

KVM 기반의 모바일 XML 문서 전송 시스템

김은수*, 전석훈*, 송정길*

*한남대학교 컴퓨터공학과

e-mail:kimes@mail.hannam.ac.kr

Mobile XML Document Transmission System Based on KVM

Eun-Soo Kim*, Seock-Hun Cheon*, Jung-Gil Song*

*Dept of Computer Engineering, Han-Nam University

요 약

최근 인터넷과 무선통신기술의 융합으로 인한 Mobile 서비스의 급격한 발전으로 인해 무선 인터넷 시장이 도입 및 성장단계에 이르렀다. 그러나 현재 무선 인터넷 환경은 음성 서비스 위주의 개인사용자 시장이 한계에 도달하면서 이동통신의 무선 단말기를 통해 다양한 비즈니스를 벌일 수 있는 기업용 모바일 솔루션 시장이 크게 확대되고 있고 모바일 데이터서비스 시장의 비중이 크게 늘어가고 있다. 이러한 데이터서비스 교환에 XML 문서를 사용한다면 데이터의 상호 처리 능력을 향상시키고 데이터의 공용성과 프로그램의 유연성을 향상시키는데 필요한 비용과 노력을 훨씬 절감할 수 있다. 본 논문에서 제안된 시스템 설계를 통해 XML 문서를 데이터 교환에 사용하여 무선 장치의 CPU, 메모리, 네트워크 대역폭의 부하를 줄임으로써 무선 인터넷에서 보다 빠르고 효율적으로 문서를 주고 받을 수 있도록 구현하였다.

1. 서론

최근 들어 유선 인터넷 환경에서의 다양한 콘텐츠가 무선 인터넷 기반의 환경으로 이동하고 있다. 그 중에서도 현재 무선 인터넷 서비스 중에서 가장 인기 있는 아이템은 단연 캐릭터와 멜로디 다운로드 서비스와 단순한 무선 메시지 전송 서비스일 것이다.

그동안 모바일 통신사업자의 비즈니스 초점은 주로 개인사용자 시장(Consumer Market)에 맞춰져 왔다 해도 과언이 아니다. 하지만 음성 서비스 위주의 개인사용자 시장이 한계에 도달하면서 2002년에는 사용자당 월 매출액(ARPU: Average Revenue Per User)이 최저 수준을 기록하게 됐다. 사업 다각화의 필요성을 인식한 모바일 통신사업자들은 음성과 함께 기업용 데이터 및 부가 서비스를 시작했다. 기업들의 입장에서도 구축한 애플리케이션을 언제 어디서나 접속 가능하게 즉, 새로운 서비스를 사용함으로써 업무효율을 증대시킬수 있다는 점에서 이들 서비스를 선호했다. 양측의 이익이 맞아 떨어지면서 오는 2006년 기업용 모바일 데이터 서비스 비중은 크게 늘어 전체 모바일 서비스 시장의 25%에 이를 것으로 전망되고 있다[1].

그리고 소형기가 갖는 소량의 메모리, 저속의 CPU 한계는 고용량 데이터베이스(DB)를 고속으로 처리해야 하는 모바일 데이터서비스의 발목을 잡고 있다. 소형, 경량화라는 끊임없는 시도를 하고 있는 모바일 디바이스에서 제한된 메모리 활용에 대한 최대 이슈는 풍부한 메모리에 익숙한 사용자를 끌어들이

는데 있다. 따라서 제한된 메모리의 활용을 최대화하는 유일한 방법으로 다운로드와 삭제를 반복하는 구조를 채택하는 것이다.

이러한 문제점을 극복하고 단순 텍스트가 아닌 독립적인 실행구조를 갖는 애플리케이션으로 다운로드가 가능하고 오프라인상태 에서도 반복실행 가능한 환경인 VM(Virtual Machine)이 휴대폰의 제약을 극복하고 급변하는 통신표준과 사용자 요구를 빠르게 대응할 수 있는 유일한 열쇠가 될 것으로 전망되고 무선 데이터서비스에서 XML[2]을 이용하면 데이터의 공용성과 프로그램의 유연성을 향상시키는데 필요한 막대한 비용과 노력을 절감할 수 있다.

본 논문에서는 Mobile 환경에서 XML 문서를 전송하기 위해 JAVA의 JVM(Java Virture Machine) 및 API로 구성된 J2ME(JAVA 2 Micro Editon) 기술과 무선 애플리케이션용 XML 자체 개발 파서를 이용하여 XML 문서 데이터베이스로부터 파싱하여 직접 모바일(Mobile)환경으로 XML 문서를 전송하여 CPU의 부하를 덜어주고 무선 장치의 메모리, 네트워크 대역폭의 부하를 현저하게 줄일 수 있는 시스템을 설계 및 구현하였다.

본 논문의 구성은 2장에서는 시스템 설계에 필요한 관련연구를 살펴본 후 3장에서는 J2ME 기반의 모바일 XML 문서 전송 시스템 설계와 시스템에 대하여 논의한 후 4장에서는 실제 구현된 시스템에 대해 기술하고, 끝으로 5장에서 결론 및 향후 연구방향을 제시한다.

2 관련연구

2.1 무선단말기 플랫폼 기술 (Virture Machine)

무선단말기 브라우징(Browsing)을 지원하는 Markup Language, WAP과 ME 그리고 i-mode가 1세대 폰브라우징 기술이라면, VM은 사용자가 무선인터넷 서버에서 필요한 콘텐츠나 애플리케이션을 다운로드받아 무선단말기에서 구동할 수 있는 플랫폼 기술로 그 수준이 한단계 업그레이드된 것으로 규정할 수 있다. 즉, VM은 최근 보급대수가 급격히 증가하고 있는 컬러 휴대폰을 비롯한 25G 휴대폰의 다양한 무선인터넷 통신서비스를 지원하는 핵심 기반기술이다.

국내에서는 2000년 9월 LG 텔레콤이 썬마이크로시스템의 자바 플랫폼인 Kitty Hawk를 탑재한 자바폰(i-book)을 출시하면서 처음으로 무선플랫폼 시대를 열었다. 국내 기술로는 신지소프트의 GVM이 가장 먼저 시작했다. 현재 국내에서 상용화된 VM 기술은 데이터 스크립트를 다운로드받아 실행시켜주는 플러그인 형태와 API(Application Programming API)를 제공해 프로그램 실행파일 형태의 바이너리로 내려보내는 방식이 있다. 썬마이크로시스템의 자바(J2ME), 신지소프트의 GVM, XCE의 XVM(SKVM)은 전자의 방식이고, 모빌탑의 MAP과 켈컴의 BREW는 후자의 방식이다. 현재 국내 시장에서 자기 다른 5개의 VM 플랫폼 기술 표준이 난립해 있고 이에 따라 이동통신 사업자별로 자기 다른 플랫폼을 채택해 서비스를 실시하고 있다[3].

표 1은 관련연구과정에서 도출된 자료를 토대로 작성한 VM 기술들을 비교 분석한 자료이다.

표 1. VM 기술별 비교와 서비스 현황

플랫폼	개발사	개발언어	제공방식	채택업체	서비스개시일
KVM	썬마이크로시스템	J2ME	바이코드	LGT	2000.9
GVM	신지소프트	Mobile C	스크립트	SKT	2000.10
MAP	모빌탑	C언어	바이너리	KTF	2001.3
XVM (SKVM)	XCE	J2ME	스크립트 방식	SKT	2001.7
BREW	켈컴	C언어	바이너리	KTF	2001.9

2.2 J2ME(JAVA 2 Micro Edition)

썬마이크로시스템사는 다양한 내장형 시스템에 적합한 맞춤형 실행 환경을 제공하기 위한 Java Virtual Machine 및 API 세트에 구성되어 있고 많은 제조사들의 휴대폰과 다양한 하드웨어 구조를 가지는 모바일 기기에서 하나의 플랫폼으로 원활하게 동작하도록 만든 J2ME[4] 기술을 개발하였다. J2ME의 장점은 동적으로 애플리케이션을 다운로드 받을 수 있고 크로스-플랫폼 독립적이며, HTTP 1.1을 지원한다. 이 플랫폼은 KVM을 기본 가상 머신으로 내장한 CLDC(Connected Limited Device Configuration)[5] 기반의 MIDP(Mobile Information Device Profile)[6]로 구성되어 있다. 그림 1은 자바 플랫폼을 탑재한 무선 단말기의 전체 구조이다.

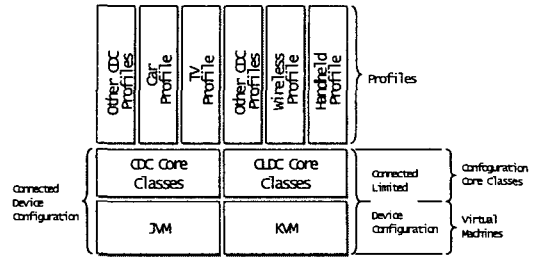


그림 1. J2ME Architecture

2.3 무선 애플리케이션용 경량급 XML 파서

MIDP 플랫폼에서 사용할 수 있는 소형 XML 파서들은 대부분 오픈소스로 배포되고 Size 칼럼은 압축된 클래스의 어렵값이다. 유형에서 pull은 문서 분석을 위해서 특정 메소드를 계속해서 호출해야 한다는 것을 뜻하고 push는 파서가 스스로 문서 전체를 파서하는 것을 말하고, 특별한 이벤트 발생시에는 콜백 메소드를 호출하면 된다. SAX-호환 파서들은 push 방법을 쓴다. 분석이 끝난후 이 모델을 검증해야 하며 각 원소 데이터를 추출해야 한다[7]. 본 시스템에서는 자체 개발한 무선 인터넷용 경량급 XML 파서를 이용하였다. 표 2는 관련연구과정에서 도출된 자료를 토대로 작성한 무선 인터넷용 경량급 XML 파서들을 비교 분석한 자료이다.

표 2 무선 인터넷용 XML 파서

파서	URL	크기	MIDP	유형	SAX
kXML	http://locml.enhydra.org/	34KB	yes	pull	no
MiniXML	http://www.wilson.co.uk/xml/minxml.htm	13KB	no	push	1.0
NanoXML	http://nanoxml.sourceforge.net/	10KB	patch	model	1.0
TinyXML	http://www.gibradunnnsrac.org/tiny	13KB	no	model	no
wbxml	http://www.trantor.de/wbxml	19KB	no	push	1.0

3. 시스템 설계

3.1 구체화된 시스템의 하드웨어 사양

우선적으로 MIDP에서 구체화된 하드웨어 디바이스들의 구성 요구사항을 살펴보면 다음과 같다.

- ① 디스플레이 요구사항
 - 96×54 이상의 Screen Size, 1비트 이상의 디스플레이
- ② 입출력 요구사항
 - 한손용 키보드 또는 두손용 키보드, 터치 스크린으로 입력가능
 - 작은 스크린 사이즈에 적용 가능하고 포인팅 device가 없는 단말기에 적용 가능
- ③ 메모리 요구사항
 - MIDP 컴포넌트를 위한 128K의 비휘발성 메모리 공간

- 애플리케이션이 생성하는 데이터의 저장을 위한 8K의 비휘발성 메모리 공간 및 자바 런타임을 위한 32K의 휘발성 메모리 공간

④ 네트워킹 요구사항

- 쌍방향이고, 무선이며 제한된 대역폭에 간헐적으로 연결될 수 있음

3.2 Java에서의 구현을 위한 유저 인터페이스 설계

MIDP의 유저 인터페이스는 javax.microedition.lcdui 패키지를 통해 제공되고 있다. MIDP는 스크린 기반의 디자인을 가지고 있는데, 스크린은 사용자 입력을 렌더링하는 특정 device 그래픽스를 캡슐화하는 객체라고 정의하고 있다. 다시 말하자면, 스크린이라는 것은 사용자에게 나타나는 화면이며, 사용자는 한번에 하나의 스크린 만을 볼 수 있고, 사용자 입력에 대한 처리는 이 스크린에 의해서 이루어진다고 할 수 있을 것이다. 그림 3에서는 두가지 종류의 스크린이 있다는 것을 알 수 있는데, Screen 텍스트 박스, 리스트 등의 상위 레벨 API를 위한 최상위 추상화를 제공하고 Canvas 클래스는 게임과 같은 하위 레벨 API를 사용하는 application을 위한 스크린을 제공한다. Screen 클래스를 상속하는 List, TextBox, Alert와 같은 클래스들은 사전에 정의된 구조를 사용할 수 있는 컴포넌트들이고, 이렇게 사전에 정의된 구조를 사용할 수 없는 TextField, ImageItem, Gauge 등의 클래스들은 Screen 클래스를 상속하는 Form 클래스를 상속하도록 설계되어 있다.

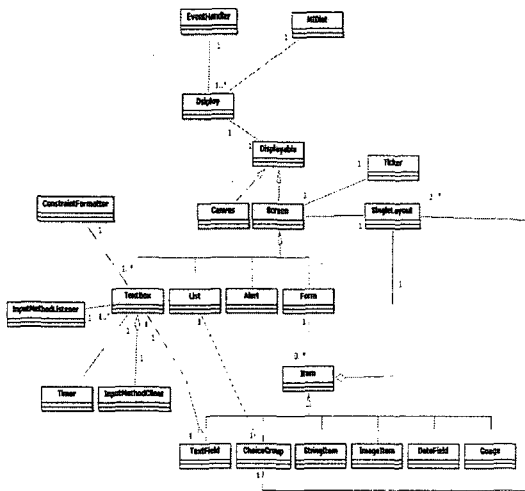


그림 2 LCDUI 클래스 다이어그램 설계

3.3 휴대폰 상의 XML 문서 디스플레이 흐름도

휴대폰 상에서 XML 문서는 SAX 파서를 통해 문서 전체를 한번에 분석한 후 파싱된 결과만을 디스플레이 되게 되는데 XML START Document View, XML END Document View, XML Document Error View로 디스플레이 되도록 구성하였고 XML 문서의 디스플레이 흐름도는 그림 3과 같다.

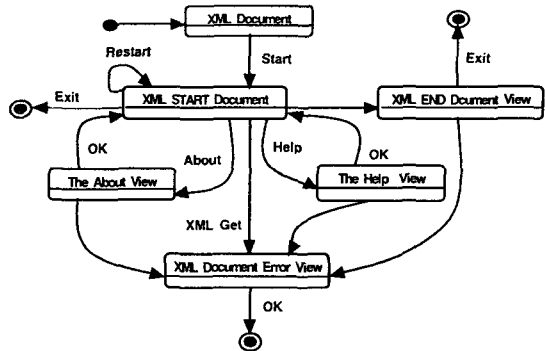


그림 3 XML 문서 디스플레이 흐름도

3.4 전체 시스템 흐름도

본 KVM 기반의 XML 문서 전송 시스템은 XML 문서 데이터베이스로부터 XML 문서 데이터를 파싱하여 클라이언트로 전송하는 모바일 XML 서버 모듈, 서버로부터 전송 받은 XML 문서 데이터를 단말기의 출력 창에 XML 문서 데이터를 출력하는 모바일 XML 클라이언트 모듈로 그림 4와 같이 구성된다.

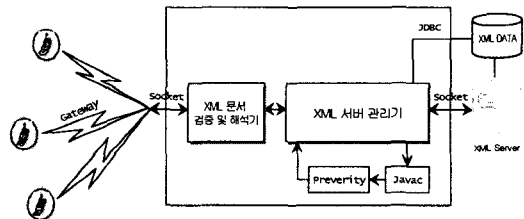


그림 4 무선인터넷으로의 XML 문서 전송시스템

4. 구현

본 논문에서 구현하고자 하는 시스템은 서버와 클라이언트 사이에서 단순히 웹 상에서 제공하는 정보가 아닌 시간과 장소에 따라 사용자가 필요로 할 수 있는 무선 인터넷 상에서 경량급의 그림 5와 같은 이력서 XML 문서를 전송할 수 있도록 구현하였다.

4.1 XML 문서

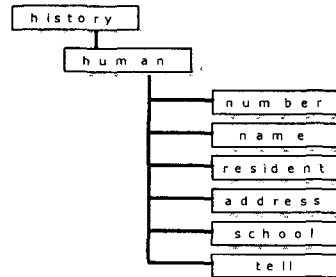


그림 5. 이력서 XML 문서 계층도

4.2 프로그래밍

무선 단말기 상에 XML 문서를 Display 되도록 Display() 메소드를 이용하여 XML 문서가 있는 웹 서버를 지정하여 XML 문서에서 읽어들이 데이터를 Vector에 저장한다.

```
public class MobileXml extends MIDlet implements CommandListener{
    private Display display;
    private String url = "http://syssw.hannam.ac.kr/~eskim/history.xml";
    static final String TITLE = "PERSONAL.HISTORY";
    List newList = new List (TITLE, Choice.IMPLICIT);
    TextBox tb = new TextBox ("", "", 256, TextField.ANY);
    Vector numberList=new Vector();
    .....
    HttpConnection hc = (HttpConnection)Connector.open(url);
    Reader parser = new XmlReader (new
    InputStreamReader(hc.openInputStream()));
    .....
    parser.require (Xml.START_TAG, null, "history");
    int i=0;
    while (parser.skip () != Xml.END_TAG) {
    .....

```

HttpConnection으로부터 주어진 XML이 있는 웹 서버에 연결을 하여 XmlReader를 통해 서버의 XML 문서를 읽어 들이게 된다. 읽어들이 XML 문서를 Xml.START_TAG()와 Xml.END_TAG() 메소드를 사용하여 XML 문서를 파싱하여 파싱한 토큰 중 한 토큰을 읽어들이 필요한 XML 문서의 엘리먼트 값을 각 Vector에 저장하게 된다.

```
public void getXml() throws IOException{
    try {
        HttpConnection hc = (HttpConnection)Connector.open(url);
        Reader parser = new XmlReader (new
        InputStreamReader(hc.openInputStream()));
        parser.relaxed = true;
        parser.skip ();
        parser.require (Xml.START_TAG, null, "history");
        int i=0;
        while (parser.skip () != Xml.END_TAG) {
            parser.require (Xml.START_TAG, null, "human");
            while (parser.skip () != Xml.END_TAG) {
                parser.require (Xml.START_TAG, null, null);
                String name = parser.getName ();
                parser.next();
                String text = parser.readText ();
                if(name.equals("number")) {numberList.addElement(text);i++;}
                else if(name.equals("name")) nameList.addElement(text);
                else if(name.equals("resident")) residentList.addElement(text);
            }
        }
    }
}

```

4.3 구현 결과

구현한 시스템은 그림 6과 같으며, PC와 무선 단말기는 하드웨어의 차이가 크기 때문에 실행 속도면에서는 많은 차이를 보일 수 있다. 따라서 반드시 에뮬레이터와 단말기 테스트를 거쳐야 한다. 본 논문에서는 서버와 클라이언트 사이에서 단순히 웹 상에서 제공하는 정보가 아닌 시간과 장소에 따라 사용자가 필요로 할 수 있는 데이터를 XML 문서화 하여 무선 인터넷 상에서 전송할 수 있도록 구현하였다.

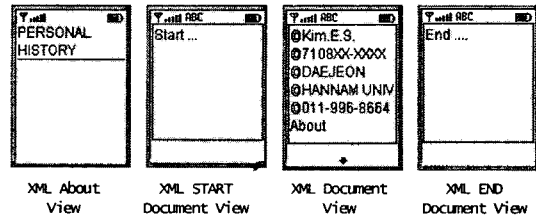


그림 6. XML 문서 데이터 무선인터넷으로의 전송 결과

5. 결론 및 향후 연구과제

본 논문에서는 휴대폰 등의 무선 단말기에 XML 문서를 전송하는 시스템을 설계하고 구현한 결과를 보았다. 본 논문에서 설계하고 구현하여 보인 시스템은 무선 인터넷 상에서 XML 문서를 사용하여 전송하는데 CPU의 부하를 덜어주고 무선 장치의 메모리, 네트워크 대역폭의 부하를 현저하게 줄일 수 있는 서비스를 가능하게 한다.

향후 연구되어야 할 내용은 다양한 모바일 데이터 서비스의 활성화를 위하여 지속적인 국내 관련 표준 확립과 내장형 경량 암호처리 기술 및 인증 기술(WPKI), 무선기반의 전송 계층 보안(WTLS) 및 종단간 보안 기술(Wireless End-to-End Security), 무선 신분 확인 기술(WIM) 등과 같은 보다 체계적이고 적극적인 모바일 비즈니스 정보보호 기술에 대한 연구개발이 이루어져야 할 것이고 XML 문서 전송시 무선 단말기의 전송 용량을 해결하기 위해 단순한 Signature, XSL 및 DTD가 완벽하게 지원되는 파서가 개발되어 많은 모바일 디바이스 어플리케이션에서 이용되어야 할 것이다.

참고문헌

- [1] W3C, "Extensible Markup Language", <http://www.w3.org/XML/>
- [2] 이타임즈인터넷, <http://www.etimesi.com/>
- [3] 소프트뱅크리서치, <http://www.sbresearch.co.kr/>
- [4] J2ME, "J2ME™ Platform", <http://java.sun.com/j2me/>
- [5] CLDC, "Connected Limited Device Configuration", <http://java.sun.com/products/cldc/>
- [6] MIDP, "Mobile Information Device Profile", <http://java.sun.com/products/midp/>
- [7] 공기식, 송문배 공역, "Wireless™ Java : Developing with Java™ 2, Micro Edition : 무선 자바 프로그래밍", 인포북, pp.178-187, 2002
- [8] 송정길, "Mobile 기반의 XML 문서 전송 시스템 설계 및 구현", 한국 인터넷 정보학회, '02 춘계 학술 발표회, 제3권 1호, 2002
- [9] 무선 인터넷백서편찬위원회, "무선인터넷 백서 2002", 소프트뱅크 미디어, 2002
- [10] Artem Garmash, "A Geographic XML-based Format for the Mobile Environment", Processing of the 34th Hawaii international Conference on System Science, 2001