

J2ME기반의 SK-VM을 이용한 다중 메시징 시스템 설계 및 구현

Design and Implementation of Multi Messaging System Using SK-VM Based On J2ME

김응곤* · 최완규** · 이성주*

* 조선대학교 전자계산학과

** 광주대학교 컴퓨터전자통신공학부

Eung Gon Kim · Wan Kyoo Choi · Sung Joo Lee

* Dept. Computer Science, Chosun Univ.

** Div.Of Computer, Electronics & Communication Engineering, Kwangju Univ.

(poledae@cspost.chosun.ac.kr)

ABSTRACT

In the case of mobile internet service using WAP, It was connected to http protocol using WAP Gateway. So, Users take incresed cost of mobile internet service. and it was generated inner security problem because it watched user information in the WAP Gateway.

To solve this problem we use java language. Which is independant of platform and low cost and intensely security and downloadable application. Additional, Using socket connection, Multi Messaging System(MMS) will connect real time between PC-Client and Mobile-Client, Mobile-Client and Mobile-Client, and so on.

In this paper, as design and implementation of multi messaging used SK-VM based on J2ME, It will be foundation to develop various mobile application in the future.

KeyWords : Mobile, J2ME, Messaging Service

1. 서 론

1960년대 두 네트워크사이의 안정성 있는 고속통신을 가능하게 하는 TCP/IP라는 통신규약이 만들어지면서 시작된 인터넷은 현재에 이르러 무선인터넷이라는 새로운 개념의 이동 통신 서비스가 등장하게 되었다.

우리 나라에서도 이동전화 사용자들이 초기에는 조금씩 늘어나기 시작했지만, 일정한 시점에 도달하자 폭발적으로 증가하기 시작했고, 이제는 거의 대부분의 사람들이 이동전화를 사용하고 있다. 다양한 무선인터넷 서비스를 받고자 하는 사용자들의 욕구가 늘어나면서 무선인터넷 서비스 비즈니스 모델 또한 다양하게 발전하고 있다.

기존의 WAP 무선인터넷 서비스 경우, http 프로토콜을 직접 이용하지 못하고 WAP Gate

way를 통하여 http프로토콜과 연동할 수 있었다. 이로 인해 사용자의 무선인터넷 서비스 비용 증가 및 WAP Gateway에서 사용자의 정보를 관찰할 수 있음으로 인해 발생하는 내적 보안의 취약성이 대두되었다.

그리하여, 선 마이크로 시스템즈에서는 작은 크기의 가상머신(KVM)을 무선 단말기에 탑재하여 무선 어플리케이션을 실행할 수 있도록 무선 단말기의 하드웨어적인 컨피규레이션과 이 컨피규레이션을 바탕으로 각 디바이스들의 기능, 시장의 요구사항에 맞추어 프로파일을 정의함으로써, 플랫폼의 통일성과 다양성 및 보안성까지도 만족시킬 수 있는 J2ME라는 플랫폼이 등장하게 되었다[1].

현재 이동 통신 사업자 중 SK-Telecom에서 개발 진행 중에 있는 SK-VM 같은 경우 Socket 통신이 가능하게 되어 있다. 따라서, 본

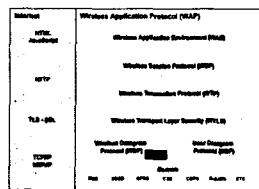
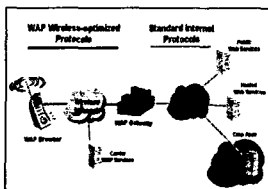
논문에서는 J2ME를 기반으로 Socket 통신을 사용할 수 있는 SK-VM을 이용한 다중 메시징 시스템(이하 MMS)을 설계하고 구현함으로써 향후 개발되어질 무선인터넷 어플리케이션의 모체가 되고, IMT-2000관련 어플리케이션 개발 시 화상 통신뿐 아니라 다양한 어플리케이션에 확장이 용이하도록 무선 대 무선, 무선 대 유선, 유선 대 유선의 각각 쌍방향통신이 가능한 MMS를 설계하고 구현하였다. 또한 WAP과의 비교 분석을 통하여 향후 전개되어질 무선 인터넷 발전 방향에 대하여 제시하였다.

II. 무선인터넷 기술

2.1 WAP

WAP의 가장 큰 특징은 기존의 인터넷 환경을 그대로 수용하면서 무선환경에 최적화하는데 중점을 두었다는 점이다. [그림1]의 WAP 아키텍처를 보면, WAP 게이트웨이가 인터넷 망과 무선망과의 프로토콜 변환을 통한 망간의 연동을 통해 두 망을 이어주고 있으며, 무선망에서는 단말기에 탑재된 WAP 브라우저가 CDMA, CDPD, GSM을 비롯한 무선망의 종류와는 독립적인 WDP/WTLS/WTP/WSP로 구성된 WAP 프로토콜을 통해 WAP 게이트웨이와의 통신을 수행하도록 되어 있다[그림2][2].

무선인터넷 서비스 개발자는 WAE 계층에서 정의된 WML(Wireless Markup Language), WMLScript를 통해 무선 애플리케이션을 개발할 수 있다. WML(Wireless Markup Language)은 차세대 인터넷 언어의 표준으로 각광받고 있는 XML(extensible Markup Language) 언어에 기반을 두고 있으며, WMLScript는 JavaScript 언어를 기반으로 두고 있으므로, 웹 개발자는 기존의 웹 개발 경험을 쉽게 무선 애플리케이션 개발에 적용할 있다. 그리고, CGI(Common Gateway Interface)나 ASP(Active Server Page)와 같은 동적인 페이지 생성 기술도 그대로 적용할 수 있다.

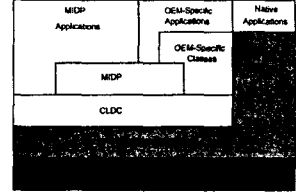
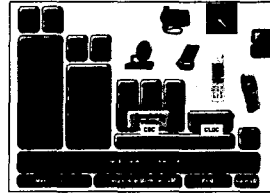


[그림1]WAP아키텍처 [그림2]WAP프로토콜

2.2 무선 자바 플랫폼 : J2ME(CLDC/MIDP)

자바 플랫폼은 크게 두 가지 요소로 구성된다. 자바 가상머신과 표준 API 집합이 그 두 가지 요소이다. 이러한 자바 플랫폼은 자바2플랫폼으로 진화하면서 J2SE, J2EE, J2ME라는 세 가지의 영역으로 분할되었다. 이것은 각각

데스크탑 시장, 엔터프라이즈 서버 시장, 소비자/임베디드 디바이스 시장을 목표로 한 역할 분담이다[그림3].



[그림3]자바2 플랫폼 [그림4]J2ME 아키텍처

2.3 CLDC/MIDP 아키텍처

[그림4]에서 모바일 인포메이션 디바이스(MID) 위에 실시간 운영체제(RTOS)가 올라가고, 그 위에 KVM을 포함한 CLDC가 위치하고, CLDC 상위에 MIDP가 배치되는 것이다. 그리고, MIDP 어플리케이션은 MIDP와 CLDC의 코어 API를 이용한 응용 프로그램으로 작성할 수 있다[1]. JAM은 이 아키텍처 상에서는 CLDC 계층의 네이티브 시스템 소프트웨어의 영역에 존재한다고 보아도 무방할 것이다.

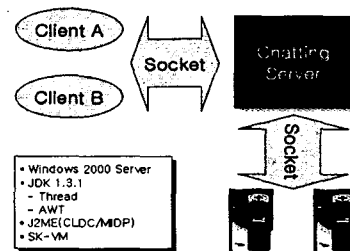
[그림4]에서 주의 깊게 보아야 할 것은 OEM-Specific 클래스와 OEM-Specific 어플리케이션들이다. OEM 기반의 비표준 API와 어플리케이션의 역할을 무시할 수 없고, 이는 CLDC 상위에서 이루어진다. 예를 들어, SMS와 같은 메시징 서비스와, 위치 기반 서비스와 같은 무선 네트워크의 특성 및 단말기의 독특한 특성을 이용할 수 있는 서비스는 OEM 기반의 API로 플랫폼에 추가되고, 그 API를 이용해서 서비스를 할 수 있다[3].

III. MMS 설계 및 구현

3.1 시스템 구성

1) 개요

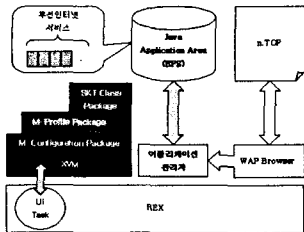
본 시스템의 시나리오는 기존의 유선 상에서의 메시징 시스템을 J2ME를 이용하여 유선과 유선, 유선과 무선, 무선과 무선사이의 메시징 시스템을 설계하고 구현하였다. 특히 WAP Gateway를 이용하지 아니하고, 무선단말기상에서 직접 메시징 서버에 접속하는 TCP/IP를 이용한 Socket 통신 메시징 시스템이다.



[그림5] 시스템 개요

2) SK-VM 플랫폼

SK-VM은 SK Telecom에서 clean room 기반으로 자체 개발한 J2ME 자바 실행 환경으로 가상머신 및 단말 확장 UI, 네트워크, IO를 포함하는 클래스 라이브러리로 구성되어 있으며, 무선이동단말기에서 응용프로그램을 다운로드 및 실행시킬 수 있는 환경을 제공한다. SK-VM 플랫폼에서는 CLDC와 호환되는 M-Configuration(가칭)과 MIDP와 호환되는 M-Profile(가칭), 그리고 SK Telecom에서 정의한 OEM-Specific Class (SKT-Profile)의 세 가지 API를 제공한다[4].



[그림6] SK-VM 플랫폼

3.2 시스템 설계

1) PC용 클라이언트

본 논문에서는 시스템의 실시간 메시지 서비스를 위하여 다음과 같이 설정하였다.

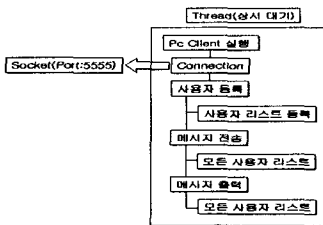
첫 번째, 사용자가 사용하는 컴퓨터에 JDK 1.1 이상만 설치되어 있으면 PC용 클라이언트 파일을 다운받아 직접 실행할 수 있다.

두 번째, 실시간으로 메시지 전송이 이루어져야 한다.

세 번째, 각 PC용 클라이언트 및 Mobile용 클라이언트에서 전송하는 메시지는 정확하게 전송되어야 한다.

네 번째, 사용자들이 좀 더 쉽게 사용하기 위한 환경(AWT)를 제공한다.

이러한 기능을 수행하기 위하여 PC용 클라이언트를 자바언어와 자바 AWT를 이용하여 GUI를 구성하였고, Thread를 이용하여 실시간 메시징 서비스를 할 수 있도록 구성하였다.



[그림7] PC Client 흐름도

2) Mobile용 클라이언트

본 논문에서는 시스템의 실시간 메시징 서비스를 위하여 다음과 같이 설정하였다.

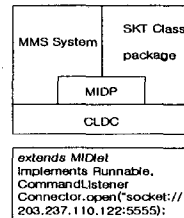
첫 번째, 사용자가 J2ME 플랫폼중 CLDC가 지원하는 무선단말기에서는 SK-VM을 이용하여 사용할 수 있어야 한다.

두 번째, 실시간으로 메시지 전송이 이루어져야 한다.

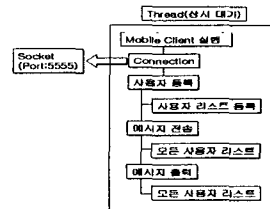
세 번째, 각 PC용 클라이언트 및 Mobile용 클라이언트에서 전송하는 메시지는 정확하게 전송되어야 한다.

네 번째, 무선단말기에서도 실시간으로 어떤 사용자가 입장하고 퇴장하는 지 알아야 한다.

이러한 기능을 수행하기 위하여 Mobile용 클라이언트를 자바언어를 이용하여 J2ME-Spec에 맞게 Mobile용 클라이언트 어플리케이션을 구성하였고, Thread를 이용한 실시간 메시징 서비스를 할 수 있도록 구성하였다.



[그림8] Mobile Client 구성도



[그림9] Mobile Client 흐름도

3) MMS 서버

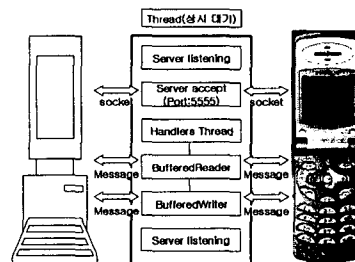
본 논문에서는 시스템의 실시간 메시징 서비스를 위하여 다음과 같이 설정하였다.

첫 번째, PC용 클라이언트 및 Mobile용 클라이언트 모두 지원하여야 한다.

두 번째, 실시간으로 메시지 전송이 이루어져야 한다.

세 번째, 각 PC용 클라이언트 및 Mobile용 클라이언트에서 전송하는 메시지는 정확하게 전송되어야 한다.

이러한 기능을 수행하기 위하여 메시징 서버를 자바언어를 이용하여 구성하였고, Thread를 이용한 실시간 메시징 서비스를 할 수 있도록 구성하였다. 서버 측에서는 객체지향의 장점을 살리고자 데이터만을 전송해 주는 클래스와 메시징 서버를 사용하는 사용자들을 관리하기 위하여 Thread로 구현한 클래스로 구성하였다.



[그림10] Server 흐름도

IV. 구현 방법

4.1 구현 방법

1) 개요

본 시스템을 위해 인터넷상의 여러 최신 기술을 무선인터넷 서비스와 접합시킴으로서 그 효율가치를 높였다. 현재 본 시스템에서 사용하고 있는 기술들을 열거하면 다음과 같다.

1. 자바 언어 : 인터넷상에서의 클라이언트/서버 통신설정을 위해 사용한다.

2. J2ME : 무선인터넷 서비스를 사용할 수 있도록 한다.

3. SK-VM : 실제 Socket 통신을 처리 할 수 있도록 하는 애플레이터

이러한 기술들을 사용하여 각각을 구현함에 있어 핵심 이슈가 되는 사항으로는 각각의 무선택말기 및 메시징 서버를 구현함에 있어서 재사용이 가능한 클래스의 형태로 만드는 것, 무선인터넷 서비스를 하는 데 있어서 Socket 통신을 통하여 실시간 유·무선 메시징 서비스를 즐길 수 있다는 것, 현재 진행되고 있는 J2ME 플랫폼과 관련된 CLDC/MIDP Spec을 이해하고 활용할 수 있다는 것 등이 있다.

2) 인터넷 통신

인터넷 통신의 기본은 클라이언트 서버 모델의 소켓통신이다. 자바 언어에서 이러한 소켓 통신은 java.net package의 Socket 클래스와 ServerSocket 클래스에서 상속을 받아서 구현한다. Socket 클래스는 클라이언트 소켓에 필요한 여러 메소드들을 정의한 자바 API의 일부분이며, ServerSocket 클래스는 자바 어플리케이션 서버의 기능을 모아 놓은 자바 API중 하나이다.

a. 서버의 구현

서버의 주요 기능은 다중 접속자를 처리할 수 있어야 하는 것이다. 이는 자바 언어의 Thread 클래스로부터 상속을 받은 Handlers thread 클래스를 정의하여 구현하였다. 이 클래스는 한 클라이언트 접속 시에 각각 하나씩 instantiate를 하여 각 클라이언트들을 다루도록 하였다.

Handlers thread = 접속한 클라이언트들의 정보 수집 및 실제 데이터의 전송

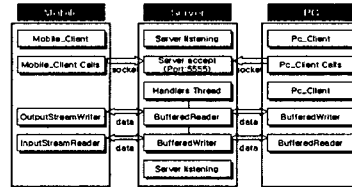
또 다른 기능은 다중 접속자들이 전송하는 데이터에 대하여 일관성 있는 데이터의 전송이 필요하다. 이것은 자바언어의 synchronized라는 키워드를 사용하여 Data 클래스를 정의하고 구현하였다.

Data = 다중 접속자들에게 데이터의 일관성을 가지고 데이터 전송

b. 클라이언트의 구현

클라이언트는 지속적인 서버의 변화를 지켜

볼 수 있어야 한다. 이러한 지속적인 변화를 관찰, 표현하기 위해 implements Runnable를 수행한다. PC용 클라이언트와 Mobile용 클라이언트 모두 implements Runnable를 수행함으로써, 다중 상속을 받을 수가 있다. Mobile 클라이언트와 PC 클라이언트와 서버간 양쪽의 데이터 흐름을 아래 [그림11]에 나타내었다.

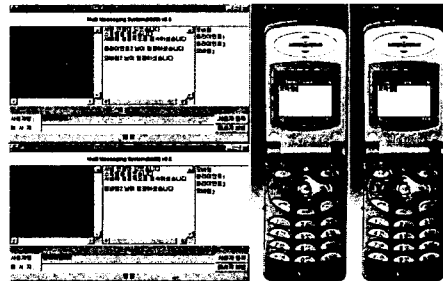


[그림11] 인터넷 통신 과정

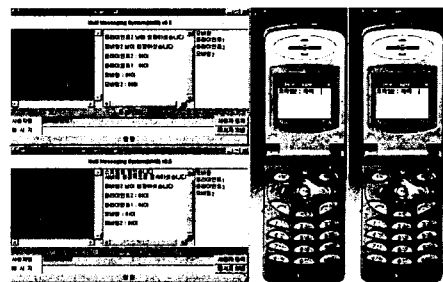
4.2 구현

1) 클라이언트의 구현

PC용 클라이언트는 연결 버튼을 눌러 메시징 서버에 접속하여 현재 사용자가 누구인지 확인하고 유선 또는 무선에 있는 모든 사용자와 메시징 서비스를 할 수 있도록 하였다. 따라서 무선 단말기를 가지고 있는 사용자가 언제, 어디서든지 메시징 서버에 접속만 하면 현재 메시징 시스템을 이용하고 있는 사람과 직접 메시지를 주고받을 수 있다.



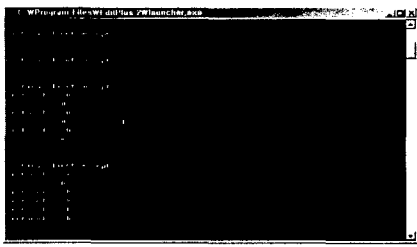
[그림12] 클라이언트 (사용자 등록)



[그림13] 클라이언트(메시지 전송)

2) 서버 구현

서버는 아래 [그림14]처럼 간단한 콘솔 어플리케이션으로 제작하였다. Client사용자의 소켓과 스트리밍이 연결되는지 연결 안되는지를 확인하고, 어떤 사용자가 어떠한 이름으로 접속했는지를 알 수 있도록 하였다.



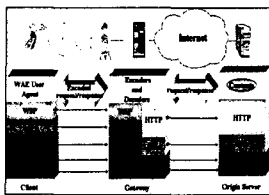
[그림14] 서버 APP Consol

3) WAP과의 비교 분석

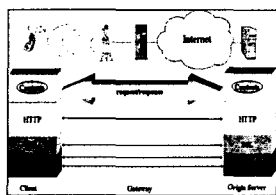
기존의 WAP 모델에서는 무선단말기(Client)와 인터넷 서버 사이에 WAP Proxy라 불리는 WAP Gateway를 두도록 하고 있다. WAP Gateway의 주요 역할은 WAP 프로토콜과 인터넷 TCP/IP 프로토콜을 중간에서 변환해 주는 것이다. 즉, 모든 무선단말기의 인터넷 서비스 요구는 WAP Gateway를 거치도록 되어 있고, 다시 서비스를 요청한다.

이러는 과정 중 WAP Gateway를 보유하고 상용화 서비스를 하는 무선인터넷 서비스 사업자들은 데이터의 이동 및 전송 경로 등을 파악할 수 있다. 이로 인해 사용자들의 내적 보안이 이루어 질 수 없다[그림15].

이러한 문제를 해결하기 위하여 J2ME를 이용한 무선 서비스를 할 수 있다. 즉, WAP Gateway를 이용하지 않고 무선단말기와 인터넷 서버와의 통신을 TCP/IP를 이용하여 직접 통신 할 수 있다[그림16]. 또한, J2ME를 이용할 경우 무선인터넷 서비스 어플리케이션을 개발하고 배포하기 위한 클래스 파일을 생성시 사전검증기(preverifier)를 통하여 검증단계를 거치고 무선단말기에서는 이러한 사전검증기를 통한 어플리케이션을 다운로드 받기 전 다시 한번 사전검증기를 거침으로 보안성의 강화 및 메모리 절약과 속도 향상을 가질 수 있다.



[그림15]WAP아키텍처



[그림16]J2ME아키텍처

그리고, WAP을 이용한 무선인터넷 서비스를 제공하는 경우 사용자들이 유선 인터넷 서버에 접속을 지속적으로 연결을 설정해야만 한다. 그러므로, 사용자들의 무선인터넷 비용이 증가하게 된다. J2ME를 이용할 경우, 무선단말기 자체에서 어플리케이션을 다운로드 받을 때 유선인터넷 서버에 접속하고, 다운 받은 어플리케이션의 정보를 자체적으로 무선단말기에 저장시킨 후 중요한 정보나 데이터를 유선 인

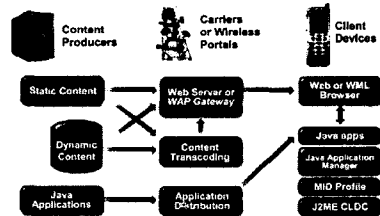
터넷 서버에 접속하여 저장하면 된다. 그러므로, 무선인터넷 서비스를 저렴하게 즐길 수 있다. 또한 무선사업자의 경우 비용이 많이 드는 WAP Gateway를 사용하지 않음으로 인한 비용 감소가 있을 수 있다.

V. 결 론

본 논문은 J2ME를 이용한 MMS를 설계하고 구현함으로써 향후 개발되어질 무선인터넷 어플리케이션의 토대가 된다. Socket 통신 및 http 통신을 이용하여 무선단말기에서의 데이터 저장 및 무선인터넷 서비스 서버로의 데이터 전송 등의 기반이 될 수 있다.

또한, J2ME를 기반으로 자바언어를 사용함으로써 플랫폼에 독립적이며, 또한 TCP /IP기반의 Socket 통신을 이용한 다중 메시징 시스템을 설계하고 구현함으로써 향후 IMT-2000 관련 어플리케이션 개발 시 화상 통신뿐 아니라 다양한 어플리케이션에 확장이 용이하다.

하지만 J2ME를 이용한 무선인터넷 서비스만이 모든 것을 해결해 줄 수는 없다. 왜냐하면, 현재 무선인터넷의 표준으로 자리 잡고 있는 WAP의 진영과 WAP을 배제하자는 Anti-WAP 진영이 무선인터넷 서비스에서 상당한 위치를 차지하고 있다. 그러므로, WAP 및 Anti-WAP, 그리고 J2ME의 진영이 상호 호환적인 부분으로 나아가야 할 것이다.



[그림17]무선인터넷 서비스모델

VI. 참고문헌

- [1] J2ME 홈페이지, <http://java.sun.com/j2me>
- [2] WAP 포럼 홈페이지, <http://www.wapforum.org/>
- [3] Yu Feng and Dr. Jun Zhu, 자바누리 김용범 역 "Wireless Java Programming with J2ME", 정보문화사, 2001, pp. 33-34
- [4] XCE Homepage, <http://www.xce.co.kr>
- [5] CLDC 홈페이지, <http://java.sun.com/products/cldc>
- [6] MIDP 홈페이지, <http://java.sun.com/products/midp>
- [7] SK-VM 개발자 페이지, <http://developer.n-top.com/main.asp>