

# 개인 휴대단말에서 응용 프로그램 동기화를 위한 자동설치 시스템의 설계 및 구현

나 승 원<sup>†</sup> · 오 세 만<sup>††</sup>

## 요 약

새로운 무선인터넷 디바이스로써 주목받고 있는 PDA는 기능 개선을 위해서 다양한 애플리케이션 개발이 이루어지고 있으며 수시로 응용프로그램이 설치되어지고 있다. 또한 PDA는 전원이 방전 될 경우에 RAM(Random Access Memory)에 저장되어 있는 데이터가 모두 삭제되는 단점이 있어서 그때마다 프로그램을 다시 복원해야 한다. 본 논문에서는 PDA의 응용 프로그램을 설치할 경우에 사용자가 직접 애플리케이션을 설치해야 하는 기존의 문제점을 해결하기 위하여 자동으로 PDA 응용프로그램이 설치되는 시스템(PAIS : PDA Automatic Installation System)을 제안하였다. 이 시스템을 적용할 경우 사용자에게는 설치에 소요되는 시간과 노력을 감소시켜 편의를 제공하고, 기업에게는 프로그램 설치 관련 홍보물 제작에 소요되던 비용을 절감 할 수 있는 효과가 있다.

## Design and Implementation of Automatic Installation System for Application Synchronization in PDA

Seungwon Na<sup>†</sup> · Seman Oh<sup>††</sup>

## ABSTRACT

The application for PDA as a new wireless Internet device is variously developed to improve the function of PDA, and the application software must be installed to PDAs. In addition, when the power supply of PDA becomes fully discharged, all data stored in the RAM (Random Access Memory) could be vanished, and then the application programs should be reinstalled. This paper presents an automatic installation system of application program, PAIS (PDA Automatic Installation System). It is designed to solve the problem of PDA that PDA's users have to install the application programs on their PDA themselves. When this engine is applied, PDA users can save the time and effort for installation. It's convenient for the PDA users. The PDA and application software company can save the cost to create materials to explain the installation process.

**키워드 :** 자동설치시스템(Auto Installation System), 동기화 엔진(Sync Engine), SyncML, PDA(Personal Digital Assistant)

### 1. 서 론

인터넷의 대중적 확산과 이동통신 기술의 발전으로 출발한 무선 인터넷은 1999년 이후, 국내에서 휴대폰을 통해서 처음으로 제공되었다. 그러나 휴대폰의 제한된 처리 능력과 협소한 메모리는 무선 인터넷 전용 디바이스로 활용 되기에는 많은 개선 사항이 요구되고 있었다. 따라서 휴대폰 보다 다양한 기능을 제공할 수 있는 터미널이 요구되었고, 새롭게 등장한 PDA(Personal Digital Assistants)가 부족했던 기능을 대신하며 차세대 모바일 인터넷 디바이스로 주목받는 상황이 되었다[1]. 현재 PDA는 차세대 무선 인터넷 도구로써 많은 수요가 발생하고 있다. 국내 이동 3사의 PDA의 가입자는 2003년 9월말 현재 20여 기종에 30만명이 이용하고 있으며 가입자는 계속해서 확산될 추세이다[13].

그러나 PDA는 응용분야가 확대됨에 따라서 다양한 형태의 애플리케이션이 개발되고 있으며 사용자는 손수 해당 프로그램을 설치하여야 한다. 또한 PDA는 전원이 방전될 경우 램(RAM : Random Access Memory)에 저장되어 있던 모든 데이터가 삭제되는 단점이 있어서 기존에 있던 응용 프로그램까지 재 설치해야 하는 단점이 있다.

본 논문에서는 PDA에서 응용 프로그램이 자동으로 설치될 수 있는 애플리케이션 자동설치 시스템(PAIS : PDA Automatic Installation System)을 제안하였다. PAIS를 적용할 경우 사용자에는 재 설치에 대한 시간과 노력을 감소시켜 사용자 중심의 컴퓨팅 환경을 제공할 수 있으며, 제조사나 이동 통신사에게는 설치를 위해 홍보물과 CD 제작등 기존 제작에 소요되던 비용을 절감할 수 있는 효과를 제공한다.

### 2. 관련 연구

관련 연구 사항으로 PDA의 메모리 구조와 데이터 동기

† 준 회원 : 동국대학교 대학원 컴퓨터공학과

†† 종신회원 : 동국대학교 컴퓨터공학과 교수

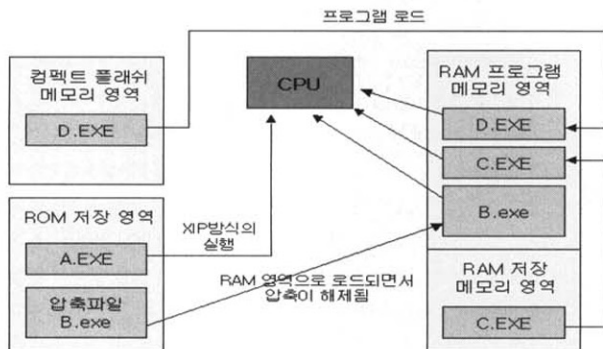
논문접수 : 2003년 7월 21일, 심사완료 : 2003년 11월 12일

화의 표준을 위한 SyncML을 소개하고 기존에 소프트웨어 자동 배포와 관련된 지식들에 대해서 표준화한 언어와 스키마 형식들을 조사하였다.

2.1 PDA의 메모리 구조

PDA의 메모리는 롬(ROM : Read Only Memory)과 램(RAM : Random Access Memory)으로 구성된다. 롬에는 운영체제와 DLL(Dynamic Linking Library) 그리고 운영체제에 포함되어 있는 응용 프로그램들이 저장되어 있다. 롬은 일반적인 롬과 플래쉬 메모리 롬으로 다시 구분된다. 롬 저장 영역에서는 저장된 프로그램 모듈이 압축되지 않은 상태인 경우에는 램에 로드(load)없이 바로 실행되는 XIP(Execute In-Place)방식으로 제공된다. 반면에, 롬 저장 영역에 압축된 파일들은 램 영역으로 로드된 후 실행되는 구조이다[2]. 이 경우 램은 프로그램 메모리 영역을 칭한다. 일반적인 롬은 사용자가 읽기만 가능한 영역이다. 이에 반해서 사용자가 읽고 쓰기가 가능한 플래쉬 메모리(Flash Memory)는 배터리 방전시에도 데이터가 보존되는 영역이다.

그리고 램은 저장용 메모리(Storage Memory)와 프로그램 메모리(Program Memory)의 영역으로 구분된다. 저장용 메모리는 데스크 탑의 하드디스크와 유사한 기능을 제공하며 오브젝트 스토어(Object Store)로 불려진다. 여기서는 응용 프로그램과 데이터 파일을 저장하는 파일 시스템, 사용자 데이터를 저장하는 데이터 베이스, 시스템 정보를 구성하는 레지스트리로 구성되어 있다. 그리고 램 프로그램 메모리(그림 1)에서는 파일(B, C, D.exe)등이 로드된 후 실행되는 영역을 말한다. 램 프로그램의 응용 프로그램이 실행될 때 힙(heap)과 스택(stack)에 데이터를 저장한다. 램은 충전 배터리의 전원이 모두 소멸될 경우 램에 모든 데이터가 소멸되어 버리는 특성을 가지고 있으며 (그림 1)은 윈도우 CE의 메모리 구조를 나타낸 사항이다[14].



(그림 1) PDA의 메모리 구조

(그림 1)에서 롬 공간에 있는 프로그램 실행 파일이 압축되어 있는 구조이면 램에 로드된 후 실행되지만, 압축되지 않은 일반적인 프로그램은 롬에서 바로 수행되어 램의 공간을 추가로 할당하지 않아도 된다.

2.2 SyncML(Synchronization Markup Language)

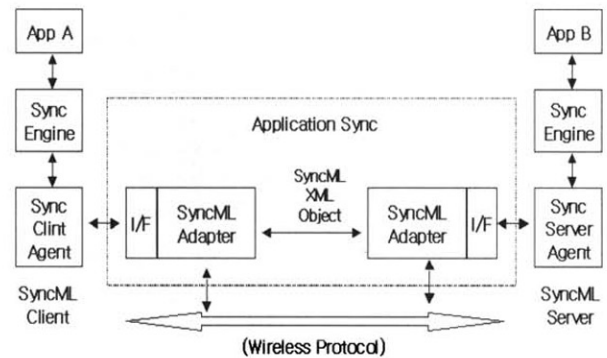
SyncML은 모든 종류의 네트워크에서 모든 디바이스에 있는 데이터의 동기화를 목적으로 만든 언어이며 XML을 사용한다. SyncML이 메시지를 전송하기 위해서 메시지의 데이터 포맷을 정의한 표현(Representation) 프로토콜과 동기화 규칙을 정의한 동기화(Synchronization) 프로토콜 그리고 전송 프로토콜의 바인딩을 정의하고 있다[4,5].

첫째, SyncML 데이터 표현 프로토콜은 데이터, 메타 데이터, 동기화 명령어등과 같이 동기화를 수행하기 위해서 필요한 모든 표현 양식을 표시할 수 있는 XML의 DTD(Document Type Definition)를 정의하고 있는 SyncML 메시지에 대한 구조체 규약이다.

둘째, 동기화 프로토콜은 SyncML 서버와 클라이언트 간에 데이터를 추가, 삭제, 변경하기 위해서 SyncML 메시지가 교환하는 방법, 실제 동기화를 위해서 동작하는 방법 그리고 동기화 타입의 방법을 정의한다.

셋째, 전송 프로토콜은 HTTP, WSP(Wireless Session Protocol), OBEX(Object Exchange Protocol)로써 데이터 표현 프로토콜과 동기 프로토콜이 전송 프로토콜에 독립적으로 다른 전송 프로토콜과 바인딩이 가능하다.

다음 (그림 2)는 SyncML의 프레임워크 구조도를 나타낸 사항이다[7].

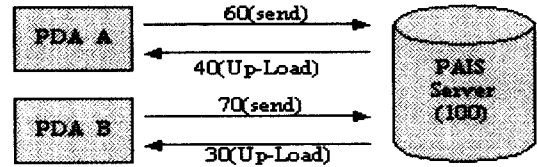


(그림 2) SyncML 프레임 워크 구조도

(그림 2)에서 프레임워크는 SyncML 표현 프로토콜과 SyncML Adapter, SyncML I/F로 이루어져 있다. 이들 프레임 워크는 "App A"와 "App B"의 데이터 동기화를 위한 구성 요소로 HTTP, WSP, OBEX와 같은 네트워크 트랜스포트들 통해 연결되어 있다. 그리고 Sync Engine은 데이터 동기화 프로토콜을 구현하고, 응용 서비스가 이를 이용한다. Sync Server Agent는 네트워크로 접근하는 Sync 엔진을 관리하고 클라이언트 애플리케이션과 데이터 동기화를 위한 통신 기능을 하며, Sync Server Agent의 기능들은 SyncML I/F를 호출함으로써 수행된다. SyncML I/F는 SyncML Adapter에 대한 API이고 SyncML Client Agent는 SyncML I/F를 통해 App B가 네트워크와 SyncML Adapter에 접근하도록 한다.

### 2.3 응용 프로그램 자동설치 시스템

본 글에서는 웹에서 반영될 소프트웨어 자동 배포와 관련된 연구 사항으로 배포시 필요한 지식 들을 표준화된 언어와 스키마 형식으로 기술한 것에 대한 연구를 소개하고자 한다[8].



(그림 3) PAIS 동작원리

#### 2.3.1 DSD(Deployable Software Description)

DSD는 콜로라도 대학에서 개발된 소프트웨어 시스템의 기술 형식으로 XML 응용의 하나이다. DSD의 특징은 다양한 플랫폼에 탑재되는 모든 소프트웨어를 하나의 DSD 형식으로 기술하고 있다. 이것은 동일한 제품군에 속하는 특정 버전과 플랫폼에 해당되는 소프트웨어에 대한 관리를 용이하게 지원한다는 특징이 있으며, 소프트웨어가 배포되는 사용자 사이트에 대한 정보 즉, 플랫폼, 운영체제, 이전에 설치된 소프트웨어 정보 등의 기술을 지원하며 간단한 형태의 수식 언어(Expression Language)가 내재되어 있다. DSD를 이용한 자동배포 시스템으로는 Software Dock Framework이 있다.

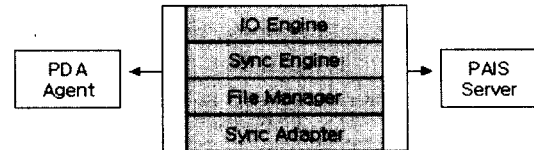
#### 2.3.2 OSD(Open Software Description)

OSD는 푸시(Push) 패러다임을 적용한 세 가지의 특징을 제공하기 위하여 Marimba와 Microsoft가 공동으로 W3C에 제안한 XML의 응용이다. 첫째는 사용자의 개입이 없는 소프트웨어의 자동설치를 의미하는 Hans-Free Install이고 둘째는 소프트웨어 자동 업그레이드인 Easy and Timely Upgrade 그리고 셋째는 동일한 소프트웨어 버전에 해당하는 상이한 플랫폼에 탑재된 소프트웨어들의 정보들을 하나의 OSD 형식으로 기술함으로써 소프트웨어 설치 시에 플랫폼에 따른 복잡성을 줄이는 것을 뜻하는 Cross Platform을 의미한다. OSD는 소프트웨어 패키징을 위한 구성 요소들의 정보 즉, 버전정보, 내부구조, 요소간 상호관계를 기술하기 위한 단어로 구성된다.

(그림 3)에서 각각의 PDA(A, B)의 내부 애플리케이션 정보(60, 70)를 서버로 송신하고, 서버의 기준 정보(100)와 매핑 테이블에 의해서 비교한 후, 설치 파일(40, 30)을 결정하고 해당 패키지를 생성한다. 이것을 디바이스로 업로드하는 과정이다.

PAIS의 설계는 크게 두 가지로 구성된다. 첫째는 PDA 내부의 에이전트 설계이고 둘째는 설치 파일을 관리하는 PAIS 서버의 설계이다.

PAIS 서버는 실제로 클라이언트로부터 수신받은 데이터와 서버가 가지고 있는 데이터를 매핑하여 클라이언트 프로그램의 버전 체크 또는 존재 유무를 판단하는 역할을 하며 PAIS 클라이언트와 서버는 파일 버전 관리와 성능 향상을 위하여 (그림 4)와 같은 형식의 엔진 모듈로 구성되어 있다.



(그림 4) PAIS 엔진 모듈구조

(그림 4)의 엔진 모듈은 클라이언트와 서버영역 모두에서 적용되었으며 세부 기능은 다음과 같다.

첫째, IO Engine은 SyncML 형태의 문서를 이용 가능한 Transport Binding. 즉, HTTP 프로토콜에 맞추어서 서버와 클라이언트에 대한 전송과 수신을 담당한다.

둘째, Sync Engine은 SyncML Package의 프로세싱을 담당하며 동기화 모듈들을 핸들링 한다. 실제로 Sync Operation을 수행하는 모듈과 클라이언트측의 데이터를 비교하여 데이터를 추가할지, 갱신할지, 삭제할지를 결정하는 역할을 한다.

셋째, File Manager는 클라이언트의 디바이스 정보 및 프로그램 버전등 존재유무가 체크된 후 해당 디바이스와 프로그램에 대한 다운로드를 관리한다.

넷째, Sync Adapter는 SyncML Interface에서 추출된 데이터(vCard, iCalendar)를 Sync Engine의 형식에 맞추어 인식할 수 있도록 분석하고 SyncML 문서로 만드는 기능을 수행한다.

## 3. PDA 자동설치 시스템의 설계

본 글에서는 이동 단말인 PDA에서 수행되는 응용프로그램의 실행 파일을 무선으로 다운받아 자동 설치하는 시스템인 PAIS(PDA Automatic Installation System)을 제안하고자 한다. PDA 상에서 무선으로 데이터를 다운받아 설치하는 시스템은 아직 상용화된 적이 없다. 이것은 통신 망의 제약으로 인해서 무거운 설치 파일을 전송받기가 부담스러웠으나 현재 국내에서 CDMA 1X EV-DO가 상용화되고 있는 시점에서는 충분히 제공될 수 있는 서비스 형태이다.

### 3.1 PAIS의 전체구성

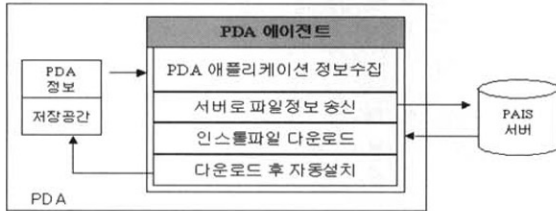
PAIS는 PDA 내부의 실행파일 정보를 서버로 송신한 후 최종의 설치 파일 들을 관리하고 있는 서버로부터 인스톨 패키지 파일을 다운로드 받아 자동으로 설치하는 시스템이다.

PAIS의 동작 원리는 (그림 3)과 같다.

### 3.2 PDA 에이전트의 설계

PDA 에이전트의 주요 기능은 PDA의 기존 응용프로그램

정보를 수집하여 PAIS 서버에 전달하고 서버로부터 최종의 설치 파일을 다운로드 받아 해당 디렉토리에 자동으로 설치한다.



(그림 5) PDA Agent의 구조

(그림 5)에서 정보를 수집하는 애플리케이션의 정보는 <표 1>의 내부 수집 정보를 기준으로 한다.

그리고 데이터 전송 후 서버로부터 다운받는 파일은 패키지 파일이며 본 시스템에서는 캡(cab) 파일로 처리하였다. 서버로부터 다운 받은 파일은 자동 설치를 수행하게 되며 <표 1>은 PDA에서 수집하는 테이블 사례이다.

<표 1> PDA 내부 수집정보

구분	상세 정보	ID군
고객정보	고객ID, 전화번호(MIN)	10~
PDA 종류	HP, Samsung, SONY	20~
PDA O/S	PPC 2002, Palm, Linux	30~
응용 프로그램	Internet Brower, e-Book viewer, PIMS, VM등	40~

레지스트리에서 수집되는 PDA의 정보를 수집하는 과정을 (그림 6)으로 나타내었다.



(그림 6) PAIS Agent에서 수집정보 파일 처리과정

(그림 7)은 서버로 송신될 때 애플리케이션 정보를 SyncML 형태로 처리하여 보내주는 형태로써 PDA 에이전트에 서 설계한 SyncML 코드의 일부이다.

```
<?xml version = "1.0" encoding = "UTF-8"?>
<SyncML xmlns = 'SYNXML : SYNXML1.0'>
<SyncHdr><VerDTD> 1.0 </VerDTD><VerProto> SyncML/1.0
</VerProto>
<SessionID> 1 </SessionID> <MsgID> 2 </MsgID>
<Target><LocURI> pdal </LocURI></Target>
```

```
<Source><LocURI> http://www.PAIS.com/basic.asp </LocURI>
</Source>
</SyncHdr>
<SyncBody>
<Status>
<CmdID> 1 </CmdID><MsgRef> 2 </MsgRef><CmdRef> 0
</CmdRef>
<Cmd> SyncHdr </Cmd>
<TargetRef> 203.236.33.43/cgi-bin/pais.cgi </TargetRef>
<SourceRef> pdal </SourceRef>
<Data> 212 </Data></Status>
<Status><CmdID> 2 </CmdID><MsgRef> 2 </MsgRef>
<CmdRef> 3 </CmdRef>
<Cmd> Sync </Cmd>
<TargetRef> PAIS </TargetRef>
<SourceRef> PAIS </SourceRef>
<Data> BEGIN : VCARD
USER_ID : angel
PASSWD : aaaaa
PHONE_NUMBER : 011-234-5678
DEVICE_ID : 22
OS : 31
APP1 : 41/20
APP2 : 42/11
APP3 : 44/12
END : VCARD
</Data>
</Status>
<Sync><CmdID> 4 </CmdID>
<Target><LocURI> PAIS </LocURI></Target>
<Source><LocURI> PAIS </LocURI></Source></Sync>
<Final/>
</SyncBody></SyncML>
</xml>
```

(그림 7) PAIS 클라이언트의 SyncML 적용 코드사례

### 3.3 PAIS 서버의 설계

PAIS 서버는 다섯 가지의 엔진 모듈로 구성된다. 주요 기능은 PDA에 최신의 애플리케이션을 제공하는 기능이 있으며 (그림 8)과 같이 엔진 모듈로 구성되어 있다.

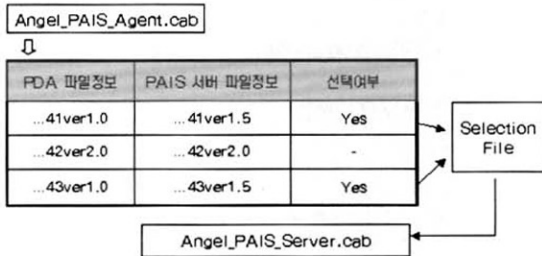


(그림 8) PAIS Server의 구조

(그림 8)에서 I/O 엔진은 PDA 애플리케이션 정보를 수신하고 해당 파일을 HTTP 프로토콜을 통해서 전송하는 역할을 담당한다. Sync 엔진은 PDA와 서버측의 애플리케이션 데이터를 비교하여 데이터를 추가할지, 갱신할지를 결정하게 된다. File 매니저는 PDA로 송신할 애플리케이션의 최신 정보를 관리하는 영역이다. Sync 어댑터는 PDA로부터 송신된 SyncML 문서 형식의 애플리케이션 정보를 파싱하고 서버측의 애플리케이션 데이터를 SyncML 문서로 만들어서 I/O 엔진으로 넘겨준다. 마지막으로 SMS(Short Me-

ssage System) 서비스는 최신의 정보가 필요한 상황일 경우에 SMS 서버로부터 데이터를 공지하는 기능이 연계되어 수행할 수 있다.

파일 매니저 영역에서는 PDA에 설치될 가장 최신의 애플리케이션을 관리하는 영역으로 (그림 9)와 같이 매핑 테이블로 관리하게 된다.



(그림 9) 설치파일 매핑테이블 관리 사례

(그림 9)의 첫 번째 열과 두 번째 열이 비교되어 마지막에서 PDA로 송신할 애플리케이션이 선택되고 파일들의 모음인 패키지 파일이 생성되는 일련의 과정이다. PAIS 서버에서 파일을 비교하는 기준은 버전 정보를 통해서 이루어지며 상위의 버전일 경우에만 확정된 설치 파일로 선택된다. 패키지 파일에는 기본 파일목록 URL, 상세 파일정보 URL, 다운로드 URL 등 3개의 URL 정보가 포함되어 있으며 적용 사례는 다음과 같다.

전송방식 : 서버주소?UID = 서비스ID\_플랫폼ID\_응용프로그램ID  
 [BASIC\_URL] = http://www.PAIS.com/basic.asp  
 [DETAIL\_URL] = http://www.PAIS.com/detail.asp  
 [DOWNLOAD\_URL] = http://www.PAIS.com/download.asp

(그림 10)은 PAIS서버 측면에서 적용한 SyncML 코드의 일부 사례이다.

```
<?xml version = "1.0" encoding = "UTF-8"?>
<SyncML xmlns = 'SYNCML:SYNCML1.0'>
<SyncHdr><VerDTD> 1.0 </VerDTD><VerProto> SyncML/1.0
</VerProto>
<SessionID> 152 </SessionID><MsgID> 2 </MsgID><Target>
<LocURI> PAIS </LocURI></Target><Source>
<LocURI> http://www.PAIS.com/download.asp </LocURI>
</Source></SyncHdr>
<SyncBody>
<Status>
<CmdID> 1 </CmdID><MsgRef> 2 </MsgRef><CmdRef> 0
</CmdRef>
<Cmd> SyncHdr </Cmd><TargetRef>
http://www.PAIS.com/download.asp </TargetRef>
<SourceRef> PAIS </SourceRef><Chal><Meta>
<Format xmlns= 'syncml:metinf' b64 </Format>
<Type xmlns= 'syncml:metinf' syncml: auth-basic </Type>
</Meta></Chal>
<Data>BEGIN : VCARD
APP1 : 41/30 // 수정된 버전 정보
APP2 : 42/12
```

```
APP3 : 44/20
END : VCARD
</Data>
</Status><Final/>
</SyncBody>
</SyncML>
</xml>
```

(그림10) PAIS 서버의 SyncML 코드 적용예제

#### 4. PDA 자동설치 시스템의 구현

본 논문에서 제안된 PAIS를 프로토타입으로 구현하여 보았으며 이를 위한 개발도구 및 구현환경은 다음과 같다.

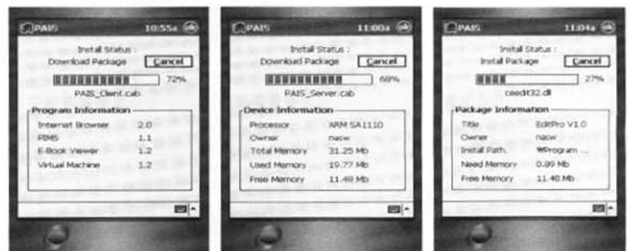
- 개발도구 및 구현환경
  - Embedded Visual C++ 3.0(language)
  - Pocket PC PDA
  - Pocket2002(O/S)
  - CDMA 1X EV-DO
- 실험 대상 애플리케이션
 

실험 대상이 되는 애플리케이션의 정보는 기존 PDA에서 가장 빈번하게 사용되는 프로그램으로 <표 2>와 같이 적용하였다.

<표 2> 실험파일 예제

실험 파일	파일 크기
Internet Browser	1.0MB
PIMS	1.6MB
e-Book Viewer	0.9MB
Virtual Machine	1.4MB

PDA의 응용프로그램을 다운로드 받아 자동으로 설치되는 과정을 (그림 11)과 같이 구현하였다.

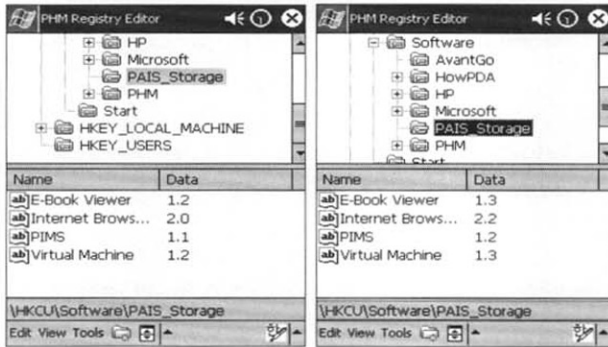


(그림 11) 구현 결과

(그림 11)에서 좌측은 디바이스의 정보를 서버로 송신하는 화면이고, 중간 그림은 서버로부터 인스톨 파일을 다운로드 받는 화면이다. 이때 PDA에서 서비스를 위한 실행 파일의 형태를 패키지 형식으로 다운로드 받는다. 그리고 오른쪽 그림은 설치 받은 패키지 파일이 자동으로 인스톨 되고있는 화면이다. 설치 파일을 호출하여 다운로드 받아 설치하는 과정은 모두 12분 정도 소요되었다.



PDA에 다운로드 받은 후 설치된 결과는 (그림 12)와 같이 PDA의 레지스트리 정보를 통해서 알 수 있다.



(그림 12) 설치 전후 레지스트리 결과화면

(그림 12)에서 PDA의 레지스트리 내부에 있는 HKEY\_CLASSES\_ROOT의 디렉토리에 애플리케이션의 설치 변경 정보를 나타내었다. 좌측의 그림은 이전의 레지스트리 정보를 보이는 화면이고, 우측은 PAIS 적용 이후 변경된 레지스트리의 정보를 보여주고 있는 화면이다.

### 5. 결론 및 연구방향

휴대 정보터미널인 PDA는 휴대폰을 대신해서 차세대 무선 인터넷 디바이스로 활용되어 가고 있다. 무선인터넷 서비스가 텍스트에서 동영상 형태의 서비스로 변경되어지는 현재의 환경에서 PDA는 적합한 모바일 디바이스로 대두되고 있는 사항이다. 그만큼 다양한 서비스를 제공할 수 있는 영역이 휴대폰에 비해서 우수하기 때문이다. 이와 같이 PDA에서는 다양한 서비스를 수행하기 위해서 많은 기능 개선이 이루어지고 있으며, 사용자는 그 만큼 번거로운 설치 작업을 수행하여야 한다. 또한 현재의 PDA 메모리 구조는 배터리의 전원이 방전될 경우 램에 저장되어있는 모든 데이터 파일들이 소멸되는 단점이 있다. 이 경우에도 사용자는 램에 설치하였던 애플리케이션을 다시 설치해야만 한다.

본 논문에서는 이러한 단점을 해결하기 위해서 PDA 내부에 수행되는 애플리케이션을 자동으로 설치하여 주는 시스템에 대해서 제안하였다. PAIS를 적용할 경우 단말기 제조사나 이동통신 회사에게는 기존 설치 프로그램의 홍보물 제작에 소요되던 비용을 절감하는 효과가 있으며 사용자에게는 설치에 대한 시간과 노력을 감소시켜 편리성을 제공하여, 나아가서는 무선 인터넷의 활성화를 기대할 수 있다. 그러나 무선상에서 원활한 자동설치시스템의 적용을 위해서는 무엇보다도 통신 속도의 개선이 현실화 되어야 할 것이다.

향후의 연구 계획은 PAIS를 기타 임베디드 시스템에서도 적용이 가능한 범용성이 있는 수행 구조로 고도화를 추진하는 연구를 진행할 계획이다.

### 참 고 문 헌

- [1] Intromobile, Mobile Multimedia Technology Trend, <http://www.intromobile.co.kr/solution/>.
- [2] JAMES Y, WILSON, Building Powerful Platforms with windows CE, 2Edition, Addison Wesley, 2000.
- [3] Microsoft PocketPC Official site, <http://www.microsoft.com/mobile/pocketpc/>.
- [4] Sync Initiative, <http://www.openmobilealliance.org/syncml/>.
- [5] Sync Initiative, Sync Architecture version 0.2, 2000.
- [6] Uwe Hansman, Synchronizing and Managing Your Mobile Data, Prentice Hall, 2002.
- [7] 류수희, "Sync프로토콜을 이용한 데이터 동기화 서버 Agent 설계 및 구현", 정보처리학회 추계학술 대회, Vol.8, No.2, pp.1337-1530, 2001.
- [8] 박지은, "이동 단말을 위한 SyncML 기반 응용관리 시스템의 설계 및 구현", 정보과학회논문지C, Vol.8, No.1, pp.62-70, 2002.
- [9] 안태균외 3인, 포켓 PC와 함께하는 모바일 프로그래밍, 정보게이트, 2002.
- [10] 이병윤, "세션 매니저를 이용한 Sync 동기화 시스템의 설계 및 구현", 한국정보과학회논문지C, Vol.8, No.6, pp.647-656, 2002.
- [11] 이지연, "Sync 데이터 동기화를 위한 데이터 베이스의 설계 및 구현", 한국정보처리학회 추계학술대회, Vol.8, No.2, pp.1343-1346, 2001.
- [12] 일본 멀티미디어 통신 연구회, 그림으로 보는 모바일 컴퓨팅, 교보문고, 2001.
- [13] 정보통신진흥국, 무선인터넷 가입자 현황 [http://www.mic.go.kr/jsp/mic\\_d/d700-0002-1.jsp](http://www.mic.go.kr/jsp/mic_d/d700-0002-1.jsp).
- [14] 정 훈, Add-on WindowsCE API Bible, 글로벌, 2001.
- [15] 하인숙, "모든 동기화 프로그램의 조상 SDA 분석", 마이크로 소프트웨어, pp.309-313, 2001.



#### 나 승 원

e-mail : nasw@dgu.ac.kr  
 1985년 단국대학교 농경제학과(학사)  
 1996년 단국대학교 경영대학원 전자정보관리(석사)  
 1997년~현재 (주)SK Telecom 플랫폼 연구원 재직중

1999년~현재 동국대학교 대학원 컴퓨터공학과 박사과정  
 2001년~현재 명지전문대학 정보통신과 겸임교수 재직중  
 관심분야 : 프로그래밍 언어, 모바일 프로그래밍, 이동 에이전트



#### 오 세 만

e-mail : smoh@dgu.ac.kr  
 1977년 서울대학교 사범대학 수학과(학사)  
 1979년 한국과학기술원 대학원 전산학과(석사)  
 1985년 한국과학기술원 대학원 전산학과(박사)

1985년~현재 동국대학교 컴퓨터공학과 교수  
 관심분야 : 프로그래밍 언어, 컴파일러, 모바일 프로그래밍