

KANNEL WAP게이트웨이와 연동하는 WAP브라우저의 설계 및 구현

진 민 식*, 황 옥 철*, 임 동 기*, 최 원 호*, 정 민 수*
*경남대학교 컴퓨터공학과

Design and Implementation of WAP Browser Interworking Kannel WAP Gateway

Min-Sik Jin*, Ook-Chul Hwang*, Dong-Ki Lim*, Won-Ho Choi*, Min-Soo Jung*
*Dept. of Computer Engineering, Kyungnam University.
E-mail : comsta6@hawk.com.kyungnam.ac.kr

요 약

요즘 작고 빠른 장치에 관한 새로운 열풍이 불고 있다. 이들은 이동 중에도 정보에 접근할 수 있고, 웹과 이동 전화기의 개념을 합쳐 놓은 장치라고 할 수 있다. 무선 애플리케이션 프로토콜(Wireless Application Protocol:WAP)은 산업 전반적인 표준으로서 인터넷 콘텐츠와 서비스를 이동 전화기에서 접속할 수 있도록 해주는 통신 프로토콜과 애플리케이션 환경을 정의한다.

본 논문에서는 오픈소스로 진행되고 있는 Kannel Wap Gateway를 이용하는 WAP브라우저를 설계 및 구현 하였다. 임베디드 리눅스가 탑재된 모든 단말기에서 동작할 수 있으며 무선 인터넷을 통해 WAP사이트에 접속하고 콘텐츠를 디스플레이 할 수 있도록 해준다.

1. 서론

월드와이드웹(WWW)으로 상징되는 인터넷은 이미 사회, 경제, 정치적인 측면에서 인류의 삶에 새로운 패러다임의 변화를 촉진하고 있다. 이동 통신 기술의 발달과 더불어 시공간을 초월한 인터넷 접근이라는 모토아래 수많은 기업들이 무선 인터넷이라는 새로운 기술과 기회에 대한 도전에 직면해 있다.

이러한 기술의 한 예로 모바일 인터넷 환경과 모바일 단말기는 현재의 데스크톱 기준으로 접근하기에는 아직 전력 소모량, 메모리 크기, 디스플레이 크

기, 전송 속도, 안정성 등에서 많은 어려움이 있다. 따라서 유선 인터넷의 표준을 그대로 사용하는데 한계가 있으므로, 기존 표준을 가능하면 따르면서 무선 환경에 적합한 프로토콜을 만들려는 움직임이 일어나게 되었다. 이러한 무선 인터넷의 제약사항과 시대적 조류에 편승하여 많은 솔루션들이 시장에 출현하였으며 대표적인 것으로 마이크로소프트사의 mHTML, Phone.com사의 HDML, 일본의 NTT DoCoMo사의 cHTML, WAP포럼의 WML등이 있다. 이 중 WAP 포럼의 WAP(Wireless Application Protocol)표준은 전 세계의 폭 넓은 지지를 등에 업고 무선 인터넷 분야의

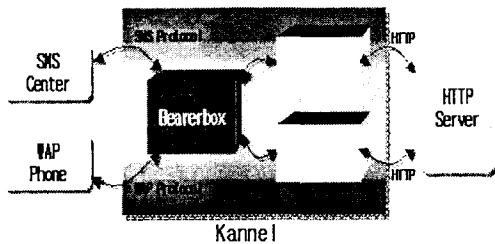
HTTP요청(Request)으로 바꾸어 해당 WAP사이트로 전송한다.

Step4: 해당 WAP사이트는 WAP게이트웨이로부터 받은 HTTP요청에 대한 응답(Response)을 HTTP프로토콜을 이용하여 WAP게이트웨이로 보낸다.

Step5: WAP게이트웨이는 WAP사이트로부터 HTTP응답(Response)을WAP응답을 바꾸고 데이터를 인코딩(Encoding)하여 사용자의 무선 단말기로 전송한다.

Step6: WAP클라이언트를 가지고 있는 무선 단말기는 WAP게이트웨이로부터 받은 WAP응답을 해석하고 수신한 데이터를 디코딩(Decoding)하여 사용자의 무선 단말기 화면에 출력한다.

본 연구에서는 무선 인터넷(WAP)과 유선 인터넷을 연결시켜 주는 역할을 하는 게이트웨이로서 Kannel WAP게이트웨이를 사용하였다.



<그림 3> Kannel WAP게이트웨이의 구성

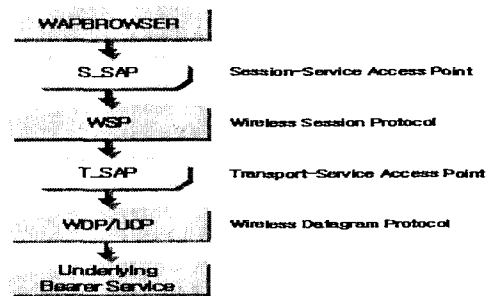
3. 이동 단말기용 WAP브라우저 설계

3.1 WAP스택 클라이언트의 설계

WAP클라이언트를 구성함에 있어 WAP이 가지고 있는 모든 프로토콜을 전부 구현해야만 WAP클라이언트로서 사용이 가능한 것은 아니며 사용할 무선 네트워크 베어러의 특징 및 WAP어플리케이션의 사용목적에 따라 다양한 구성이 가능하다.

본 연구에서는 Connectionless 모드라고 할 수 있는 비접속 지향 방식의 클라이언트를 설계하고 구현한다. 즉, WDP레이어가 WSP레이어의 아래에 있고 데이터 송수신은 단지 클라이언트의 요청에 의해서만 이루어진다. 접속을 유지할 필요가 없으므로 경량의 데이터 트랜잭션을 관장하는 프로토콜인 WTP레이어가 존재하지 않는다.

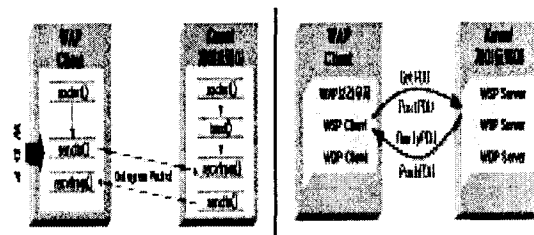
구현할 WAP스택은 단지 IPv4 Bearer위에서 Connectionless Mode를 지원한다. 중요한 문제는 다양한 다른 Bearer를 지원하기 위해 결국엔 ConnectedOriented Mode를 지원하도록 발전시켜야 한다. WSP의 주요한 역할은 클라이언트와 서버사이의 요청과 응답의encoding/decoding이다. WSP레이어는 HTTP의 모든 메소드와 Content-Type, Date와 같은 중요하다고 생각되는 HTTP/1.1 헤더의 인코딩과 디코딩을 지원하도록 설계하였다.



<그림 4>비접속 지향 방식 클라이언트의 WAP스택

WAP클라이언트와 게이트웨이의 WSP계층에서는 요청의 형태에 따라 4종류의 PDU(Protocol Data Unit)가 존재한다. 클라이언트의 요청은 Get, Post에 따라서 GetPDU, PostPDU가 있으며 게이트웨이에서는 요청에 대한 응답인 ReplyPDU, 서버로부터의 일방적인 데이터 전달인 PushPDU가 있다.

WSP아래에 있는 WDP는 Bearer와 통신을 담당하며 현재 IPv4만을 지원하며 WDP는 UDP소켓을 연결하고 사용자의 요청을 전달 게이트웨이에 전한다.



<그림 5> 설계된 WDP와 WSP 레이어

3.2 WAP브라우저의 설계

WAP브라우저는 게이트웨이로부터 전송 받은 WML인

