최신의 일정들이 저장된다. 이로 인해서, 각 외근 직원들은 변경된 일정에 대해서 메일이나 전화 통화와 같은 주기적인 확인 절차를 겪지 않고서도 변경된 업무 일정에 맞춰 원활한 근무가 가능해진다.

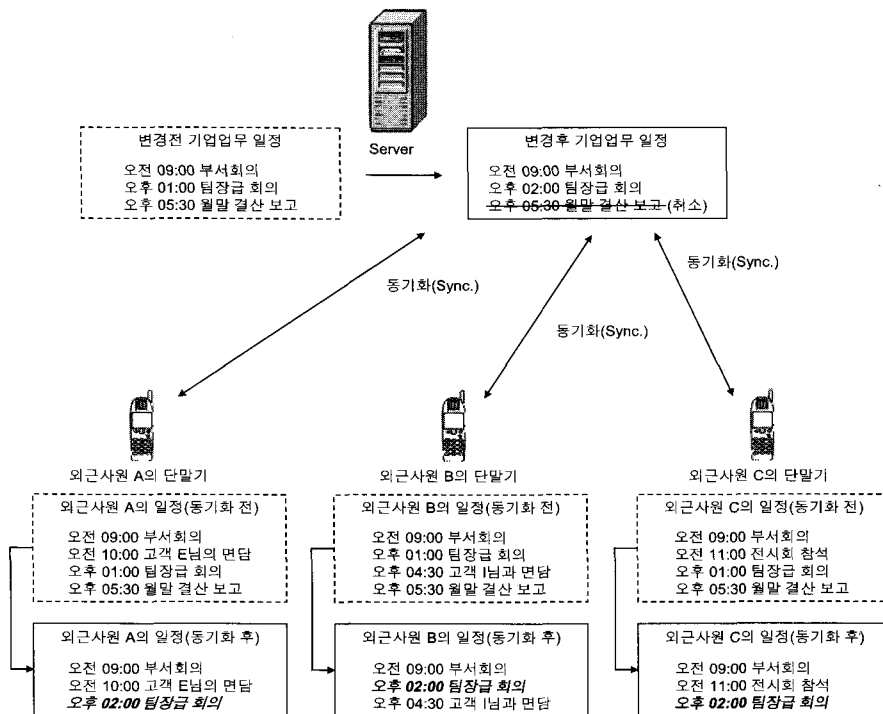
따라서, 이러한 문제를 해결할 수 있는 방안 중 하나로서, 앞에서 언급한 자료 동기화 방법을 생각해 볼 수 있다. 현재, 자료 동기화 문제를 해결하기 위하여 여러 회사들이 각자의 방법을 사용하고 있다. HP 회사에서는 Windows CE에서 동

작하는 ActiveSync를 사용하고(ActiveSync), Palm 회사의 경우에는 Palm OS에서 동작하는 HotSync를 사용하는 등(HotSync), 다양하고 많은 자료 동기화 프로토콜들과 방법들을 사용하고 있기 때문에 표준화에 대한 요구가 높아졌다. 따라서 2000년 2월 IBM, Lotus, Nokia, Palm, Motorola, Psion, Starfish Software 등의 이동무선통신 관련 업체들을 중심으로 한 연합체인 OMA(Open Mobile Alliance)에서 SyncML(Synchronization Markup Language)을 제안하여 공개적인 표준화를 주도하고 있다(장대진 등, 2005; Ju, 2004; SyncML Initiative, 2000; Hansmann, et al., 2002).

본 논문에서는 자료 동기화 방법의 실제 응용 분야에 적용하기 위한 연구의 일환으로써, SyncML 기반의 자료 동기화 방법을 이용하여 기업 운영의 효율성을 향상시키기 위해서 각각의 외근사원들이 소유하고 있는 휴대형 단말기의 일정 관리를 효율적으로 일치시키기 위한 솔루션을 개

발하였다. 본 논문에서 개발한 이동통신 자료 동기화를 이용한 외근 사원 일정관리 시스템 명칭은 ThinkSync 일정관리 솔루션으로 부여하였으며, 각각의 단말기에 대한 vcards(주소록), vevents(업무), notes(메모)로 구성된 PIMS(Personal Information Management System: 개인정보관리시스템) 정보 및 일정 정보에 대해서 자료 동기화를 수행함으로써, 동일한 자료에 대한 일치성을 보장할 수 있다(장대진 등, 2005).

본 논문은 다음과 같이 구성된다. II장에서는 본 논문에 대한 관련 연구를 간략히 소개한다. III장에서는 본 논문에서 개발한 ThinkSync 일정관리 솔루션의 설계에 대해서 설명하고, IV장에서는 PIMS 정보를 이용하여 휴대형 단말기에서 동작하는 일정관리 프로그램의 자료 동기화 수행 여부에 대한 적합성 결과를 제시한다. 마지막으로, V장에서는 결론에 대해서 언급한다.



〈그림 1〉 이동통신 자료 동기화를 이용한 외근사원 일정관리 예

## II. 관련 연구

전 세계적으로 이동무선통신에서의 자료 동기화 방법들은 상용화되었으나, 각 회사마다 서로 다른 자료 동기화 방법을 사용하기 때문에 호환성이 결여되었다. 따라서, 사용자가 인터넷 서버의 자료 동기화 방법과 동일한 방법을 사용하는 이동무선통신 단말기만을 구입해야 하거나, 혹은 인터넷 서버 관리자가 모든 가능한 자료 동기화 방법들을 지원해야 하므로, 경제적 손실과 함께 불편함이 있다. 이 장에서는, 각 회사마다 다르게 사용하는 대표적인 자료 동기화 방법들을 살펴본다.

### 2.1 SyncML

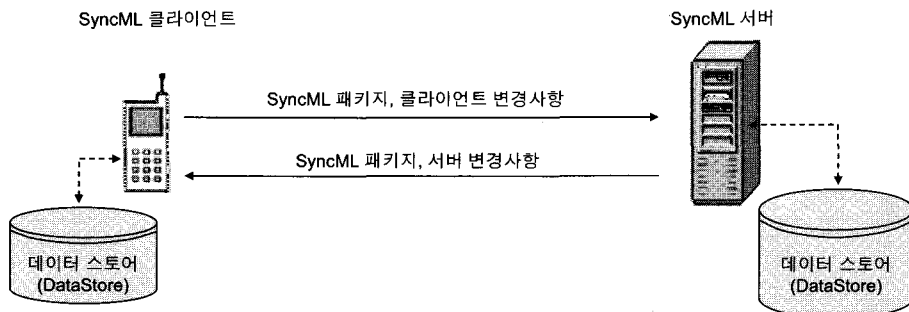
SyncML은 각 제조회사의 단말기가 가지고 있는 서로 다른 플랫폼, 통신 프로토콜, 자료 형태, 응용 서비스에 이용될 수 있는 자료 동기화 방식과 장치 관리에 대한 개방형 표준 인터페이스 개발을 목적으로 한다. <그림 2>는 SyncML을 기반으로 하는 클라이언트와 서버간의 자료 동기화를 나타내고 있다(SyncML Initiative, 2000; Hansmann *et al.*, 2002).

SyncML 클라이언트가 먼저 서버에게 갱신된 자료를 포함한 메시지를 전송하면, SyncML 서버는 클라이언트가 요청한 동기화 타입에 의해서 서버 측 자료와 동기화 작업을 수행한 후, 다시

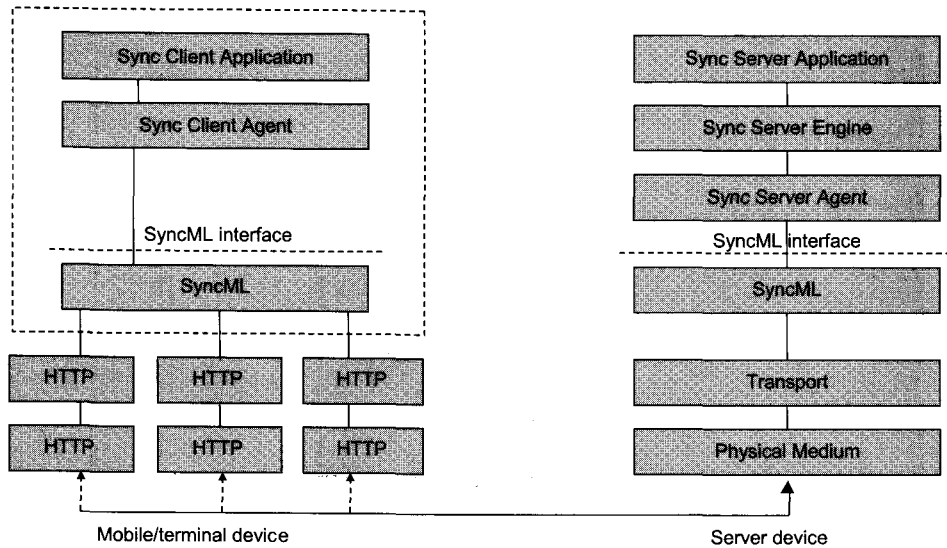
클라이언트에게 작업 결과 및 서버 자신의 변경된 자료도 전송한다. 이러한 몇 번의 메시지 교환을 수행함으로써, 자료 동기화가 이루어지며 클라이언트와 서버간의 해당 자료에 대하여 일치성을 보장하게 된다.

SyncML은 이동무선통신 단말기와 서버간의 원격 동기화를 목표로 설계되었으나, 로컬 동기화와 유선 네트워크로 연결된 장치들 간의 동기화에도 사용될 수 있다. SyncML을 기반으로 하는 클라이언트와 서버간에 동기화가 수행되는 동안, 동기화 대상이 되는 객체는 논리적으로 패키지를 교환한다. SyncML 프로토콜의 구조는 <그림 3>과 같다.

Sync Server Engine은 기존의 서비스와 추가되는 서비스간의 독립 및 종속 관계를 이용하여 새로운 서비스의 추가를 신속하게 하며, 클라이언트(이동무선통신 단말기)에서 전송된 자료와 서버의 자료간의 충돌(conflict)이 발생했을 때, 이를 해결한다. 또한, Basic 혹은 MD5 등을 이용하여 클라이언트 사용자에게 대한 인증 절차를 수행하고, 인코딩/디코딩 등의 부가적인 서비스를 제공한다. Sync Server Agent는 Server Application에 독립적인 부분을 처리하는데, 클라이언트에 전송된 메시지를 처리하여 자료 동기화를 위한 관련 모듈들을 호출하고, 클라이언트의 메시지에 대한 응답을 생성하기 위한 명령을 생성하는 관련 모듈들을 호출한다(SyncML Initiative, 2002).



<그림 2> SyncML을 기반으로 하는 클라이언트-서버 구조



〈그림 3〉 SyncML 프로토콜 구조

### 2.1.1 SyncML 규격

SyncML 자료 동기화 규격은 XML 기반의 자료 표현(Data Representation) 프로토콜, SyncML 동기화(Synchronization) 프로토콜 그리고 전송 바인딩(Transport Bindings) 프로토콜로 구성되어 있다. SyncML 자료 표현 프로토콜은 자료 동기화를 위해 교환되는 SyncML 메시지의 논리적인 구조와 형태를 XML 형식으로 정의하고 있다. 각각의 필드가 어떠한 정보를 담고 있으며 해당 정보가 어떤 의미를 내포하는 것인지에 대한 약속을 정의하고 있다. SyncML은 메시지의 구조체를 정의하기 위해 세 가지의 규격을 제시하고 있다. 자료 표현 프로토콜은 SyncML의 기본 표현을 기술한 핵심 규격이며, 장치 정보 표현을 위한 SyncML 장치 정보(Device-Information) DTD 규격과 추가적인 동기화 정보와 해석을 표현하기 위한 SyncML 메타 정보(Meta-Information) DTD 규격을 포함하고 있다(SyncML Initiative, 2002; Hansmann *et al.*, 2002).

SyncML 동기화 프로토콜은 SyncML 클라이언트와 서버 간에 이루어지는 자료 표현 프로토콜 규격에 의해 생성된 자료의 추가, 삭제, 갱신

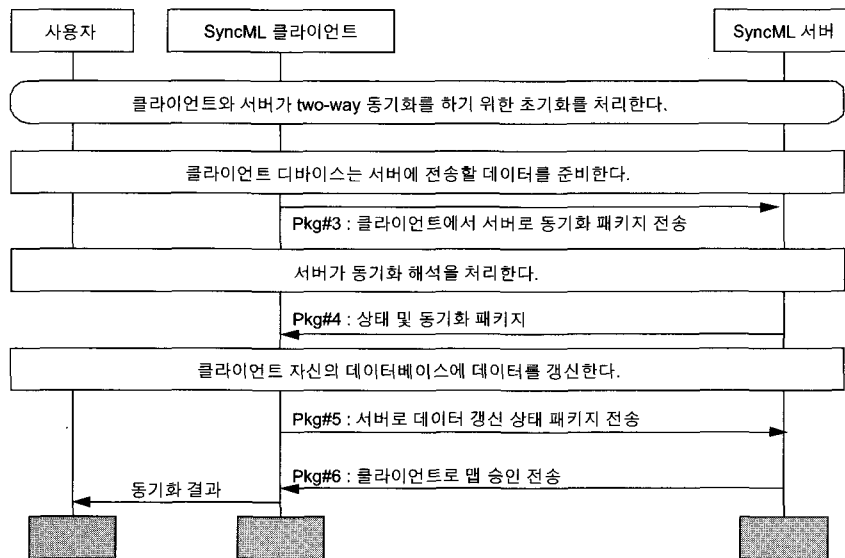
과 같은 동기화 명령과 그 밖의 상태 정보에 대한 메시지가 교환되는 방법에 대해서 정의하고 있다. 일반적으로 동기화 처리는 클라이언트가 먼저 SyncML 메시지를 서버측으로 전송하고, 서버는 클라이언트가 전송한 메시지 내의 자료와 동일한 자신의 데이터스토어에 저장되어 있는 자료 간에 동기화를 수행한 후, 응답 메시지를 보내는 일련의 과정을 통해서 이루어진다. <표 1>은 SyncML 동기화 프로토콜의 7가지 모드를 기술하고 있다(SyncML Initiative, 2002; Hansmann *et al.*, 2002).

<표 1>에서 제시한 방식 중에서 가장 많이 사용되는 Two-way Synchronization 프로토콜은 <그림 4>와 같이 진행된다.

SyncML이 동기화 메시지를 전송하기 위해서 사용하는 전송 바인딩 프로토콜은 HTTP, WSP, OBEX 등과 같은 프로토콜인데, SyncML 규격에서는 이러한 전송 바인딩 프로토콜에 대해서 새로운 프로토콜을 정의하는 것이 아니라, 기존의 전송 프로토콜과 바인딩 규칙만을 정의한다. 따라서, 자료 표현 프로토콜과 동기화 프로토콜이 전송 프로토콜과 독립적으로 구성되어 있으므로

〈표 1〉 SyncML 동기화 프로토콜

동기화 모드	설 명
Two-way Synchronization	서버와 클라이언트가 변경된 자료에 대한 정보를 교환하는 가장 일반적인 프로토콜
Slow Synchronization	모든 자료 항목들에 대한 자료 동기화를 수행하는 프로토콜
One-way Synchronization from Client Only	클라이언트가 자신의 변경된 자료를 서버에게 일방적으로 전달하는 프로토콜
Refresh Synchronization from Client Only	클라이언트가 자신의 모든 자료를 서버에게 일방적으로 전달하는 프로토콜
One-way Synchronization from Server Only	서버가 자신의 변경된 자료를 클라이언트에게 일방적으로 전달하는 프로토콜
Refresh Synchronization from Server Only	서버가 자신의 모든 자료를 클라이언트에게 일방적으로 전달하는 프로토콜
Server Alerted Synchronization	서버가 클라이언트에게 지정된 프로토콜을 이용하여 자료 동기화 시작을 요청하는 프로토콜



〈그림 4〉 Two-way Synchronization 동기화 과정

향후 다른 전송 프로토콜과의 바인딩이 가능하다 (SyncML Initiative, 2000; Hansmann, et al., 2002).

## 2.2 ThinkSync 시스템

ThinkSync 시스템은 이미 본 연구팀에서 개발

한 SyncML 기반의 자료동기화 솔루션이다. 또한, ThinkSync 시스템은 SyncML 1.1.2의 규정을 준수하도록 설계 및 구현하였으며, 유선 네트워크로 연결되어 있는 인터넷 서버에 탑재된 동기화 서버(Server)와 휴대형 단말기에 탑재된 클라이언트(Client)로 구분된다. ThinkSync 자료동기

화 클라이언트 시스템은 SyncML 규격을 준수하면서 이동무선통신 환경의 특성 및 상대적으로 처리속도가 낮은 CPU, 제한된 저장 공간에서 구동하는 내장형 소프트웨어의 특성을 고려하여 구현하였다(장대진 등, 2005; Ju, 2004).

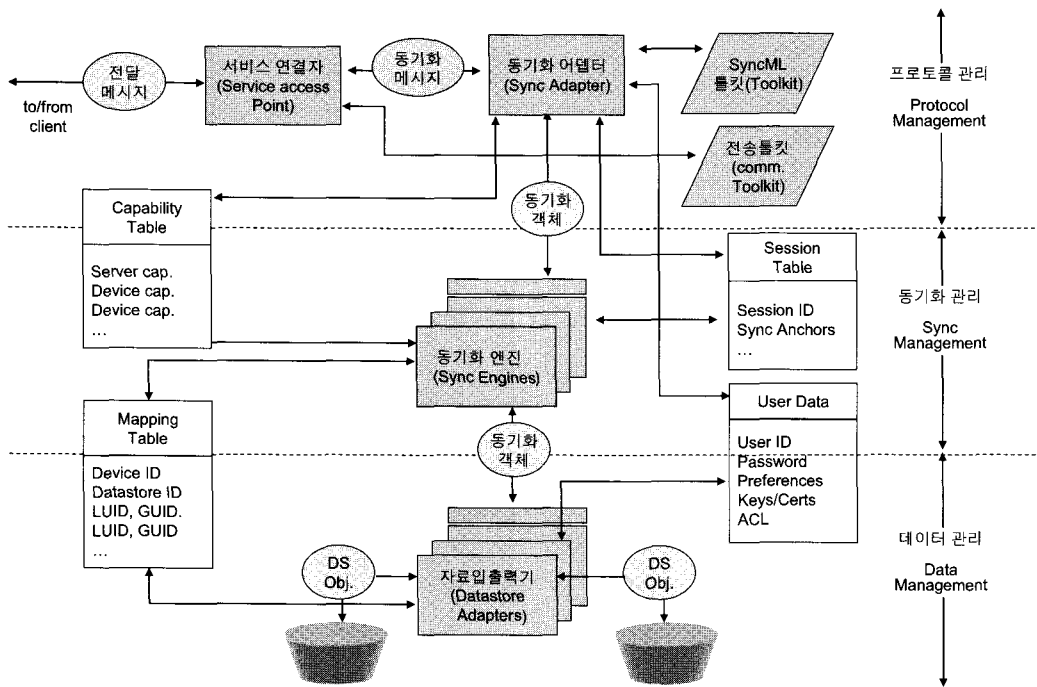
### 2.2.1 ThinkSync 서버

ThinkSync 일정관리 서버는 동기화 상대가 되는 이동 단말기부터 수신한 메시지에 대한 처리와 서버 측 변경 사항에 대한 메시지 송신 기능을 담당하며, 반영된 결과를 클라이언트에게 송신하는 기능을 갖는다. <그림 5>는 ThinkSync 자료동기화 서버의 구조를 보여주고 있다.

SyncML 기반의 동기화 서버는 프로토콜 관리, 동기화 관리 그리고 자료 관리 모듈로 구성되어 있으며, 각 모듈의 기능은 다음과 같다. 프로토콜 관리 모듈은 메시지를 처리하고 생성하며, 연속적인 자료를 교환하는 기능을 담당한다. 또한 서비스 연결자(Service Access Point), 동기

화 어댑터(Adapter), SyncML 툴킷(Toolkit), 통신 툴킷으로 구성되어 있으며, 세션 테이블, 사용자 자료, 장치 성능 테이블을 참조한다.

SyncML 기반의 동기화 서버는 다음과 같은 진행 과정을 수행한다, 우선, 인코딩된 메시지가 서비스 연결자에 도착하면 서비스 연결자는 메시지 인증 절차를 수행하고 해당 메시지를 디코딩한 후, 메시지의 결과를 동기화 어댑터로 보낸다. 이 메시지는 클라이언트의 변경 정보를 담고 있으며, 동기화 어댑터는 세션 테이블을 갱신하고 메시지를 통하여 클라이언트가 데이터베이스에 접근 권한이 있는지 판별하고, 그리고 동기화 어댑터는 메시지를 툴을 이용하여 동기화 객체로 변환하고 동기화 엔진을 호출한다. 호출된 동기화 엔진은 자료의 충돌 여부를 파악하고 해결한다. 이러한 해결 과정에서 동기화 엔진은 데이터스토어 어댑터가 사용하는 데이터 아이템을 불러와서 사용하게 될 수도 있다. 또한 동기화 엔진은 클라이언트 혹은 서버에서 아이템의 생



<그림 5> ThinkSync 자료동기화 서버

성과 변경이 발생했는지를 검사하고, 동기화 객체를 데이터스토어 어댑터가 받아서 데이터스토어 객체로 변경한 후, 데이터베이스에 저장하고 맵핑 테이블을 갱신한다.

### 2.2.2 ThinkSync 클라이언트

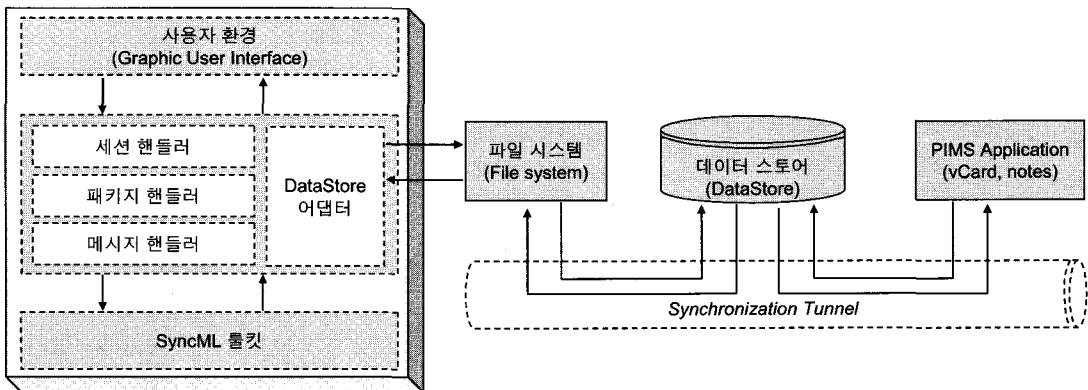
SyncML 기반 ThinkSync 자료동기화 클라이언트는 동기화 엔진, 데이터스토어 어댑터, 클라이언트 정보, 동기화 정보, SyncML 툴킷 그리고 클라이언트 응용 부분으로 구성되어 있다. 클라이언트 엔진 부분은 동기화 할 자료를 포함하여 서버에게 전송하여야 할 동기화 메시지를 생성하고, 서버로부터 수신된 메시지를 처리 및 반영한다. 클라이언트 엔진은 동기화 단위에 따라 분류된 세 개의 모듈로 구성되어 있다. <그림 6>은 SyncML 기반의 자료 동기화 클라이언트를 보여주고 있다.

ThinkSync 자료동기화 클라이언트는 광범위하게 세 부분으로 나누어지는데, 사용자가 필요한 정보를 조작하고 동기화 명령을 요청하는 GUI 부분과 실제로 동기화가 수행되는 동기화 엔진, SyncML Reference Toolkit version 4.3으로 구성되어 있다. 동기화 엔진은 다시 내부적으로 네 부분으로 나누어진다. 동기화 세션에 관여하는 세션 핸들러와 생성할 메시지를 판단하는 패키지 핸들러, 직접적으로 메시지를 생성하는 메

시지 핸들러, 다양한 종류의 자료들과 동기화 엔진을 이어주는 데이터스토어 어댑터가 있다.

GUI로 구현된 사용자 환경을 통하여 사용자가 동기화 명령을 요청하면 동기화 엔진이 작동하게 된다. 동기화 세션을 열기 위해서 세션 핸들러에서 통신을 위한 통신 세션을 RTK에서 제공하는 인터페이스를 사용하여 열고 앞으로 동기화 세션에 필요한 정보들을 초기화하게 된다. 서버와 동기화 할 준비가 완료되면 패키지 핸들러에서 생성할 메시지를 구분하고, 메시지 핸들러를 통하여 SyncML 메시지를 완성한다. 이때, SyncML 메시지가 생성되는 과정 중 서버와 동기화 할 자료가 있다면 데이터스토어 어댑터를 통하여 메시지 내에 포함되어야 한다.

세션 핸들러(Session Handler)는 실제 메시지의 송신과 수신은 SyncML Toolkit 부분이 담당하며, 툴킷 부분에서 연결된 전송 프로토콜(Transport Binding : HTTP, WAP, OBEX) 형태에 따라 선택된 프로토콜 세션을 열고 닫는다. 그리고 동기화 해야 할 메시지의 헤더 부분을 생성하는 기능을 한다. 패키지 핸들러(Package Handler)는 SyncML 문서의 Syncbody 부분의 논리적 구성을 담당한다. 논리적 구성을 위해서 메시지 핸들러에게 서버로부터 받은 메시지의 종류를 확인하고, DS Adapter에게 이전 동기화 이후의 변경사항을 확인한다. 메시지 핸들러(Message Handler)는 SyncML



<그림 6> ThinkSync 자료동기화 클라이언트 구조



클라이언트 엔진 최하부에 해당하는 부분으로 주로 이벤트 발생에 의해 호출되는 모듈이다.

동기화 정보 모듈은 동기화 대상 간에 교환되는 세션과 메시지의 참조 번호에 대한 정보를 담고 있으며, 클라이언트와 서버간의 역할과 기능을 결정하는 동기화 종류에 대한 정보를 담고 있다. 그리고 이전의 동기화 수행과 현재 수행하려 하는 동기화간의 일치성을 검사하기 위한 앵커 정보를 담고 있다. 클라이언트 응용 모듈은 실제 동기화를 요구하는 사용자와 클라이언트 엔진 간의 인터페이스를 제공하며 SyncML 툴킷은 서버로부터 수신된 메시지를 디코딩하고, 클라이언트에서 전송할 메시지를 인코딩하는 기능을 담당하며 전송 프로토콜간의 실제 메시지의 송신과 수신을 담당한다.

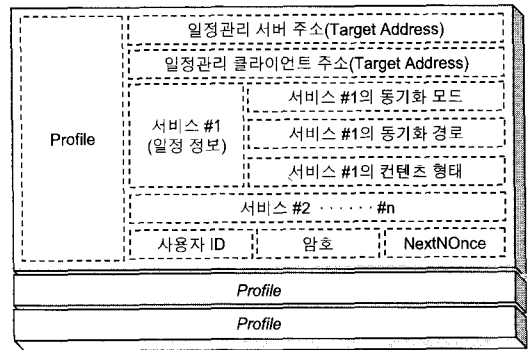
### III. ThinkSync 일정관리 솔루션 설계

ThinkSync 일정관리 솔루션은 이미 본 연구팀에서 개발한 ThinkSync 자료동기화 시스템을 기반 기술로 하여, 그 응용 분야로써 외근 사원들의 개인 및 부서 전체의 일정관리를 쉽게 해결할 수 있도록, 일정관리 PIMS 정보를 동기화 자료 대상으로 하여 본 시스템을 설계하였다. 본 절에서는, ThinkSync 일정관리 솔루션 시스템에서 일정관리 서버와 외근 사원들의 이동무선 단말기들 간의 정보 교환 대상이 되는 일정관리 정보에 대한 구조 및 이러한 정보들과 연동하기 위한 PIMS 응용프로그램에 대해서 살펴보겠다.

#### 3.1 ThinkSync 일정관리 PIMS 정보의 구조

GUI 기반의 사용자 환경 설정 프로그램은 일정관리 동기화 작업에 참여하는 ThinkSync 클라이언트 정보에 대하여 각 사용자 정보의 추가, 수정 및 삭제가 가능하다. <그림 7>은 ThinkSync

일정관리 PIMS 정보의 구조를 보여주고 있다.

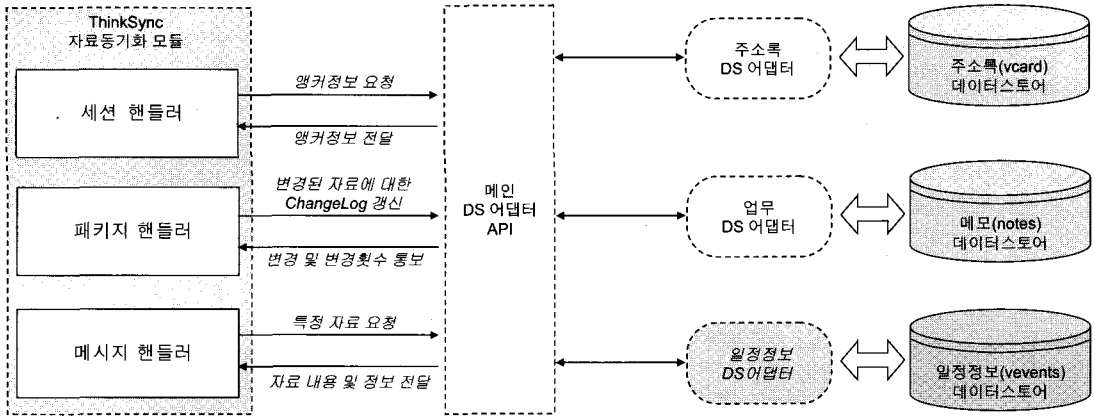


<그림 7> ThinkSync 일정관리 PIMS 정보의 구조

ThinkSync 일정관리 PIMS 정보는 다수의 각 사용자에게 대한 프로필(Profile) 정보와 동기화 대상이 되는 서비스를 가지고 있다. 목적지 주소는 동기화 대상이 되는 서버의 주소이며, 소스 주소는 동기화를 요청하는 클라이언트의 주소이다. 서비스는 동기화 대상이 되는 각각의 콘텐츠 타입들을 의미하고, 서비스 내에는 동기화를 시도하는 콘텐츠의 동기화 모드를 설정하고, 서버 측과 클라이언트 측의 동기화 경로를 가진다. 사용자 인증을 위해 사용자 ID와 패스워드, NextNonce에 대한 값이 저장되어 있으며, Base64와 MD5 방법을 이용하여 인증 절차를 수행한다.

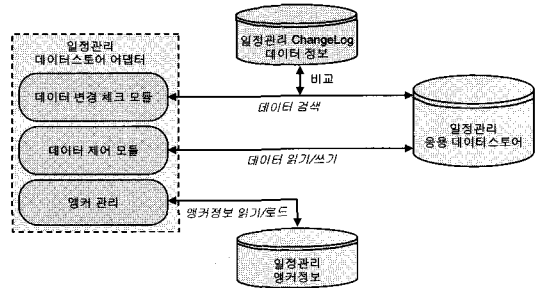
#### 3.2 ThinkSync 데이터스토어(DataStore: DS) 어댑터 구조

데이터스토어 어댑터는 다른 핸들러와 자료 사이를 연결하는 인터페이스 역할을 하며, 여러 가지 PIMS 서비스에 따라 각각의 데이터스토어 어댑터가 있다. 또한, 외부로부터 실제 데이터베이스에 접근하여 자료를 삽입, 삭제, 갱신할 수 있는 방법을 제공하며, 동기화를 위해 이전에 변경되었던 사항에 대한 로그정보를 기반으로 새로 변경된 사항들을 검사하여 동기화 할 자료를



<그림 8> ThinkSync 일정관리 클라이언트의 데이터스토어 구조

결정해준다. 본 논문에서는 이러한 여러 가지 PIMS 서비스 및 일정정보를 위한 새로운 자료 정보를 제공한다. 또한, ThinkSync 클라이언트 시스템에서 제공되는 기존의 주소록 및 메모 형태와 같은 PIMS 정보 외에도 외근 사원들의 일정관리를 위한 일정정보 PIMS 형태를 새롭게 추가하였다. <그림 8>은 ThinkSync 클라이언트 시스템이 탑재된 외근사원의 이동무선단말기에 저장된 각 서비스별 데이터스토어 어댑터 및 일정관리 정보와 자료동기화 엔진 모듈간의 구조도를 보여주고 있다.



<그림 9> 일정관리 데이터스토어 어댑터 내부 구조

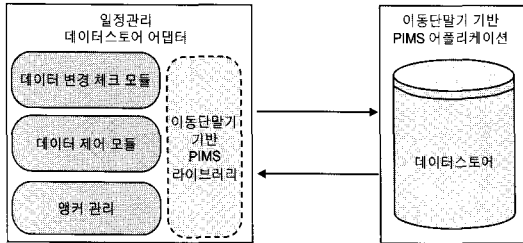
### 3.3 일정관리 데이터스토어 어댑터의 내부구조

<그림 9>는 ThinkSync 일정관리 클라이언트의 데이터스토어 어댑터에서 일정관리 부분을 구체화한 것이다. 일정관리 데이터스토어 어댑터는 내부적으로 크게 3가지 모듈로 구조화할 수 있다. 이전 동기화 이후, 클라이언트에 변경된 자료를 검사하기 위해 데이터 변경체크 모듈은 ChangeLog 파일을 리스트로 만들어 실제 데이터 객체와 비교 검사한다. 이때 변경된 사항을 새로 ChangeLog에 반영하고, 변경된 사항의 개수를 패키지 핸들러에게 알려준다.

### 3.4 일정관리 PIMS 정보 데이터스토어와 데이터스토어 어댑터의 연동

데이터스토어 어댑터는 PIMS 어플리케이션이 사용하는 데이터 스토어 접근을 위하여 데이터 제어모듈과 데이터 변경 체크 모듈에서 PIMS 라이브러리를 사용한다. 데이터 변경 체크 모듈에서는 이전 동기화 시점의 체인지 로그와 현재 데이터 스토어의 아이템 상태를 비교하여 동기화할 체인지 로그를 작성하고, 데이터 제어 모듈에서는 서버로부터 추가, 변경 및 삭제의 요청이 있을 경우 PIMS 라이브러리를 사용하여 데이터 스토어에 반영하게 된다. 앵커관리 모듈의 경우에는 앵커 정보는 데이터 스토어와 관련이 없는 동기화 시간에 관련된 정보이기 때문에 PIMS

라이브러리를 사용하지 않는다. <그림 10>은 일정관리 PIMS 응용프로그램과 데이터스토어 어댑터간의 연동 방법을 보여주고 있다.



<그림 10> 데이터스토어 어댑터와 PIMS 응용프로그램 간의 연동

<표 2> 데이터베이스 반영을 위해 사용되는 클래스들의 각 필드

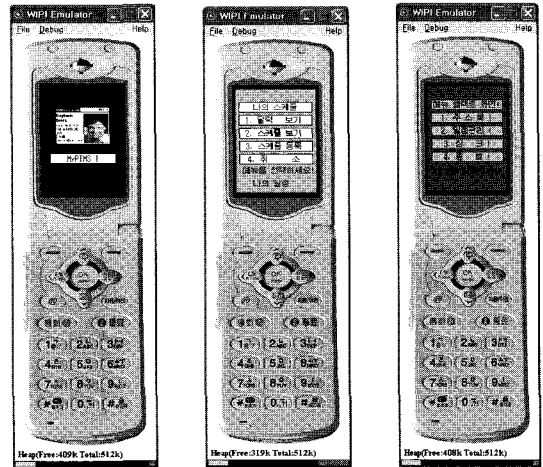
멤버 함수명	역할
updateChangeLog	동기화에 사용할 ChangeLog 리스트를 로드
getNextChange	ChangeLog 리스트에서 로그를 하나씩 얻음
updateItemState	일정관리 서버에 반영된 아이템을 반영
writeChangeLog	ChangeLog를 파일에 저장
getAnchor	앵커정보를 가져옴
putAnchor	앵커정보를 저장
readChangeLog	데이터스토어로부터 ChangeLog를 로드
getData	해당 아이템의 내용을 가져옴
getChangeValue	해당 아이템의 checksum을 가져옴
dsReplace	해당 아이템을 변경
dsDelete	해당 아이템을 삭제
dsAdd	아이템을 추가

ThinkSync 일정관리 서버로부터 변경된 일정 정보를 수신했을 때, 단말기에 내장된 ThinkSync 클라이언트는 수신된 일정정보를 해석한 후, 일정정보 데이터스토어의 아이템에 추가, 변경 및 삭제가 필요한지를 판단하고, 데이터스토어 어

댑터를 호출한다. <표 2>는 데이터스토어 어댑터가 일정관리 정보가 저장된 데이터베이스에 해당 정보를 반영하기 위해서 사용하는 클래스들의 각 필드에 대해서 보여주고 있다.

#### IV. 실험 및 검증

본 논문에서 개발한 SyncML 동기화 방법을 이용한 일정관리 솔루션은 LINUX와 C-언어 환경에서 개발하였으며, 실험에 적용한 휴대형 단말기로는 WIPI Emulator를 사용하였다(위피(WIPI)). <그림 11>은 휴대형 단말기에서 동작하는 일정 관리 솔루션의 실행 후 초기 화면들을 보여주고 있다. 실제 실험에 사용된 모든 일정 정보들은 동기화 서버 및 다른 모든 단말기에도 같은 내용이 저장되어 있다.



<그림 11> 휴대형 단말기용 일정관리 솔루션

<그림 12>는 두 명의 사용자가 자신의 단말기에 일정을 등록하고, 열람할 수 있는 화면과 각각의 일정에 대한 자료를 입력 한 후, 동일한 일정 자료에 대해서 특정 사용자가 정보를 변경하는 화면을 보여주고 있다. 예를 들면, 오전에 출근을 한 A와 B라는 사원이 각각 자신의 휴대형 단말기에 오늘의 일정을 입력한 후, 영업활동을

하기 위해 사무실 바깥을 나선다. 그 후, A 사원의 일정 변경 요청으로 인해 팀장회의 시간이 연기된다. 이러한 변경된 사실을 중앙관리 서버와 자료 동기화를 통해 B 사원의 휴대형 단말기에도 변경된 스케줄이 저장된다.

<그림 13>은 동일한 일정 자료에 대해서 변경

이 발생했을 경우에 자료 동기화 과정을 수행함으로써, 각각의 단말기에 저장된 동일한 자료에 대해서 일치성이 보장됨을 보여주고 있다. 따라서 A와 B 사원의 휴대형 단말기에는 갱신된 팀장회의 시간이 일치되어 같은 일정정보가 저장되어 있음을 알 수 있다.



<그림 12> A와 B 사원의 변경 전 일정(좌)과 변경 후 일정(우)



<그림 13> 자료 동기화 수행 화면과 수행 후 일치성이 보장된 화면

## V. 결 론

본 논문에서는 자료 동기화 방법의 실제 응용 분야에 적용하기 위한 연구의 일환으로써, SyncML 기반의 자료 동기화 방법을 이용하여 기업 운영의 효율성을 향상시키기 위해서 각각의 외근 사원들이 소유하고 있는 휴대형 단말기의 일정 관리를 효율적으로 일치시키기 위한 ThinkSync 일정관리 솔루션을 개발하였다. 또한, 실제 일정 자료를 각각의 단말기에 입력한 후, 동일한 자료에 대해서 변경을 발생했을 때에 자료 동기화 동작이 적합하게 수행되어 자료가 일치됨을 검증해 보았다. 또한, 본 논문에서 개발한 SyncML 자료 동기화를 이용한 ThinkSync 일정 관리 솔루션의 전체적인 구조 및 각 데이터 조작 모듈 구조도를 제시하였고, 데이터 조작 모듈 부분의 처리 흐름을 중점적으로 기술하였으며, 개발한 일정 관리 솔루션은 LINUX와 C-언어 환경에서 개발하였고, 휴대형 단말기로는 WIPI Emulator를 사용하였다.

향후, 다양한 인증 방법을 제공함으로써 보안 기능을 강화시킴과 동시에, 다수의 단말기에서 동일한 자료에 대해서 동시에 변경사항이 발생했을 경우에 자료 충돌에 대한 효율적인 정책을 제공할 필요가 있다.

## 참 고 문 헌

권지인, "휴대형 단말기(Handheld Device)시장 동향 및 전망", 정보통신정책, 제16권, 20호, 2004년 11월.

위피(WIPI), <http://www.wipi.or.kr>.

장대진, 박기현, 주홍택, "SyncML 기반의 자료 동기화 클라이언트 개발", 정보과학회 논문지, 제11권, 제4호, 2005년 8월, pp. 357-367.

한국전자통신 연구원기술평가팀, "PDA 및 스마트폰 기술 및 시장 동향", 정보통신기반조성사업 보고서, 2005년 6월.

ActiveSync, <http://msdn.microsoft.com/library/windowsmobile>.

Ju, H. T., D. J. Jang, K. Park, B. H. Ha, M. C. Lee, and S. Bae, "Design of ThinkSync DM based on SyncML Device Management", The 3rd Asia Pacific International Symposium on Information Technology, KIPS, Dec. 2004, pp. 569-574.

HotSync, <http://www.palmsource.com/palms/hot-sync.html>.

Ren, R. and J. Song, "Data Synchronization in the Mobile Internet", The 7th IEEE International Conference, September 2002, pp. 95-98.

Butrico, M., *et al.*, "Enterprise Data Access from Mobile Computers: An End-to-end Story", The 10th IEEE International Workshop on Research Issue in Data Engineering, February 2000, pp. 9-16.

Agarwal, S., D. Starobinski, and A. Trachtenberg, "On the Scalability of Data Synchronization Protocols for PDAs and Mobile Devices", *IEEE Networks*, Vol.16, No.4, June 2002, pp. 22-28.

SyncML Initiative, SyncML White Paper version 1.0, 2000.

SyncML Initiative, SyncML Representation Protocol Specification version 1.1, 2002.

SyncML Initiative, SyncML Synchronization Protocol Specification version 1.1, 2002.

Hansmann, U., R. Mettala, A. Purakayastha, and P. Thompson, *SYNCML: Synchronizing and Managing Your Mobile Data*, Prentice Hall, 2002.

## Schedule Management for Outside Duty Personnels Using Mobile Communication Data Synchronization

Dae Jin Jang\* · Kee Hyun Park\*\* · Hong Taek Ju\*\* · Sang Jin Yoo\*\*\*

### Abstract

Schedule Management for outside duty personnels and their groups is fairly difficult since they spend most of their time on working outside individually rather than inside office. Everytime when schedule change for personnels or their groups is needed, it is necessary to call every personnel in order to notify the schedule change. It would be much easier to handle the schedule management problem if outside duty personnels are able to download schedule of their groups (and, upload their schedule) using their mobile communication devices such as PDAs.

In this paper, using a data synchronization protocol in mobile communication environments, a schedule management system for outside duty personnels is designed and implemented. Data synchronization in mobile communication environments is a process of maintaining consistency between data stored in mobile devices and data stored in a (central) server at office. In other words, using SyncML data synchronization protocol proposed by OMA(Open Mobile Alliance) as an open standard for data synchronization in mobile communication environments, a schedule management system for outside duty personnels which allows them to check schedule of their groups and to notify their schedule to their groups at anytime anyplace is designed and implemented.

**Keywords:** *Schedule Management, Data Synchronization, SyncML, OMA*

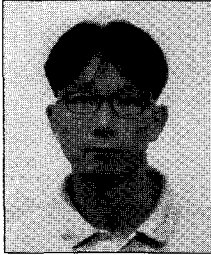
---

\* Doctor candidate, Department of Computer Engineering, Keimyung University

\*\* Professor, Department of Computer Engineering, Keimyung University

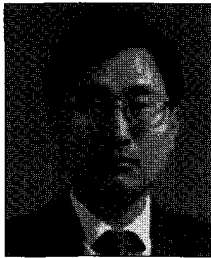
\*\*\* Professor, Department of MIS, Keimyung University

## ◎ 저 자 소개 ◎



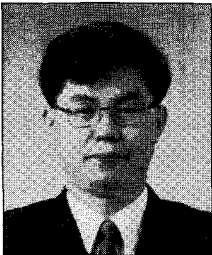
**장 대 진 (djang19@kmu.ac.kr)**

계명대학교 컴퓨터공학과에서 학사, 석사학위를 취득하였으며, 현재 계명대학교 컴퓨터공학과 박사과정에 재학하고 있다. 주요 관심분야는 모바일 데이터동기화, 무선통신 시스템 등이다.



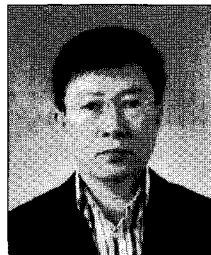
**박 기 현 (khp@kmu.ac.kr)**

경북대학교에서 컴퓨터공학전공으로 전자공학과를 졸업하였으며, 한국과학기술원 전자계산학과에서 석사학위를, 미국 밴드빌트 대학교 컴퓨터공학과에서 박사학위를 각각 취득하였다. 1981년부터 계명대학교 컴퓨터공학부 교수로 재직하고 있으며, 주요 관심분야는 병렬운영체제, 모바일통신 소프트웨어, u-헬스케어 시스템, 성능분석 등이다.



**주 흥 택 (juht@kmu.ac.kr)**

한국과학기술원 전산학과를 졸업하고, 포항공과대학교 컴퓨터공학에서 석사 및 박사학위를 취득하였다. 대우통신 종합연구소에서 선임연구원으로 재직하였고, 현재 계명대학교 컴퓨터공학부 조교수로 재직 중이다. 주요 관심분야는 Web-based Network Management, Network Monitoring 등이다.



**유 상 진 (yoosj@kmu.ac.kr)**

서강대학교에서 물리학(이학사), 경영학(경영학사)을 복수 전공하였으며, 미국 University of Nebraska-Lincoln에서 MIS전공으로 박사학위를 취득하였다. 현재 계명대학교 경영정보학과 교수로 재직 중이며, 현직에 오기 전에는 미국 Bowling Green State University 조교수로 재직하였다. 한국경영정보학회 부회장, 한국 정보시스템학회 회장을 역임하였으며, 현재 대구경북 CIO협의회 회장, 대구경북 ECRC전문위원, 대한상사 중재원 중재인으로 활동하고 있다. 주요 관심분야는 IS/IT의 전략적 활용, 경영혁신, 지식경영, 인터넷창업 등이다.