

무선단말기 소프트웨어 배포 관리자 시스템 ThinkSync DM SoftMan 개발

최은준, 주홍택, 박기현
e-mail:{niceguy0727, juht, khp}@kmu.ac.kr

Software Release Manager System ThinkSync DM SoftMan for Wireless Devices

Eun-Jun Choi, Hong-Taek Ju, Ki-Hyun Park
College of Information and Communication, Keimyung University

요 약

무선단말기를 위한 소프트웨어 배포관리 시스템을 사용하여 단말기를 사용하고 있는 중에 사용 환경의 변화에 적응하거나 성능 및 기능 개선 그리고 새로운 서비스 개시를 위하여 새로운 소프트웨어를 단말기에 설치, 실행 시킬 수 있다. 또한 불필요한 소프트웨어를 제거하거나 정상동작 여부를 감시할 수도 있다. OMA DM은 무선통신단말기를 관리하기 위한 표준이다. 본 논문에서는 OMA DM 표준을 기반으로 무선단말기 관리 응용으로서 무선통신 단말기 소프트웨어 관리 시스템 ThinkSync DM SoftMan 개발 결과를 제시한다.

1. 서론

무선이동통신 단말기는 기능이 고도화되고 다양한 네트워크 환경에 적응하며 다양한 서비스를 선택적으로 사용할 수 있게 발전되고 있다. 이러한 발전을 가속화하기 위해서는 무선단말기 소프트웨어를 중앙에서 관리하는 소프트웨어 배포관리 시스템이 필요하다. 무선단말기를 위한 소프트웨어 배포관리 시스템을 사용하여 단말기를 사용하고 있는 중에 사용 환경의 변화에 적응하거나 성능 및 기능 개선 그리고 새로운 서비스 개시를 위하여 새로운 소프트웨어를 단말기에 설치, 실행 시킬 수 있다. 또한 불필요한 소프트웨어를 제거하거나 정상동작 여부를 감시할 수도 있다 [1].

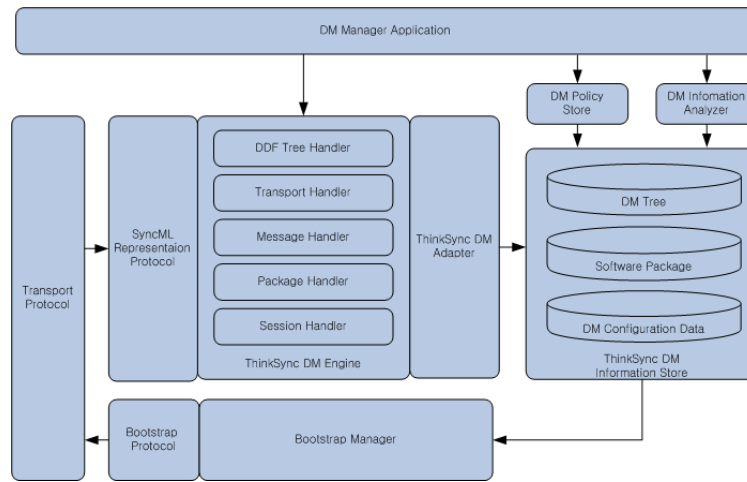
OMA DM은 무선통신단말기를 관리하기 위한 표준이다 [3,4]. OMA DM을 기반으로 무선단말기의 고장, 성능, 구성, 보안 등 다양한 관리 응용이 가능하다. 본 논문에서는 OMA DM 표준을 기반으로 무선단말기 관리 응용으로서 무선통신 단말기 소프트웨어 관리 시스템 ThinkSync DM SoftMan 개발 결

과를 제시한다 [2]. ThinkSync DM SoftMan은 매니저 시스템과 에이전트 시스템으로 구성되어 있으나 본 논문에서는 매니저 시스템 개발 결과를 제시한다. ThinkSync DM SoftMan 관리자 개발 결과에는 시스템 개발 요구사항, 설계 결과 그리고 구현 결과가 포함되어 있다.

2. 관련연구

2.1 OMA DM 단말기 관리 표준

OMA DM 단말기 관리 표준은 관리 프로토콜, 관리 객체 그리고 부트스트랩으로 나누어져 있다. 관리 프로토콜은 관리 명령이 관리 매니저와 관리 에이전트들 간에 전달되는 메시지에 표현되는 형식과 절차에 대하여 기술하고 있다. 단말기 관리 객체는 무선단말기의 구성 요소이면서 단말기 관리의 대상이 되는 객체들에 대한 정보를 트리 구조로 구성하고 그에 따른 객체에 대한 내용을 기술하고 있다. 단말기 관리 부트스트래핑은 단말기 에이전트와 매니저 사이에 관리를 시작하기 전에 수행되는 준비 단계이다.



(그림 1) SoftMan 설계 구조

2.2 OMA DM 적용 사례

공개 프로젝트인 Sync4J에서는 자바를 이용하여 OMA DM 매니저를 개발하였다. [3]. 프랑스 INRIA 연구소의 MADYNES팀에서 소스가 공개된 OMA DM 관리 시스템이다. 자바환경에서 서버가 동작하며, 단말기의 관리 정보를 관리 할 수 있다[4]. Sync4J와 INRIA의 OMA DM은 모두 실험적인 수준에서 적용되고 있으며 실제 사용할 수 있는 관리 응용으로 개발된 사례는 없다.

2.3 소프트웨어 배포 관리 시스템

마이크로소프트사에서 개발한 윈도우 업데이트 시스템은 컴퓨터 운영체제에서 새로 발견된 결함을 보완하기 위한 각종 패치 파일과 하드웨어의 드라이버 업데이트 파일을 제공한다. 이외에도 판매되는 많은 소프트웨어 들이 자체적인 소프트웨어 갱신 방법을 자체적으로 내장하고 있다. 예를 들면 안철수연구소의 V3 소프트웨어는 사용자의 조작 없이 항상 최신 엔진으로 업데이트가 가능하기 때문에 사용자의 컴퓨터에서 가장 빠른 업데이트 서버로 자동 연결되며, IDC를 통한 국내 최대의 업데이트 서버를 운용하고 있어 쉽고 빠르게 업데이트를 할 수가 있다. 프로그램 자체적으로 소프트웨어 갱신 방법을 내장하고 있는 것 이외에도 역시 표준에 의하지는 않지만 소프트웨어 갱신을 기능만을 제공하는 Gentoo의 portage와 같은 패키지 관리 시스템도 있다.

3. ThinkSync DM SoftMan 설계

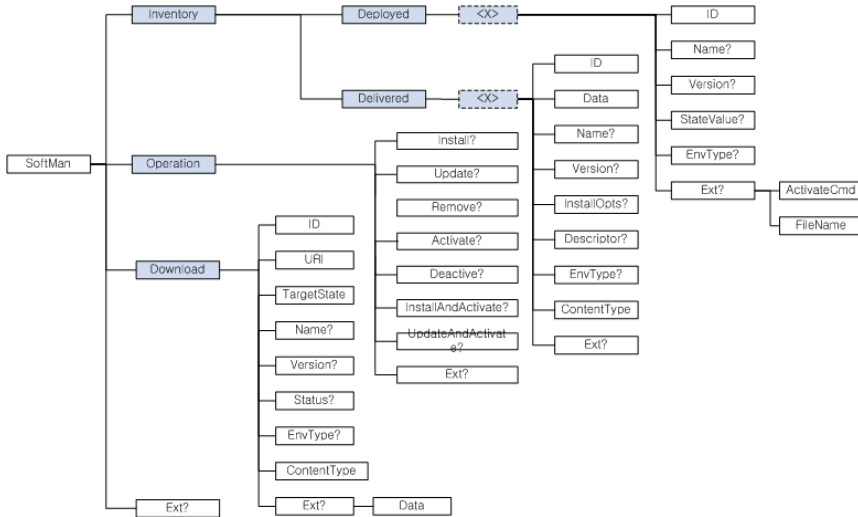
3.1 매니저 시스템 설계 구조

그림 1은 ThinkSync DM SoftMan 매니저의 구조를 나타낸다. 전송 프로토콜(Transport Protocol)은 WSP, OBEX, HTTP등을 사용하여 에이전트와 매니저 사이의 명령어 교환을 한다. 부트스트랩 매니저(Bootstrap Manager)는 사용자가 단말기 구입 시 초기 세팅을 위해서 필요한 단말기 설정 정보인 DM Configuration Data를 에이전트에게 전송하며, 이때 에이전트에게 전송하는 초기 세팅 정보는 매니저 URL, 인증에 필요한 ID와 패스워드 그리고 통신 연결을 위한 네트워크 구성 정보가 된다. 그러나 이 과정은 매니저마다 오직 한번만 수행하므로, 매니저 측에서 새로운 DM구성이 추가되어야 한다면 카드를 바꾸거나 단말기가 재 프로그램 되어야하는 결함이 존재한다.

ThinkSync DM 엔진(Engine)은 OMA DM을 구현한 결과로써 Message Handler, DDF Tree Handler, Transport Handler, Session Handler로 구성된다. DM Manager Application으로부터 전달받은 관리 명령이나 DM Policy Store에 저장된 관리 정책에 따라서 Message Handler는 에이전트에게 전달하기 위한 SyncML 명령어로 생성한다. DDF Tree Handler는 트리에 노드를 추가할 경우 저장된 DDF 명세서의 내용을 보고 트리에 노드를 추가하며, ThinkSync DM DS Adapter는 트리에 추가된 노드를 DM DDF Data에 맵핑하는 연산을 담당한다. Transport Handler는 송수신된 SyncML메시지의 처

리를 담당한다. 송신 시 생성된 SyncML 명령어를 Session Handler를 통하여 각 패키지 별로 구성하며, 수신 시에는 각 패키지 별로 DDF Tree handler를 통해서 트리를 변경시킨다. Session Handler는 각 세션을 관리하는 기능을 담당하며 DM Configuration

에 저장된 각 단말기의 정보를 이용하여 인증, 관리 그리고 통신을 관리 연산에 대한 통신을 담당한다. ThinkSync DM 에이전트로부터 수신된 관리 정보는 DM Information Analyzer에 의해 운영자에게 필요한 관리 정보로 가공되어 운영자에게 전달된다.



(그림 2) 관리 정보 구조

3.2 관리 정보 구조

그림 2는 소프트웨어를 관리하기 위해 필요한 관리 객체들을 트리 구조로 나타낸 DDF 스키마이다. 그림 2의 DDF 스키마는 현재 OMA DM에서 논의중인 규격을 기반으로 확장한 것이다. Inventory 엘리먼트는 기본 정보로서 Delivered와 Deployed인 하위 엘리먼트가 있다. Delivered는 매니저로부터 에이전트에게 전달된 패키지에 대한 정보만을 저장하며, Deployed는 전달된 패키지를 시스템에서 활성화 시킨 후의 정보를 저장한다. 전달된 패키지가 활성화 되었다면 Delivered에서 저장한 패키지의 정보를 삭제한 후 Deployed에 추가되어야만 한다. Download 엘리먼트는 패키지를 다운로드하기 위해 필요한 정보를 저장하며, URI에 의해서 식별되어야 한다. Operation 엘리먼트는 전달되었거나 현재 실행중인 패키지에 대한 현재의 상태를 변경하기 위해 사용한다. 예를 들어, 현재 패키지가 단지 에이전트에게 전달되어진 상태로만 존재하고 있다. 그 패키지에 대해서 InstallAndActivate 명령을 수행시킴으로서 전달된 패키지를 설치하고 활성화시킬 수가 있

다.

3.3 소프트웨어 관리 절차

매니저는 5가지 종류의 관리 세션을 통하여 에이전트가 설치된 무선단말기의 소프트웨어를 관리한다. Inventory 세션, Download 세션, Install 세션, Activation 세션 마지막으로 Test 세션이다.

Inventory 세션은 에이전트에게 이미 설치된 소프트웨어를 조사하여 설치하고자 하는 소프트웨어의 의존성, 버전등을 검사한다. 질의를 받은 에이전트는 매니저에게 질의에 대한 응답을 수행한다. Download 세션은 설치하고자 하는 소프트웨어에 대한 정보를 Download 엘리먼트에 추가 혹은 갱신하는 작업을 수행하며, 추가 혹은 갱신된 정보를 통해서 에이전트에게 실제 패키지 다운로드 명령을 전송한다. 에이전트는 Download 엘리먼트의 하위 노드의 정보를 통해서 패키지를 다운로드하게 된다. Install 세션은 실제 다운로드 된 패키지를 시스템에 설치하며, 설치완료 후 그 응답메시지를 매니저에게 전송한다. Activate 세션은 Install 세션을 통해 설치된 패키지를 활성화 시키는 작업을 수행한다.

활성화 완료 후 활성화 된 패키지에 대해서 필요시 Test 세션을 통하여 정상동작 여부를 확인한다.

[4] OMA, OMA Device Management Tree and Description Session v1.2, 2005

4. ThinkSync DM SoftMan 구현 결과

구현된 소프트웨어 배포 관리 매니저는 현재 리눅스의 패키지 관리 방법인 RPM을 기반으로 동작하고 있다. 소프트웨어 설치를 위한 패키지를 만드는 방법과 에이전트에서 소프트웨어 설치 및 갱신 그리고 의존성 검사등도 모두 RPM에서 제공하는 기능을 사용하고 있다. ThinkSync DM SoftMan 관리자는 아파치 웹 서버의 HTTP 프로토콜을 사용하여 명령어 교환을 한다. ThinkSync DM 엔진은 OMA DM을 구현한 결과로써 Message Handler, DDF Tree Handler, Transport Handler, Session Handler로 구성되며, C언어와 C++로 구현 되었다. 에이전트를 통해 수집된 정보를 MySQL 데이터베이스에 저장하고 데이터베이스의 연산으로 수집된 정보를 분석한다.

ThinkSync DM SoftMan은 각 세션별로 나누어 각각의 패키지 관리 연산별로 구현하였다. 각 세션을 수행할 때 소프트웨어를 설치할 에이전트, 패키지 및 각 세션에서 필요한 정보를 운영자에게 입력받은 후 이를 기반으로 관리 세션을 실행한다. 각 세션이 실행될 때 진행과정을 운용자가 확인할 수 있으며 실행 결과도 요약하여 운용자에게 제시된다.

5. 결론

본 논문에서는 OMA DM 기반의 소프트웨어 배포 관리자 설계 및 구현 결과를 제시하였다. 소프트웨어 배포 매니저의 전체적인 구조 및 정보 구조 그리고 처리 과정을 중점적으로 기술하였다. 설계된 소프트웨어 관리 시스템은 리눅스의 RPM을 기반으로 개발되었다.

참고문헌

- [1] Uwe Hansmann, Riku Mettala, Apratim Purakayastha, Peter Thompson, "SyncML Synchronizing and Managing Your Mobile Data", pp. 21-34, PRENTICE HALL PTR, New Jersey, 2003.
- [2] DaeJin Jang, Hong Taek Ju, KeeHyun Park, B.H.Ha, M.C.Lee, Sung-Chae Bae, "Design of ThinkSync DM based on SyncML Device Management," The 3rd APIS, pp. 569~574, 2004.
- [3] OMA, OMA Device Management Protocol v1.2, 2005