

主題

휴대용 무선장비에서의 웹 접근을 지원하는 무선 응용 프로토콜 기술동향

세종대학교 신동일, 신동규, 이경근

차 례

1. 서론
2. WAP(Wireless Application Protocol) 모델
3. HDML(Handheld Device Markup Language)의 개요
4. WML/HDML 저작도구 구현기술
5. 결론

1. 서론

WWW(World Wide Web)과 이와 관련된 기술의 발달로 인해서 사용자는 언제 어디서나 개인용 컴퓨터와 네트워크 연결이 있는 곳이면 자신이 원하는 정보를 손쉽게 취득할 수 있게 되었다. 1992년 말부터 인터넷상에 혜성처럼 등장한 웹(Web)은 급속한 기술의 발전과 함께 상업화, 대중화, 멀티미디어화 되고 있으며 가히 인터넷 세상이라고 할 정도로 발전하고 있다. 현재 웹에서 정보를 얻을 수 있는 장비로 쓰이는 개인용 컴퓨터들은 통신만을 위해서 이용하기에는 기능이 복잡하고 비싸고 또한 부피가 크며 무거우므로, 소형 휴대용 장비를 이용해서 다양한 정보를 취득하고자 하는 요구가 점점 증가되고 있다. 이에 따라 특히 휴대용 전화기나 PDA(Personal Digital Assistant)를 이용한 웹 정보에 대한 접근이 점차 절실히 요구되어가고

있다. 현재 이동 통신의 편리함으로 인하여, 이동통신망에 가입하여 휴대용 전화기나 PDA를 이용하는 사용자의 수는 국내에서만 1800만명을 넘었으며 앞으로 개인용 컴퓨터 사용자보다 더 많은 수의 사용자가 이를 이용하게 될 것으로 보인다.

이러한 휴대전화 및 관련 휴대용 무선장비를 통한 웹 콘텐츠 이용자의 확대는 데이터통신의 급격한 증가를 가져올 것이고, 엄청난 국내외적인 수요를 불러일으킬 전망이다. 휴대용 무선장비를 위한 웹 콘텐츠 제공 서비스는 현행 휴대전화 서비스뿐 아니라 차세대서비스로 일컬어지는 IMT2000에서와 같은 멀티미디어를 중심으로하는 통신서비스에서는 필수적인 기능이라 하겠다.

기존의 웹 기반 응용 업무를 지원하기 위한 언어인 HTML(Hyper Text Markup Language)로서는 제한된 제한을 가진 휴대용 무선장비에서 많은 정보를 효율적으로 나타낼 수 없다. 이에 대한 대

안으로 휴대용 무선장비에서의 효과적인 웹 접근을 위한 언어인 HDML (Handheld Device Markup Language)[1]이 Unwired Planet (현재 Phone.Com으로 회사명 변경)에 의하여 개발되었고, 1997년 5월 W3C(World Wide Web Consortium)에 휴대용 무선장비의 웹(Web)접근에 관한 표준안으로 제출되었다. 현재 휴대용 무선장비에서의 웹 접근을 위한 표준제정을 위하여 Ericsson, Motorola, Nokia 및 Unwired Planet등이 WAP (Wireless Application Protocol) Forum을 형성하여 HDML을 기반으로 하는 WML(Wireless Markup Language) [2]을 포함하는 무선 응용 프로토콜(WAP)의 표준 제정을 추진하고 있다. WML은 HDML의 기본구조인 데크와 카드(Deck/Card) 단위의 작동개념을 채택하여 데크와 카드들간의 하이퍼링크(hyper link)등 HDML의 작동원리를 그대로 승계하고 있으며 현재 WML 형식 1.1 버전이 발표되었다. 상용화를 위한 WML/HDML 응용프로그램의 제작이 전 세계적으로 급속도로 이루어지고 있는 상황에서, WML/HDML에 관련된 기술의 습득 및 WAP에 관한 연구는 향후 다가올 정보화 사회의 구축을 위하여 매우 중요하다고 할 수 있다.

본고에서는 휴대용 무선장비에서의 웹 접근을 가능하게 하는 무선 응용 프로토콜인 WAP(Wireless Application Protocol)에 대한 개념과 구조에 대해 설명하고, 데스크탑 컴퓨터에서, 웹 접근용으로 사용하는 HTML과 같은 무선 데이터 접근용 언어인 WML(Wireless Markup Language)/HDML(Handheld Device Markup Language)에 대해 기술하며, 현재 사용되고 있는 HDML에 대해서는 문법 및 사용환경에 대한 상세한 설명과 함께 HDML/WML 전용저작도구의 구현기술에 대하여 소개한다.

2. WAP(Wireless Application Protocol) 모델

WAP(Wireless Application Protocol) [3]은 네트워크 기술과 무선 데이터 기술 및 인터넷의 빠른 발전으로 생겨난 프로토콜이다. WAP의 전반적인 구조는 현재 이용되고 있는 웹을 기반으로 하여 만들어 졌기 때문에 이와 매우 깊은 관련을 가지고 있으며, 많은 부분에서 유사한 면을 볼 수 있다. 웹 구조는 그림 1에서와 같이 매우 유연하고 강력한

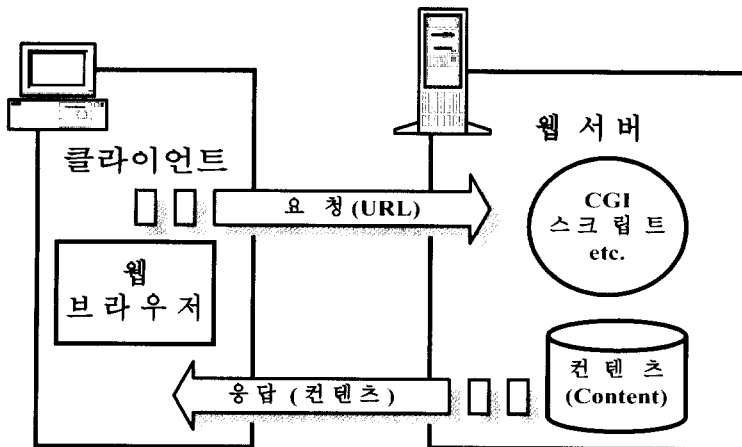


그림 1. 인터넷 웹 프로그래밍 모델

프로그래밍 모델을 제공한다. 콘텐츠(contents)는 표준 데이터 포맷인 HTML로 표현되며 웹 브라우저로 작업을 처리한다. 클라이언트에서 사용자가 원하는 데이터 객체를 웹서버에 요청하면, 웹서버는 표준 포맷을 사용하여 코드화된 데이터를 가지고 클라이언트에 응답하게 된다.

WAP 프로그래밍 모델은 그림 2에서와 같이 인터넷 웹 프로그래밍 모델과 매우 비슷하다는 것을 알 수 있다. 이 모델은 WAP 응용 개발자들에게 몇 가지 유익을 제공한다. 예를 들면, 사용자에게 익숙한 프로그래밍 모델이라든지, 안정된 구조, 그리고 현재 사용되고 있는 도구인 웹 서버나 XML (eXtended Markup Language) 저작도구 등과의 호환능력 등을 들 수 있다. 그리고 이 모델은 무선 환경을 위해 최적화 되도록 설계된다. 또한 이 모델은 가능한 어떤 곳에서도 현재의 표준을 적용할 수 있으며, WAP를 이용한 모든 응용기술에 기반이 되는 모델이다.

WAP의 콘텐츠와 응용프로그램은 우리에게 익숙한 웹 콘텐츠 형식(WWW Contents Format)을 기반으로 해서 만들어진다. WAP 콘텐츠는 웹 통신

프로토콜을 기반으로 하여 만들어진 통신 프로토콜인 WTP(Wireless Transport Protocol)을 사용하여 전송된다. 무선 터미널에서는 사용자 인터페이스인 마이크로 브라우저를 사용하여 전체 응용프로그램을 제어하는데, 이것은 웹에서 이용되는 표준 웹 브라우저와 비슷한 기능을 가진다. WAP은 이동 터미널과 네트워크 서버 사이의 통신을 가능하게 하는 표준 컴포넌트의 집합으로 정의할 수 있으며, 다음과 같은 특징을 가지고 있다.

- 표준 네이밍 모델(standard naming model) : 웹 표준 URL은 웹 서버 상에 있는 WAP 콘텐츠를 인식하는데 사용되고, 또한 이것은 장치 안에 있는 지역 자원을 인식하는데 사용된다.
- 콘텐츠 타이핑(contents typing) : 모든 WAP 내용은 웹 타입과 호환되는 특별한 타입을 가지며, 따라서 WAP 클라이언트는 웹 타입을 기초로 하여 작성된 모든 내용을 정확하게 처리할 수 있다.
- 표준 콘텐츠 형식 : WAP 콘텐츠 형식은 웹 기술에 기초를 두고 있으며, 화면 표시용 마크업(Markup), 달력 정보, 전자 비즈니스 카드, 이미지 그리고 스크립트 언어 등을 포

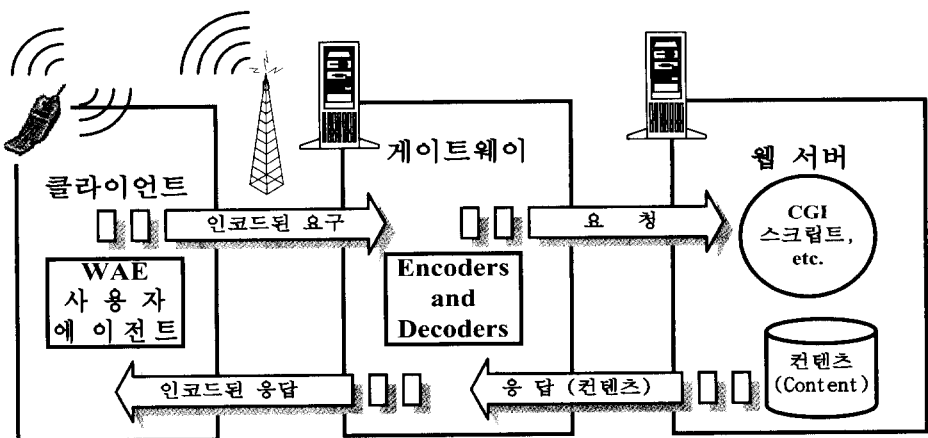


그림 2. WAP 프로그래밍 모델

함한다.

- 표준 통신 프로토콜 : WAP 통신 프로토콜은 휴대 무선장비로부터 네트워크에 연결된 웹 서버와의 통신을 가능하게 한다.

WAP 콘텐츠 유형(contents type)과 프로토콜은 통신시장과 모든 휴대형 무선 장비를 위해 최적화 되었으며, WAP은 무선 도메인과 웹 서버사이의 원활한 연결을 위해 프록시(proxy) 기술을 사용한다. WAP 프록시는 전형적으로 다음의 기능을 포함하고 있다.

- 프로토콜 게이트웨이(gateway) : 이것은 WAP 프로토콜 스택(stack)인 WSP(Wireless Session Protocol)와 WTP(Wireless Transaction Protocol) 및 WTLS(Wireless Transport Layer Security), 그리고 WDP(Wireless Datagram Protocol)를 웹 프로토콜 스택, 즉 HTTP와 TCP/IP로 전환하는 기능을 갖는다.
 - 콘텐츠 부호화기와 복호화기 : 콘텐츠 부호화기는 네트워크 상에서의 데이터의 크기를 줄이기 위해 WAP 내용을 간결하게 부호화된 포맷으로 해석한다.
- 이 구조는 휴대용 무선장비의 사용자가 다양한

WAP 콘텐츠와 응용프로그램을 폭넓게 이용 할 수 있도록 하며, 응용 시스템 제작자가 다수의 이동 터미널 기초 위에서 실행되는 콘텐츠 서비스와 기타 시스템 등을 만들 수 있게 한다. WAP 프록시는 표준 웹 서버 상에서의 콘텐츠와 응용프로그램이 추가 되도록 즉, 우선권을 가지도록 하며, CGI (Common Gateway Interface) 스크립팅과 같은 안정된 웹 기술의 사용을 가능하도록 한다. WAP 표준에서는 하나의 웹 서버와 WAP 프록시 그리고 WAP 클라이언트를 정의하고 있으므로 WAP 구조는 다른 구성 요소들을 더 쉽게 지원 할 수 있으며, WAP 프록시 기능을 포함한 웹 서버의 구현을 가능하게 해준다.

2.1 WAP 네트워크

일반적인 WAP 네트워크의 동작은 그림 3과 같다. WAP 클라이언트는 무선 네트워크 망에서 2개의 서버와 통신을 한다. WAP 프록시는 WAP 요청을 웹 요구로 바꾸고, 그것에 의하여 WAP 클라이언트가 웹 서버에 자신의 요청을 보낼 수 있도록 해준다. 또한 WAP 프록시는 클라이언트가 이해 할 수 있는 축소된 바이너리 포맷으로 웹서버로부터 응답을 코드화 한다. 만일 웹 브라우저가 WML로 만

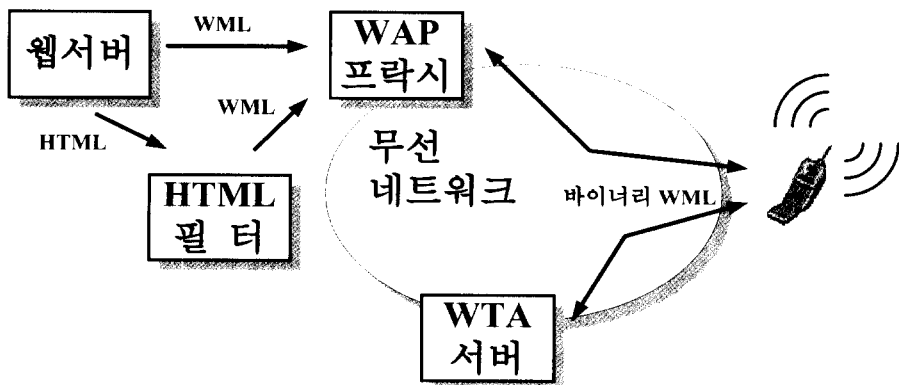


그림 3. WAP 네트워크의 예

들어진 WAP 콘텐츠를 제공한다면, WAP 프록시는 그것을 웹 브라우저로부터 직접 가져오지만 만일 웹 브라우저가 HTML로 만들어진 웹 콘텐츠를 제공한다면, 특정 필터가 웹 내용을 WAP 내용으로 바꾸기 위해 사용된다. HTML 필터를 이용하면 HTML을 WML로 바꿀 수 있다.

2.2 보안 모델

WAP은 WAP 클라이언트와 서버사이의 연결상의 보안기능을 제공하는데 초점을 맞추어, 유연성 있으며 신뢰할 수 있는 보안구조를 제공한다. WAP은 WAP 프로토콜을 이용하는 두 개의 종점(end-to-end)간 보안기능을 제공한다. 만일 브라우저와 웹 서버가 종점간의 보안을 요구한다면 그들은 WAP 프로토콜을 사용하면서 직접 연결해야 한다. 또한 종점간 보안은 WAP 프록시가 웹 서버와 물리적으로 같은 위치에 있다면 이루어 질 수 있다.

2.3 WAP 구조

WAP 구조는 이동 통신 장치를 위한 응용프로그램

개발을 위해 그림 4에서 보는 바와 같이 확장 가능한 프로토콜 환경과 전체 네트워크의 계층화된 구조를 제공한다. WAP 구조에서 각각의 계층은 상위 계층에 의해서 접근 될 수 있으며 다른 서비스나 응용프로그램에 의해서도 이용 될 수 있다. 외부 응용 프로그램은 세션, 트랜잭션, 보안, 그리고 트랜스포트 계층에 직접 접근 할 수 있다.

1) 무선 응용 환경 (WAE : Wireless Application Environment)

WAE는 웹과 이동 통신 기술의 통합을 기초로 한 일반 목적의 응용 환경이다(4). WAE의 주요 목적은 서비스 제공자가 효과적이고 유용한 방법으로 방대한 다른 무선 플랫폼(platform)에 접근 할 수 있는 응용과 서비스를 만들 수 있는 상호작용 환경을 구축하는 것이다. WAE는 다음의 기능을 포함하는 마이크로브라우저(microbrowser) 환경을 포함한다.

- WML(Wireless Markup Language): HTML과 비슷하지만 휴대형 무선단말기를 사용하는데 있어 최적화 된 경량(lightweight)의 마크업 언어이다.
- WMLScript : JavaScript와 유사한 경량의 스크립트 언어이다.

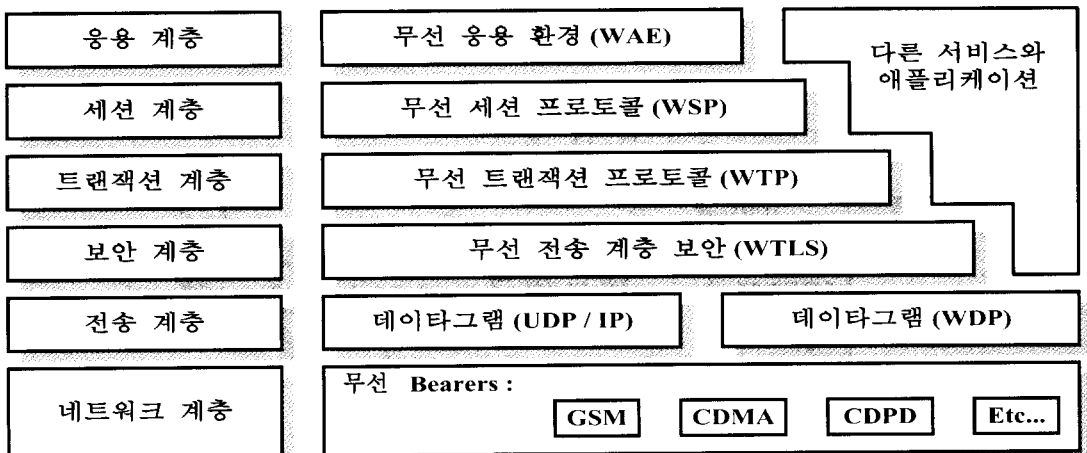


그림 4. WAP 스택(Stack)

- WTA(Wireless Telephony Application), WTAI(Wireless Telephony Application Interface)[5,6]: 전화 서비스와 이를 위한 프로그래밍 인터페이스이다.
- 컨텐츠 형식: 이미지, 전화번호 및 달력 정보를 포함한 데이터 형식을 말한다.

2) 무선 세션 프로토콜(WSP : Wireless Session Protocol)

WSP는 두 개의 세션 서비스를 위한 일관된 인터페이스를 가진 WAP의 응용 계층 프로토콜을 제공한다[7]. 첫째는 트랜잭션 계층 프로토콜인 WTP(Wireless Transaction Protocol) 위에서 동작하는 연결 지향형(connection-oriented) 서비스이고, 둘째는 보안 또는 비보안 데이터그램 서비스인 WDP(Wireless Datagram Protocol) 위에서 동작하는 비연결형(connectionless) 서비스이다. WSP는 브라우징 응용인 WSP/B(Wireless Session Protocol-Browsing)의 적절한 서비스들로 구성된다. WSP 군에 있는 프로토콜은 상대적으로 긴 지연(latency)을 가지고 있으며, 낮은 대역폭을 가진 전송 네트워크(bearer networks)를 위해 최적화 되어있다. WSP/B는 WAP 프로토콜이 WSP/B 클라이언트로서 표준 HTTP 서버에 연결 할 수 있도록 고안되어 있다. WSP는 현재 브라우징 응용에 적절한 서비스들로 구성되어 있으며, WSP/B는 다음과 같은 기능들을 제공한다.

- 축소된 인코딩 속에서의 HTTP/1.1 기능
- 영구적인 세션 상태 정보
- 세션 이동과 함께 세션 보류와 재시작
- 데이터 푸시(push)를 위한 설정 정보
- 프로토콜 특징의 상호 교환

3) 무선 트랜잭션 프로토콜(WTP : Wireless Transaction Protocol)

WTP는 데이터그램 서비스의 상위에서 실행되고 인터넷 접속 전용 컴퓨터에서 실행하기에 적당한 간단한 트랜잭션 지향형 프로토콜이다[8]. 이 WTP는 보안 또는 비보안 무선 데이터그램 네트워크 상에서 모두 효과적으로 동작한다. 즉, 트랜잭션 서비스에 있는 3개의 클래스인

- 신뢰할 수 없는 단방향 요청
- 신뢰할 수 있는 단방향 요청
- 신뢰할 수 있는 양방향 요청-응답 트랜잭션

에 대해 WTP 사용자는 자신이 받은 메시지의 확인을 위해 사용자간의 데이터를 교환함으로써 신뢰성을 유지한다. 또한 대역폭에서 벗어난 데이터가 들어올 때, 선택적으로 승인(Acknowledge)하는 기능, 전송한 메시지의 수를 감소시키기 위한 PDU(Protocol Data Unit)로의 연결과 지연의 통지 및 비동기 트랜잭션을 수행한다.

4) 무선 전송 계층 보안(WTLS : Wireless Transport Layer Security)

WTLS는 산업표준 트랜스포트 계층 보안(TLS: Transport Layer Security) 프로토콜에 기반을 둔 보안 프로토콜로서 공식적으로는 보안 소켓 계층(SSL: Secured Socket Layer)에 기반을 둔 프로토콜이다[9]. WTLS는 WAP 트랜스포트 프로토콜을 사용하며, 좁은 대역폭을 가진 통신 채널을 사용하도록 최적화 되어있다. WTLS는 터미널사이의 안전한 통신을 위해 사용된다. 그 예로는 전자 비즈니스카드의 인증을 들 수 있다. 응용 프로그램은 그들의 보안 요구와 기초를 이루는 네트워크의 특징에 의존하는 WTLS의 기능을 선택적으로 동작할 수 있게 하거나 억제 할 수 있다.

5) 무선 데이터그램 프로토콜 (WDP: Wireless Datagram Protocol)

WAP구조에서의 트랜스포트 계층 프로토콜은 WDP[10]를 사용하는데, 이 WDP 계층은 다양한

네트워크 형태에 의해 지원되는 데이터를 운반하는 능력을 가진 전송 서비스 위에서 동작한다. 일반적인 전송 서비스로서, WDP는 WAP 상위계층 프로토콜에 일관된 서비스를 제공하며, 이용 가능한 전송 서비스중의 하나를 가지고 통신한다. WDP 프로토콜이 상위계층 프로토콜의 일반적인 인터페이스를 제공하기 때문에 보안, 세션 그리고 응용 계층의 기초를 이루는 무선 네트워크 기능을 독립적으로 수행할 수 있다.

6) 데이터 전송자(Bearers)

WAP 프로토콜은 짧은 메시지, 회선제어 데이터 그리고 패킷 데이터를 포함한 다양한 전송 서비스 상에서 동작할 수 있도록 고안되어졌다. 각각의 데이터 전송자는 처리량, 에러율 그리고 지연에 대해서 다른 수준의 서비스를 제공한다. WAP 프로토콜은 이러한 다양한 수준의 서비스를 모두 수용할 수 있도록 고안되어졌으며 지원되는 데이터 전송자의 리스트도 무선통신 시장의 발전에 따라 새로운 전송

자가 지속적으로 추가 될 것이고, 이에 따른 서비스의 수준들도 다양하게 변하게 될 것으로 보인다.

7) 다른 서비스와 응용들

WAP의 계층화된 구조는 다른 서비스와 응용프로그램으로 하여금 이미 정의된 인터페이스들의 집합을 통해서 WAP 스택의 특징을 사용하도록 해준다. 외부 응용프로그램은 세션, 트랜잭션, 보안 그리고 트랜스포트 계층을 직접 접근 할 수 있다. 이것은 WAP 스택이 특별한 WAP에 의해서 뿐만 아니라, 무선 시장에서 필요로 하는 응용프로그램과 서비스들을 위해 사용할 수 있도록 허용한다. 예를 들면, 전자메일, 달력, 전화번호부, 노트패드 그리고 전자상거래 서비스와 같은 응용프로그램은 WAP 프로토콜을 사용함으로써 발전 될 수 있다.

8) WAP 기술의 구성 예제

WAP 기술은 WAP 포럼에 의해 제시된 것 이상

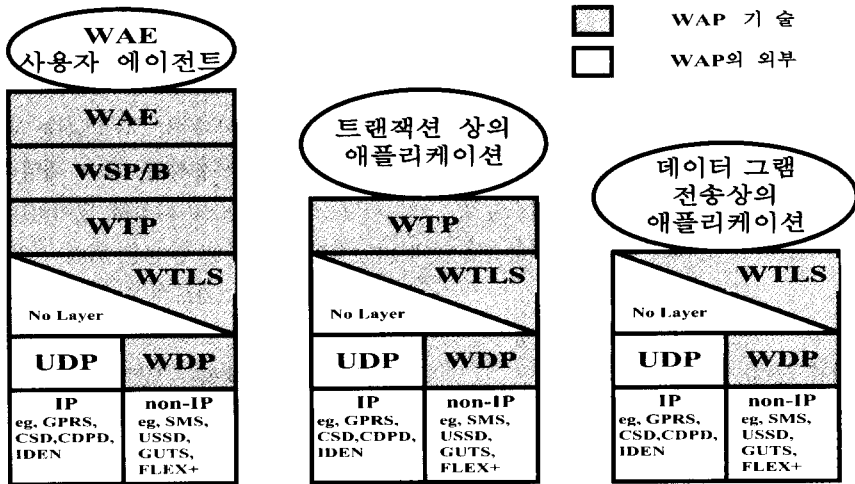


그림 5. WAP 스택의 예

*약어표

- WAE : Wireless Application Environment
- WSP/B : Wireless Session Protocol/Browsing
- WTP : Wireless Transaction Protocol
- WTLS : Wireless Transport Layer Security
- WDP : Wireless Datagram Protocol
- UDP : User Datagram Protocol

으로 응용프로그램과 서비스들을 위해서 유용하게 사용될 수 있다. 그림 5는 WAP기술을 사용함으로써 나타낼 수 있는 가능한 서로 다른 구조의 프로토콜 스택이다.

가장 왼쪽 스택은 WAP 응용의 전형적인 예를 나타내는데, WAP 기술의 완전한 스택 상에서 실행되는 WAE 사용자 에이전트를 나타낸다. 가운데 스택은 보안이 있거나 또는 없는, 트랜잭션 보안을 요구하는 응용프로그램과 서비스에 의해 사용되는 경우이다. 가장 오른쪽 스택은 보안이 있거나 또는 없는 데이터그램 전송을 단지 요청만 하는 응용프로그램과 서비스에 의해 사용되는 예제이다.

2.4 WML(Wireless Markup Language)과 WMLScript

1) WML

WML은 태그에 기초한 문서언어이며, XML에 기반을 둔 일반적인 마크업 언어이다[2]. WML은 HTML과 HDML이 가진 능력을 공유한다. WML은 XML 문서 형식으로서 나타내어지며, 전화나 다른 무선이동터미널과 같은 제한된 능력을 가진 장치 상에서 프리젠테이션과 사용자의 상호작용을 나타내는데 최적화 되어 있다. WML과 이것을 지원하는 환경은 작은 디스플레이를 포함하는 작은 휴대용 단말기, 제한된 사용자 입력 설비, 좁은 대역폭을 가진 네트워크의 연결, 제한된 메모리 자원과 제한된 CPU 자원으로 특징지어 진다.

WML은 HDML 2.0의 일부분에 기초를 두고 있다. WML은 HDML로부터 적용된 몇 가지 요소를 바꾸고 HTML과 비슷한 요소를 가진 새로운 일부 요소를 추가하였다. WML에서도 카드(card)와 데크(deck)라는 개념을 사용한다. 이것은 응용프로그램이 다중 카드로 만들어진 문서(일반적으로 데크라고 말한다)를 나타낼 수 있게 설계되어 있다. 논리

적으로 사용자는 WML 카드를 통해서 웹을 탐색할 수 있다. 카드의 내용을 다시 봄으로써, 요구된 정보로 들어갈 수도 있고, 선택을 만들 수도 있으며 다른 카드로 움직일 수도 있다. 카드 안에 있는 명령어는 특별한 상호작용에 의해 필요에 따라 웹 서버 상에서 서비스를 실행할 수 있다. 데크는 필요에 따라 웹 서버로 패치 되어진다. WML 데크는 웹 서버 상에서 정적 파일로 저장될 수 있으며, 반면에 그것들은 동적으로 웹 서버 상에서 실행되는 내용 생성기에 의해 생성될 수도 있다. 데크 안에 있는 각각의 카드는 특별한 사용자 상호 작용을 위한 기능을 가지고 있다.

WML은 넓고 다양한 장치 상에서 데이터를 표현할 수 있도록 설계되어 졌으나, 사용자로부터 들어오는 요청을 어떻게 실행하는지에 대해서는 기술하지 않는다. 예를 들면, 실행프로그램은 많은 웹 사용자 에이전트와 같이 가상적으로 사용자 입력을 청구하도록 선택하거나, 아니면 목소리에 기초한 인터페이스를 사용하도록 선택할 수 있다. 사용자 에이전트는 장치에 의존하는 카드 안에 모든 요소를 넣을 수 있는 최상의 방법을 결정해야 한다. 예를 들면, 상당히 커다란 디스플레이를 가진 장치 상에 있는 어떤 사용자 에이전트는 단일 카드 안에 모든 정보를 즉시 나타내도록 선택할 수 있다. 반면에 유사한 디스플레이를 가진 다른 사용자 에이전트는 이 커다란 디스플레이를 몇 개의 단위로 내용을 분리할 수도 있을 것이다.

WML은 다음과 같은 특징을 가진다.

- 텍스트와 이미지의 지원
- 사용자 입력을 위한 지원
- 탐색(navigation)과 히스토리(history) 스택 제공
- 국제언어의 지원
- 독립적인 MMI(Man Machine Interface)
- 좁은 대역에 최적화 된 대역 효율성
- 상태와 문맥관리 기능

2) WMLScript

WMLScript는 배우기 쉽게 고안된 스크립팅 언어이다[11]. 이것은 표준 브라우징과 행동 능력을 가진 WML의 표현 능력을 강화시킨 것이며, 진보된 사용자 인터페이스를 지원하며, 클라이언트에 기능을 추가하고, 어떠한 장치와 그것의 주변기에 접속하기에 편리한 기법 등을 제공한다. 또한 웹서버로의 왕복 여정에 따른 시간지연을 감소시킨다. WMLScript는 WML 데크에 프로시저 구문을 추가하기 위한 표준 수단을 제공한다.

3. HDML의 개요

HDML(Handheld Device Markup Language)은 휴대용 장비로부터의 효과적인 웹의 접근을 위하여 미국의 Unwired Planet에 의하여 개발된 언어로서 1997년 5월 W3C(World Wide Web Consortium)에 무선장비의 웹 접근에 관한 표준안으로 제출되었다[1, 13]. HDML은 태그(tag)에 기반을 둔 언어이며, 기존의 HTML,

XML[12], SGML과는 달리 제한된 재원을 가진 시스템 상에서의 데이터의 표현과 사용자와의 상호작용을 나타낸다. HDML 플랫폼 상에서 휴대용 무선장비는 HDML을 지원하는데 HDML 플랫폼의 구성요소로는 HDML 전용 휴대폰, HDML 문서, 휴대형 장비와 웹을 연결해주는 게이트웨이(gateway) 및 HDML을 위한 웹서비스를 제공하는 웹서버를 들 수 있다. 각 구성요소들의 관계는 그림 6에 나타나 있다.

3.1 HDML 전용 휴대폰

HDML 전용 휴대폰(HDML phone)은 브라우저(browser)를 실행하는 데이터 교환이 가능한 휴대용 장비이다. 브라우저는 휴대용 무선장비가 HDML을 해석하거나 게이트웨이(gateway)와 통신할 수 있도록 해주는 소프트웨어 모듈이다. 브라우저는 웹 브라우저(web browser)와 매우 비슷한 구조를 가진다. 사용자는 그림 7에서 보는 바와 같이 HDML 전용 휴대폰의 키패드를 이용하여 데이

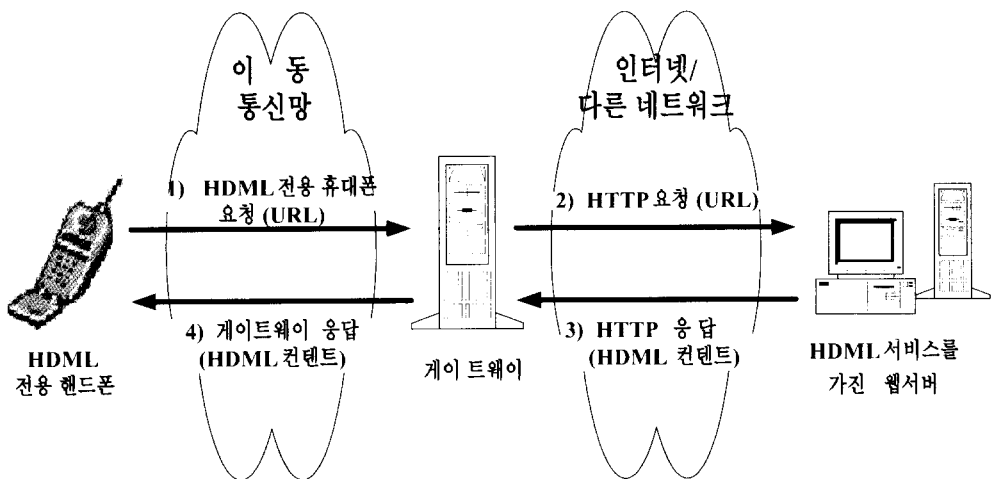


그림 6. HDML 플랫폼상의 구성요소간의 관계

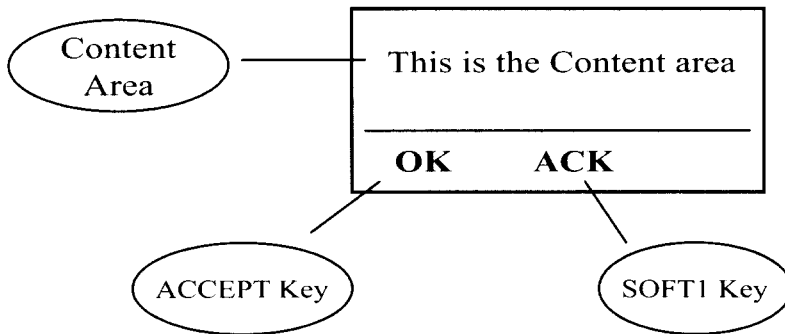


그림 7. HDML 전용 브라우저의 구조

터를 교환할 수 있으며 사용자가 다른 위치로 옮길 때, HDML 전용 휴대폰은 다른 카드를 적재하며 디스플레이와 입력 사양(input option)을 바꾸게 된다.

3.2 HDML 문서

HDML은 태그에 기반을 두며 제한된 제원을 가진 장치 상에서의 표현과 사용자 상호작용을 보이는데 최적화 되어 있는 언어이다. HDML 문장은 전체에 정보를 나타내거나, 사용자를 위한 입력 옵션을 제공하고, 사용자가 키를 누를 때, 전화가 어떻게 응답하는지를 나타낸다. HDML은 카드(card)와 데크(deck)라는 개념을 사용한다. 이것은 응용프로그램이 다중 카드로 만들어진 문서(일반적으로 이를 데크라고 하며 이는 HTML의 페이지에 해당됨)를 나타내도록 하는 설계를 포함한다. 논리적으로 사용자는 HDML 카드의 집합을 통해서 웹을 탐색하게 된다. 따라서 HDML의 기본단위는 카드이며, 이것은 전화와 사용자 사이의 일관된 상호 행동을 나타낸다.

HDML이 제공하는 카드의 종류로는 디스플레이 카드(display card), 입력 카드(entry card), 선택 카드(choice card) 그리고 노-디스플레이 카드(no-display card)가 있는데 이 네 가지 종류의 카드를 이용하여 여러 개의 카드를 하나의 데크

안에 구성함으로써 사용자가 원하는 형태의 서비스 구조를 만들 수 있다. 각각의 카드들은 다음과 같은 기능을 한다. 먼저 디스플레이 카드는 정보를 단순히 사용자에게 보여주며, 입력 카드는 사용자에게 메시지를 보여주고 사용자가 데이터를 입력할 수 있도록 하기 위해서 사용된다. 선택 카드는 메뉴 리스트를 보여주어 사용자가 원하는 메뉴를 선택할 수 있도록 지원하며, 노-디스플레이 카드는 실제로는 디스플레이카드와 똑같이 실행은 되지만 사용자에게 보여지지는 않는다. 노-디스플레이 카드는 변수를 초기화한다든지, 또는 사용자 인터페이스를 가지지 않는 하부 활동(activity)을 실행하는 것과 같은 특별한 기능을 수행하는데 사용된다.

웹서버가 HDML 전용 휴대폰에 전달 할 수 있는 HDML의 가장 작은 단위는 데크이다. 데크는 하나 이상의 카드를 포함하여 HDML 전용 휴대폰이 데크를 받아들일 때, 데크에 있는 첫 번째 카드를 실행한다. HDML 전용 휴대폰은 카드가 제공하는 정보를 보여주며 사용자가 카드에 응답하도록 해준다. 카드의 종류에 의존하기 때문에 사용자는 텍스트를 넣거나 옵션을 선택하거나 기능 키를 누름으로써 응답할 수 있다.

HDML 프로그램의 예를 그림 8에서 살펴보자. 이 예제에서는 1개의 데크 안에 3개의 카드를 사용하여 날씨 정보를 나타내고 있다. 휴대폰이 데크를

```

<HDML VERSION=2.0>
  <DISPLAY>
    <ACTION TYPE=ACCEPT LABEL=Tues TASK=GO DEST=#tues>
    Current temps
    <BR>Hi: 60
    <BR>Lo: 28
  </DISPLAY>
  <DISPLAY NAME=tues>
    <ACTION TYPE=ACCEPT LABEL=Wed TASK=GO DEST=#wed>
    Tuesday temps
    <BR>Hi: 78
    <BR>Lo: 36
  </DISPLAY>
  <DISPLAY NAME=wed>
    Wednesday temps
    <BR>Hi: 80
    <BR>Lo: 40
  </DISPLAY>
</HDML>

```

그림 8. HDML 프로그램의 예

맨 처음 실행할 때, 휴대폰은 데크 안에 있는 처음의 카드인 현재의 기온이 담긴 디스플레이 카드를 사용자에게 보여준다. 휴대폰이 계속해서 종속되는 하부 카드를 사용자에게 보여주는 순서는 사용자의 요구 사항이 표현된 데크 속의 HDML 문법에 의해 나타나는 링크의 진행과정에 의존한다. 여기서 정의된 데크는 그림에서 보이는 대로 화살표의 지시에 따른 링크의 진행과정을 보여준다. 사용자가 첫 번째 카드에 있는 ACCEPT 키를 누르면, 휴대폰은 두 번째 카드를 사용자에게 보여준다. 이러한 행동은 첫 번째 카드에 있는 다음과 같은 <ACTION> 문장에 의존하여 행동하게 된다. 이 <ACTION> 문장은 HDML 전용 휴대폰에 장착된 브라우저의 행동을 규정한다. 다음은 액션(ACTION) 문장의 예이다.

```

<ACTION TYPE=ACCEPT LABEL=Tues
TASK=GO DEST=#tues>

```

이 문장은 휴대폰에 있는 ACCEPT 키의 라벨을

“OK”에서 “Tues”로 바꾸도록 지시하며, 사용자가 이 키를 눌렀을 때, 데크 안에 있는 두 번째 카드의 이름인 “Tues”로 이동하게 된다. 마찬가지로 두 번째 카드에 있는 <ACTION> 문장은 휴대폰에 있는 ACCEPT 키의 라벨을 “OK”에서 “Wed”로 바꾸도록 지시하며, 사용자가 ACCEPT 키를 눌렀을 때, 카드의 이름이 “Wed”인 카드로 이동하게 된다. ACCEPT 키의 초기 행동은 이전 카드로 되돌아가는 것이다. 예를 들어, 위의 예제에서 데크 안에 있는 세 번째 카드는 ACCEPT 키가 어떻게 행동할지를 결정하는 <ACTION> 문장을 가지고 있지 않기 때문에, 이 세 번째 카드에서 사용자가 ACCEPT 키를 누른다면, 휴대폰은 이전의 카드로 되돌아가게 된다.

3.3 게이트웨이(Gateway)

게이트웨이는 HDML 플랫폼의 핵심부이며, HDML 전용 휴대폰 사용자들의 데이터베이스를

관리한다. 게이트웨이는 HDML 전용 휴대폰의 요청을 웹 서버에 전달해주며 웹 서버로부터의 응답을 HDML 전용 휴대폰에 전달해 준다. HDML 휴대폰이 게이트웨이와 함께 동작할 수 있도록 하기 위해서는 게이트웨이의 데이터에 등록해야만 한다. 사용자가 HDML 전용 휴대폰의 키(key)에 전달된 URL(Uniform Resource Locator) 요청을 눌렀을 때, HDML 휴대폰은 HDTP(Handheld Device Transport Protocol)를 사용하여 게이트웨이에 그 요청을 보내게 된다. HDTP는 휴대형 장비를 이용한 무선망에서의 통신을 위해 최적화된 개방형 프로토콜로서, 클라이언트와 서버 사이의 원격작동을 위한 세션 계층의 프로토콜이다. 그림 9는 게이트웨이를 이용한 데이터의 전달방법을 보여준다.

3.4 웹서버

웹서버는 HDML과 HTTP(HyperText Transfer Protocol) 요청에 대한 응답 내용을 제

공하는 웹서버 응용 프로그램을 말한다. 웹서버의 예로서는 다음과 같은 것들이 있다.

- 보안 추적 시스템(Securities tracking systems)
- 주문 추적 시스템(Order tracking systems)
- 기상 및 도로정보 보고 (Weather and road condition reports)

웹서버는 CGI(Common Gateway Interface)가 HTML을 생성하는 것과 같은 방법으로 유동적으로 HDML을 생성한다. 예를 들면, 데이터 베이스나 다른 외부 데이터 제공자에게 질문함으로써 웹 상에 원래 존재하는 HTML 내용으로부터 HDML을 생성하는 웹서버를 쉽게 개발할 수 있다. 이 단계에서 다음과 같은 풀(PULL) 거래(transaction)와 푸시(PUSH) 거래를 원하는 웹서버에서 실행할 수 있다.

- 풀(PULL) 거래 : HDML 전용 휴대폰의 요청에 응답서비스를 제공
- 푸시(PUSH) 거래 : HDML 전용 휴대폰에

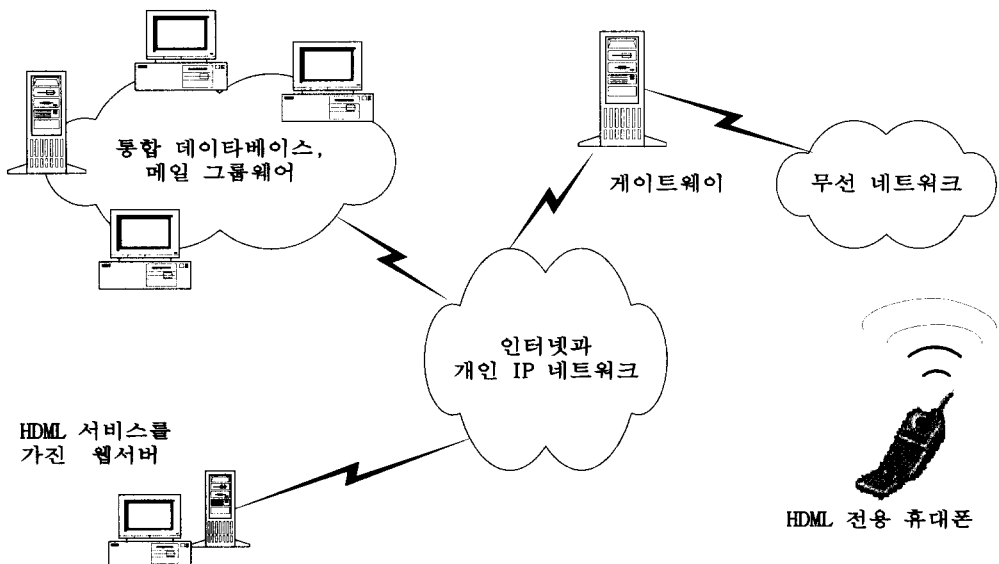


그림 9. 게이트웨이를 이용한 데이터 전달방법

비동기적으로 데이터를 보내는 서비스 제공

3.5 HDML 플랫폼의 작동

HDML 플랫폼(platform)은 핸드폰, PDA, 양방향 호출기 등과 같은 데이터 교환이 가능한 휴대용 장비를 이용하여 제공되는 서비스에 무선으로 접근할 수 있도록 하는 장치이며, 이 서비스는 인터넷 상에서 제공되는 거의 모든 서비스를 이용할 수 있다. 게이트웨이는 인터넷을 통하여 HTTP 요청을 보내거나 또는 응용프로그램 개발자에 의해 관리되는 웹서버에 직접 연결할 수도 있다. 게이트웨이의 요청을 제어하는 웹서버 위에 있는 응용프로그램이 HDML 웹서비스이다. 이것은 HDML을 사용하여 요청에 대해 응답한다. 게이트웨이가 웹서버로부터 응답을 받을 때, 게이트웨이는 요청이 시작된 HDML 전용 휴대폰으로 응답의 내용을 돌려보내는 중요한 역할을 하게되며, HDML 전용 휴대폰은 게이트웨이로부터 받은 데이터를 사용자에게 보여준다. 웹서버는 또한 사용자와의 통신을 처리 할 수 있는데 이것은 푸시(Push)기술을 이용하여 게이트웨이에 담겨진 경고와 정보를 사용자의 HDML 전용 휴대폰으로 푸시하게 된다. 이해를 돕기 위하여 웹의 동작 모습과 HDML 플랫폼의 동작 모습을 비교하여 설명하겠다.

1) 전형적인 웹에서의 동작

- ① 사용자는 웹브라우저에서 URL(Uniform Resource Locator)을 선택한다.
- ② 웹브라우저는 URL을 추출하여 웹서버에 HTTP 요청을 보낸다.
- ③ 웹서버는 어떤 파일을 가져올 것인지를 결정하기 위해 HTTP 요청을 추출한다. 만일 URL이 정적 파일을 나타낸다면, 웹서버는 파일을 가져오고 그것에 HTTP 헤더를 붙인다. 그러나 만일 URL이 CGI(Common Gateway

Interface) 프로그램을 나타낸다면, 웹서버는 프로그램을 내보낸다.

- ④ 웹서버는 HTTP 응답을 생성한다. 만일 요청된 URL이 정적 파일을 나타낸다면, 응답은 HTTP 헤더가 붙은 요청된 파일로 구성된다. 그러나 만일 요청된 URL이 CGI 프로그램을 나타낸다면, 응답은 CGI 프로그램의 출력을 나타낸다.
- ⑤ 웹브라우저는 HTTP 응답을 추출하고 사용자에게 내용을 보여준다.

2) HDML 플랫폼 상에서의 작동원리

- ① 사용자는 URL이 할당된 전화 키(key)를 누른다.
- ② 전화는 URL 요구와 HDTP 요구 속에 있는 다른 정보, 예를 들면 휴대폰 식별번호(Phone ID NUMBER)를 포함해서 게이트웨이에 URL 요청을 보낸다.
- ③ HDTP 요청은 게이트웨이를 포함하지 않고 특별한 URL을 위한 HTTP/HTTPS 요청을 생성한다. 이 HTTP/HTTPS는 이 요청을 웹서버에 보낸다.
- ④ 웹서버는 어떤 데이터를 가져올 것인지를 결정하기 위해 HTTP/HTTPS 요청을 추출한다. 만일 URL이 정적 파일을 나타낸다면, 웹서버는 파일을 가져온 후, 그것에 HTTP 헤더를 붙인다. 그러나 만일 URL이 CGI 프로그램(즉, 웹서버의 HDML 서비스)를 나타낸다면, 웹서버는 프로그램을 내보낸다.
- ⑤ 웹서버는 HTTP 헤더가 붙은 HDML 파일이 나오므로 CGI 프로그램으로부터 출력이 들어 있는 HDML을 반환(return)한다.
- ⑥ 게이트웨이는 HTTP 응답으로부터 HTTP 헤더를 제거하고 HDML을 포함하는 HDTP[14] 응답을 첨가하고, 첨가된 데이터

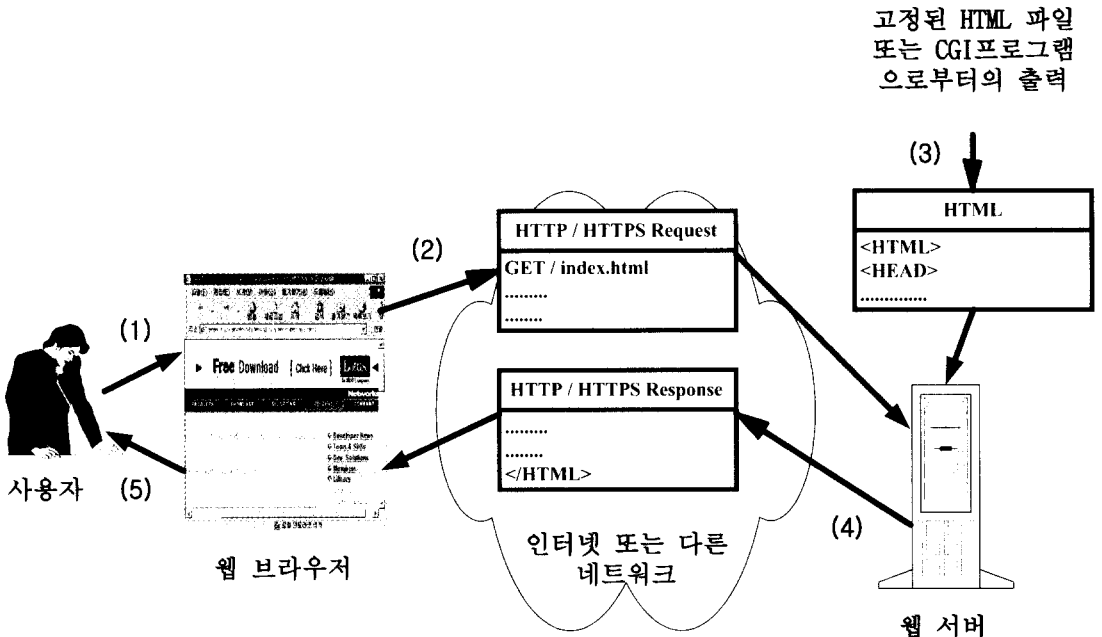


그림 10. 전형적인 웹 트랜잭션(transaction)

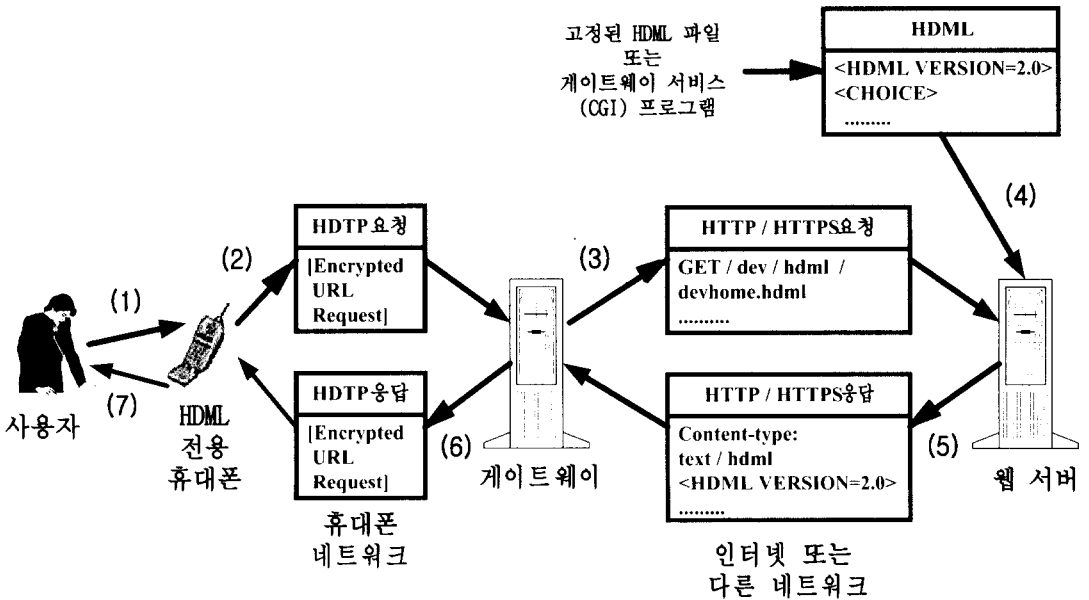


그림 11. 전형적인 HDML 플랫폼에서의 트랜잭션

를 HDML 전용 휴대폰에 보낸다.

- ⑦ HDML 전용 휴대폰은 HDTP 응답을 받고 응답으로부터 HDML을 추출하며 그리고 HDML만의 첫 번째 카드를 사용자에게 보여준다.

4. WML/HDML 저작도구 구현기술

4.1 WML/HDML 전용 저작도구의 필요성

WML/HDML과 이것을 지원하는 환경은 작은 디스플레이를 포함하여 제한된 사용자 입력설비, 작은 네트워크 용량, 제한된 메모리 자원과 제한된 처리 자원을 위하여 설계되었으므로, WML/HDML 응용업무의 제작에는 휴대용 장비의 기본적인 제약으로 인해서 다음과 같은 기술적인 난제들이 있다.

- WML/HDML 응용업무는 기본적으로 3x12 텍스트 스크린을 기반으로 제작되므로, 작은 화면에 다양한 정보의 신속한 표현이 요구된다.
- 휴대형 장비의 특성상 사용자의 다양한 설정 정보(customization information)을 비롯한 거의 모든 정보가 서버 쪽에 저장되므로 이의 원활한 이용이 요구된다.
- 모든 WML/HDML 응용프로그램의 제작 시에 일관성 있는 사용자 인터페이스의 유지와 이용의 용이성을 위하여 사용자 인터페이스 가이드 라인(User Interface Guideline)을 따를 것이 강력히 권고되고 있다.
- WML/HDML 응용업무는 데크/카드의 작동으로 구성되는데, 여러 개의 데크와 데크 내부에 4가지 종류로 구분되는 수십 혹은 수백 개의 카드가 이용된다. 데크와 데크, 카드와 카드, 데크와 카드사이에 하이퍼링크(HyperLink)가 필요한 데, 이러한 모든 링크들을 단순히 수작업으로 관리하는 데는 많은 어려움이 따른다.

- WML/HDML을 이용한 푸시 응용업무의 제공 시에는 다양한 데크들이 동적으로 제작되어야 하므로 이를 위한 DBMS(DataBase Management System) 연동이 필수적이다. 위에서 언급된 다양한 기술적 측면들을 모두 만족시키는 WML/HDML 응용업무를 단순한 편집기와 WML/HDML SDK(Software Development Kit)만을 이용하여 수작업으로 제작하려면 기술과 시간 또 많은 비용의 투자가 필요하다. 또한 WML/HDML 응용업무는 그 제공정보의 특성상 빠른 시간 안에 개발과 공급이 이루어져야 한다. 따라서 위의 기술적 측면들을 만족시키는 WML/HDML 전용 저작 도구의 개발이 절실히 요구된다.

본 연구에서 개발중인 저작도구에 구현될 핵심 모듈들의 내용은 다음과 같다.

- WML/HDML 문서의 자동생성과 하이퍼링크 관리 모듈 : 이동통신망에서의 웹 정보 검색 플랫폼을 위한 응용업무 개발을 위하여 WML/HDML 문서를 자동으로 생성할 수 있는 기술과 저작도구로 제작된 모든 링크를 자동관리하는 기술을 개발하되, 이를 CGI/PERL 스크립트 생성 기술, 미들웨어 저작도구와 데이터베이스 기술과 통합하여 국제표준의 WML/HDML 전용 저작 도구 기술의 구현.
- 이동통신망에서의 웹 정보의 실시간 푸시 모듈 : 원하는 정보를 최소한의 키 입력으로, 여러 개의 메뉴를 거치지 않고 검색하는 기술을 개발하며 이를 근간으로 사용자의 자료 요구에 부응하는 융통성 있는 인터페이스를 제공하며 또한 WML/HDML 브라우저 탑재 단말기가 기업 내 데이터베이스를 신속하게 목적에 맞게 검색할 수 있는 기술을 개발하되, 사용자의 개인적인 자료 지식을 최단시간에 추론하여 사용자의 개인 설정/취향에 부합하는 정보를 원하는 시간에 제공하는 푸시 기능 기술의 구현.
- 이동통신망에서의 음성 메일, 팩스 및 전자 메

일 통합 지원 모듈의 구현.

4.2 WML/HDML 전용저작도구의 기능

WML/HDML 전용저작도구는 다음과 같은 주요 기능모듈들로 구성되어 있다.

- 위치웍(WYSIWYG) 편집 기능과 파싱 기능 : 이동통신망에서의 웹 정보 검색 플랫폼을 위한 응용프로그램 개발을 위하여 WML/HDML 문서를 자동으로 생성하는 기능
- 링크 관리 기능 : 저작도구에서 표현되는 전체 WML/HDML 응용프로그램에 포함되는 모든 WML/HDML 문서들의 연결구조를 효과적으로 저장하고 관리하는 기능
- 브라우징 기능 : WML/HDML로 작성된 문

서를 전용 휴대폰에 미리보기 형태로 보여주는 기능

- 데이터베이스와의 연동: 데이터베이스에 WML/HDML 문서 및 관련 자료를 저장하고 모든 자료를 입/출력할 수 있는 WML/HDML 문서의 작성을 용이하게 하는 기능과 PERL을 이용한 CGI 프로그래밍에 대한 사전지식이 없이도 CGI 스크립트를 작성 할 수 있는 CGI 연동 및 자동생성 도구

4.3 WML/HDML 전용저작도구의 사용자 인터페이스

그림 12는 세종대학교에서 윈도우 95/98 및 NT 환경에서 개발중인 WML/HDML 전용 저작도구의 사용자 인터페이스를 보여준다. 현재 JAVA로

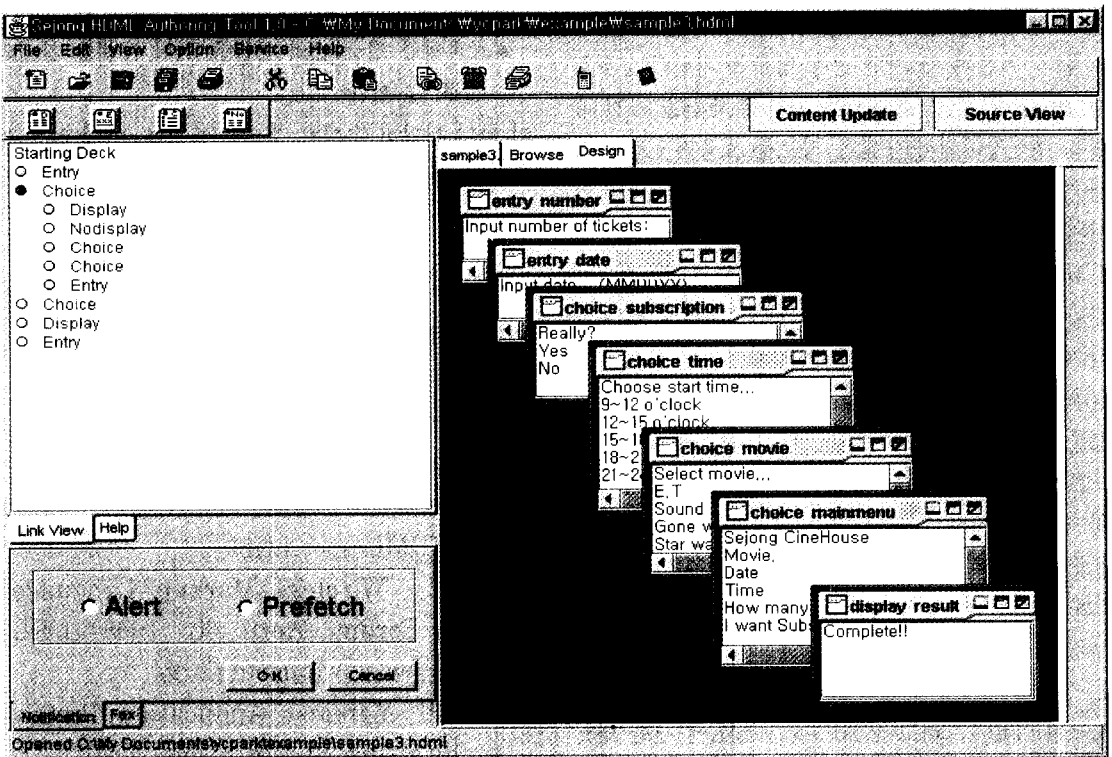


그림 12. WML/HDML 전용저작도구의 사용자 인터페이스

구현되어 프로토타입이 완성되었으며 현재 HDML 문서의 자동생성 기능, 링크관리기능, CGI Script 생성기능을 완성하였으나 앞에서 언급한 기능 모두를 구현할 계획이다.

5. 결 론

WWW와 이와 관련된 기술의 발달로 인해서 사용자는 언제 어디에서나 개인용 컴퓨터와 네트워크 연결이 있는 곳이면 자신이 원하는 정보를 손쉽게 얻을 수 있게 되었으나, 수많은 사용자를 확보하고 있는 이동통신망을 통한 웹 정보의 접근에 관한 연구 및 개발은 아직 미미한 형편이다. 또한 향후 IMT2000 휴대전화 서비스에서는 멀티미디어를 기반으로 하는 데이터 통신이 필수적이기 때문에, 무선 통신에서의 웹 정보의 접근에 대한 연구는 다가올 미래에 대한 신속한 대응책의 하나라 할 수 있다.

특히, 기존의 웹 기반 응용 업무를 지원하기 위한 언어인 HTML을 통해서는 제한된 제원을 가진 휴대용 무선장비에서 수많은 정보를 효율적으로 나타낼 수 없다. 이에 대한 대안으로 휴대용 무선장비에서의 효과적인 웹 접근을 위한 언어인 HDML이 개발되었고, 1997년 W3C에 무선장비의 웹 접근에 관한 표준안으로 제출되었다. 현재는 세계 60여개의 우수한 통신 및 컴퓨터 관련 회사들이 WAP 포럼을 형성하여 HDML을 기반으로 하는 WML의 표준제정을 추진 중에 있다. 현재 유일한 국제표준으로서 추진되고 있는 WML을 포함하는 WAP 전체 표준의 제정과 그에 따른 상용화에는 일반적인 표준화 과정의 통례로 볼 때 상당한 시일이 걸릴 것으로 예상된다. 또한 상용화를 위한 HDML 응용도구의 제작이 전 세계적으로 급속도로 이루어지고 있는 상황에서, HDML에 관련된 기술의 습득과 지원도구의 개발, 그리고 WAP 및 WML에 관한 연구는 미래의 정보통신 기술의 중심에 상당히 가깝게

위치해 있다고 하겠다.

본고에서는 휴대용 무선장비에서 웹 접근을 가능하게 하는 무선 응용 프로토콜인 WAP에 대한 개념과 구조에 대해 살펴보고, 무선 데이터 접근용 언어인 WML(Wireless Markup Language)과 HDML(Handheld Device Markup Language)에 대해 설명하였다. 특히 현재 사용되고 있는 HDML에 대해서는 문법 및 사용환경에 대해 좀더 자세하게 기술하였다. 마지막으로 소개된 것은 구현의 실례로 현재 세종대학교에서 개발 중에 있는 WML/HDML 전용 저작도구의 기능 및 사용자 인터페이스이다. 향후 이 분야에서는

- 이동통신망과 인터넷의 연동 시에 필수적인 WAP/HDML 게이트웨이에 관한 연구 및 개발
- WAP 표준화 제정 시에 WML 저작도구 및 관련 기술 개발
- 이동통신망을 위한 응용 프로토콜에 관한 연구
- 효과적인 웹 정보의 접근을 위한 휴대용 장비의 연구 및 개발

이 좀더 활발히 수행될 것으로 보이며, 이에 따른 필요 기술의 연구와 개발이 다양하게 이루어질 전망이다.

※ 참고문헌

1. [HDML] Handheld Device Markup Language Specification, <http://www.w3.org/TR/NOTE-Submission-HDML-spec.html>
2. [WML] "Wireless Markup Language", WAP Forum, April 30, 1998. URL : <http://www.wapforum.org/>
3. [WAP] "Wireless Application Protocol Architecture Specification", WAP

- Forum, April 30, 1998. URL :
<http://www.wapforum.org/>
4. [WAE] "Wireless Application Environment Specification", WAP Forum, April 30, 1998. URL : <http://www.wapforum.org/>
 5. [WTA] "Wireless Telephony Application Specification", WAP Forum, April 30, 1998. URL : <http://www.wapforum.org/>
 6. [WTAI] "Wireless Telephony Application Interface", WAP Forum, April 30, 1998. URL : <http://www.wapforum.org/>
 7. [WSP] "Wireless Session Protocol", WAP Forum, April 30, 1998. URL : <http://www.wapforum.org/>
 8. [WTP] "Wireless Transaction Protocol Specification", WAP Forum, April 30, 1998. URL : <http://www.wapforum.org/>
 9. [WTLS] "Wireless Transport Layer Security Protocol", WAP Forum, April 30, 1998. URL : <http://www.wapforum.org/>
 10. [WDP] "Wireless Datagram Protocol", WAP Forum, April 30, 1998. URL : <http://www.wapforum.org/>
 11. [WMLScript] "Wireless Markup Language Script", WAP Forum, April 30, 1998. URL : <http://www.wapforum.org/>
 12. Laurent, S., XML: A Primer, MIS Press, 1998.
 13. Getting Started with the UP.SDK, <http://www.uplanet.com/pub/index.html>
 14. HDTP Specification, <http://www.uplanet.com/pub/index.html>



신 동 일

1988년 연세대학교 전산학과(이학사)
 1993년 M.S. in Computer Science, Washington State University
 1997년 Ph.D in Computer Science, University of North Texas
 1997년 9월~1998년 2월 시스템공학연구소 선임연구원
 1998년 3월~현재 세종대학교 컴퓨터공학과 전임강사
 * 관심분야 : 이동통신, 멀티미디어 데이터베이스, CSCW, 지능형 에이전트



신 동 규

1986년 2월 서울대학교 계산통계학과 (이학사)
 1992년 8월 Illinois Institute of Technology 전산학과 (공학석사)
 1997년 8월 Texas A&M University 전산학과 (공학박사)
 1986년 2월~1991년 1월 한국국방연구원 연구원
 1997년 8월~1998년 2월 현대전자 멀티미디어연구소 책임연구원
 1998년 3월~현재 세종대학교 컴퓨터공학과 조교수
 * 관심분야 : 웹기반 멀티미디어, 멀티미디어 DB, 영상 압축, 웨이브릿 부호화



이 경 군

1981년 2월 서울대학교 전자공학과(공학사)
 1983년 2월 한국과학기술원 전기 및 전자공학과(공학석사)
 1992년 6월 Cornell University, School of Electrical Engineering (공학박사)
 1983.2~1986.7, 1992.7.-1998.2. 삼성전자 네트워크개발팀 수석연구원
 1998년 3월~현재 세종대학교 정보통신공학과 부교수
 * 관심분야 : 인터넷, 초고속통신, 이동통신