

WAP 환경을 위한 WML 변환 플랫폼과 사용자 에이전트 구현

오금용⁰ 고탉현 황인준
아주대학교 정보통신전문대학원 정보통신공학과
{pickboy, sanha, ehwang}@madang.ajou.ac.kr

Implementation of WML Conversion Platform and User Agent for WAP Environment

Keumyong Oh⁰ Takhun Ko Eenjun Hwang
Graduate School of Information and Communication, Ajou University

요 약

근래에 눈부신 통신 장비의 발달은 무선환경에서 인터넷 사용을 가능케 하였고, 또한 인터넷 사용자의 다양한 욕구 증가로 인해 무선 인터넷 서비스에 대한 기대도 급증하는 결과를 낳았다. 이를 위해 많은 통신사업자들이 연합하여 단체를 구성하고 무선 인터넷을 위한 프로토콜 표준을 제정하여 실용화하고 있다. 현재 WAP은 이들 프로토콜들을 대표하고 있고 결과로 많은 콘텐츠들이 WML로 제작되어 서비스되고 있다. 하지만 인터넷 상의 다수의 웹 페이지들은 이미 HTML을 사용하여 제작되었고 이는 WAP이 지원하지 않는 언어이므로 사용자를 위해 WAP 환경에 맞는 WML로 변환하는 과정이 필요하게 되었다. 이와 더불어 제공되는 서비스의 개발과 관리 목적으로 유선상에서도 이를 확인할 수 있는 사용자 에이전트가 필요하게 되었다. 본 논문에서는 HTML 문서를 WML 문서로 변환하는 변환 플랫폼과 이를 유선상에서 확인하고 관리할 수 있는 사용자 에이전트에 대해 제안한다.

1. 서론

디지털 혁명을 대표하는 정보통신 기술의 발전은 인류의 삶의 방식에 큰 변화를 초래하여 왔다. 개인용 컴퓨터와 인터넷의 폭발적 보급으로 인해 정보의 생성과 흐름은 시간과 공간의 제약에서 벗어나게 되었고 정보를 위한 유선 네트워크의 구축과 발전으로 거듭나게 되었다. 이러한 발전을 기반으로 정보처리의 영역을 무선으로 확대하고 유선상에서의 위치 고정성에서 벗어나 무선 상에서 누릴 수 있는 이동성과 휴대성에서 받을 수 있는 혜택을 갖게 되었다. 이러한 변화는 다양성을 선호하는 소비자의 욕구와 더불어 호출기로부터 휴대용 단말기를 거쳐 조만간 상용화 될 IMT2000 기반의 이동통신까지 최근 몇 년 사이에 이동 통신 환경의 발전에 기인하고 있다. 결과로 무선 단말기를 통해 가능한 부가 서비스에 대한 여러 가지 모색도 활발히 전개되고 있다. 하지만 유선 인터넷에서 이용되는 HTTP(Hypertext Transfer Protocol)등의 기술은 무선 단말기에 그대로 적용되기에 많은 문제가 있다. 무선 단말기는 일반 데스크탑과 달리 저용량의 프로세서와 메모리를 사용하고 작은 디스플레이와 배터리 용량의 한계라는 문제점을 가지고 있기 때문이다. 이에 따라 무선 통신 단말기에 맞는 프로토콜이나 기술들이 요구되었다.

WAP(Wireless Application Protocol)은 무선 단말기를 통한 인터넷 접근의 기술적 표준으로 탄생한 프로토콜이다. WAP은 WAP Forum에서 관련 명세서와 형식을 정의하고 있다. WAP Forum은 1997년 6월에 형성되었고, 이후 많은 기업에서 참여가 이루어지고 있다.

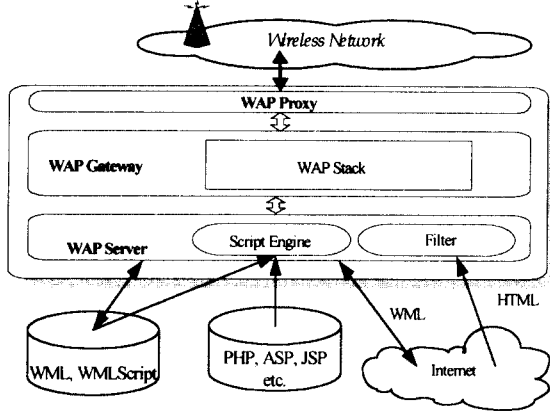
본 논문에서는 서버측에서 입력되는 HTML을 WML로 변환하는 WML 변환 플랫폼과 제공되는 무선 콘텐츠를 유선상에서 확인하고 관리할 수 있는 WAP 브라우저와 같은 사용자 에이전트 시스템에 대해서 제안한다.

2. WML 변환 플랫폼의 구현

WAP 프로토콜에 대한 지지가 늘어남에 따라, WAP 단말기상에서 서비스가 가능한 콘텐츠에 대한 제작이 늘어나고 있다. 이러한 콘텐츠들은 WML(Wireless Markup Language)을 사용하여 제작되어 있는데 WML은 기존에 많이 사용하던 HTML과는 다른 형태의 언어이다. 현재 많은 웹 브라우저들이 HTML은 지원하지 않지만 WML은 지원하지 않고 있고, 반대로 WAP 단말기에서는 WML만이 브라우징이 가능하다. 따라서 정보의 교환을 위해서는 변환하는 과정이 필요하며 이를 위해 첫째 기존에 제작되었던 HTML 문서를 WML 문서로 보여주기 위한 WML 변환 장치와 둘째 WML로 제작되어진 문서를 유선상에 있는 브라우저를 통해서 무선 WAP 단말기와 같이 볼 수 있는 WAP 브라우저가 필요하게 된다.

WML 변환 플랫폼은 HTML 문서를 WML 문서로 변환하는 장치이다. 이것을 이용하여 무선 단말기에서 기존의 HTML로 제작되어진 문서의 정보로 빠르게 접근할 수 있고 별도의 데이터베이스나 다른 부가적인 시스템의 사용 없이도 인터넷이나 인트라넷 상의 정보를 변환해 무선 단말기 상에 최적의 상태로 표시할 수 있다. 이를 위해서 HTML 문서에서 정보를 추출하고 불필요한 부분에 대한 삭제와 수정의 과정을 거쳐 이를 WML 형태로 바꾸는 일련의 과정이 필요하다. 이때 필요에 의해 재구성의 과정이 존재 할 수도 있고, 사용자 정의를 통해 서비스에 알맞은 디자인을 고려하여 변환 규칙을 제정의 할 수도 있다. 그림 1은 여러 형태의 제공되는 서비스 정보들이 WAP 게이트웨이를 통하여 무선 네트워크로 전송하는 과정을 보여주고 있다. 기존에 구축되어 있는 인터넷 망을 통해 HTML을 전송 받는 경우 WAP 서버측이나 혹은 프록시 서버에 위치하고 있는 필터를 통하여 WML로 정보가 바뀌어 전송하는 모습을 볼 수 있다. 필터의 역할을 살펴보면 다음과 같은 두 단계의 과정으로 나누어 살펴볼 수 있다: (i) 태그

제거와 치환 등의 방법을 통한 HTML 문서의 WML 문서로의 변환과정과 (ii) 용량이 큰 문서의 변환과정에서 생성된 WML 문서를 WAP 단말기의 적은 메모리를 고려하여 데크나 카드등의 알맞은 사이즈로 분할 하는 과정으로 나누어 진다.



[그림 1] 무선 인터넷 프로그래밍 모델

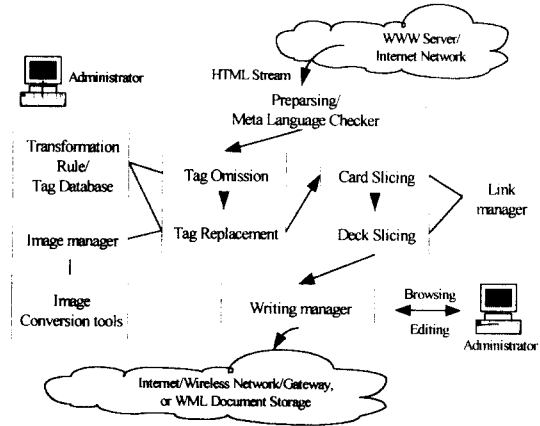
2.1 HTML 문서의 태그 변환

HTML 문서는 HTML 태그와 그에 속한 데이터로 나눌 수 있다. HTML 태그는 변환과정 중에 불필요한 태그의 제거와 다른 태그들로의 치환과정을 거치게 된다. 이러한 치환이나 제거에 관한 규칙은 개발자나 시스템 관리자에 의해 만들어 질 수 있는데, 변환 후에 브라우징 되는 화면과 서비스의 내용을 고려하여 규칙을 생성하면 된다. 생성된 규칙에 의해 입력되는 태그들은 각각의 변환 과정을 거치게 된다. 다음은 변환 플랫폼에서 실행되는 여러 단계의 변환과정에 대해서 설명하고 있다.

- 이미지 변환과 텍스트의 재구성: 이미지의 경우 웹 상에서 표현된 이미지의 컬러 농도를 재구성 할 필요가 있다. 만약 이미지를 지원하지 않는 단말기의 경우 이미지를 텍스트 링크로 대체 할 필요가 있다. 하지만 이미지가 지원되는 단말기의 경우 우선 웹상에서 사용되고 있는 이미지를 8비트나, 4비트 혹은 2비트, 1비트의 농도로 줄여줄 필요가 있고, 이미지의 크기 역시 재구성 되어야 한다. 파일 형식 또한 단말기에서 브라우징 될 수 있는 WBMP등의 형식으로 변환해야 한다. 또한 텍스트만 지원하는 단말기에서 테이블의 셀은 텍스트 형태로 재구성 되어야 하고 긴 텍스트의 경우 약어로서 표시하여 전체의 문서에 대한 링크 삽입을 통하여 해결한다.
- 치환: HTML 상에서만 사용되는 태그는 WML 상에서 유사하게 사용되는 태그로의 전환이 필요하다. 즉 <BLINK>와 같은 태그는 과 같은 태그로의 치환이 필요하다.
- 제거: <FRAME>이나 <OBJECT>등의 특정 태그나 불필요한 태그들에 대해서는 제거 과정이 있어야 한다. 또한 태그들의 속성들에 대해서도 제거 작업이 필요하다. 무선 단말기 상에서 필요 없는 애플릿이나 스크립트 코드도 제거된다. 문서의 크기를 줄이기 위해 주석문에 대한 제거도 필요하다. 이미지 지원을 하지 못하는 단말기를 위해서 이미지 표시 생략과정 역시 필요하게 된다.

그림 2는 WML 변환 플랫폼 중 태그 처리기의 전체적인 처리

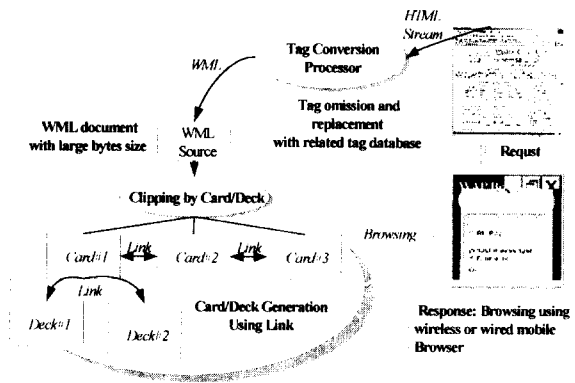
과정인데 위에서 언급한 과정이 모듈별로 설계되어 있다.



[그림 2] HTML 문서 구성요소의 치환과 분할, 그리고 제조함

2.2 카드와 데크 단위의 분할

대부분의 WAP 지원 단말기들의 적은 메모리 용량에 맞추기 위해서는 변환되는 페이지들을 일정 단위의 데크와 카드로 나눌 필요가 있다. 변환된 WML 페이지가 2000 바이트 보다 큰 카드에 의해서 생성이 되지 않도록 해야 하며, 모든 WAP 지원 단말기들에 의해 브라우징 될 수 있도록 문서를 효과적으로 나누어 주어야 한다. 데크가 너무 큰 용량일 경우 일부 WML 브라우저가 한번에 하나의 카드를 선택하여 다운로드 받는 것 대신에 전체 데크를 다운로드 받는 시도를 계속하게 되는데, 이러한 문제를 해결하기 위해서는 변환된 WML 문서를 데크 단위로 나누어야 한다. 만약 입력되는 HTML 문서의 용량이 적을 경우 별다른 분할 없이 그대로 브라우저로 출력될 수 있지만, 변환되는 HTML 문서의 용량이 크다면 이를 WAP 브라우저의 용량에 맞게 나누어야 한다. 그림 3은 이러한 과정을 설명하고 있다.

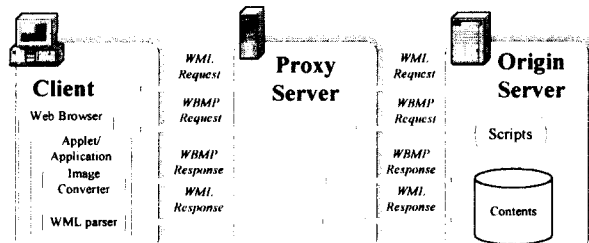


[그림 3] 변환된 WML 문서가 분할되어 브라우징 되는 과정

3. 사용자 에이전트 구현

서비스 제공자는 HTML 변환 플랫폼을 거쳐 변환된 문서들이

WAP 단말기의 마이크로 브라우저상에 올바르게 나타나지 확인하는 과정을 거쳐야 할 필요성이 있다. 이를 위한 중에 하나가 WAP 전용 애플레이터이다. 서비스 관리자는 애플레이터를 통해 WML 페이지가 어떻게 브라우징 되는지를 확인 할 수 있다. 또한 변환된 WML 페이지의 디자인이나 내용의 수정이 필요한 경우, 서비스 관리자는 제공되는 간단한 편집기를 사용하여 편집을 할 수 있다. 편집을 통해 저장된 WML 문서는 서버 내에 저장된다. 만약 다음 사용자에게 의해 페이지가 요청되는 경우 서버내의 캐쉬 기능으로 좀 더 빠른 응답을 통해 서비스 할 수 있도록 한다. WAP 브라우저와 같은 WML 사용자 에이전트의 구현은 여러 형태의 구조 모델로 정의되어 질 수 있다. 각각의 모델로 나누어지는 기준은 일반적으로 두 가지의 정보 처리 흐름에 기인하는데, 클라이언트에서 요청한 정보가 텍스트이고 이를 처리인 경우와 또 다른 경우는 이미지 처리인 경우이다. 이미지처리의 경우 무선 환경을 위한 WBMP의 이미지를 웹을 통해 볼 수 있도록 웹에서 지원하는 JPG나 GIF와 같은 그림 형태로 변환하여 웹 브라우저나 혹은 애플릿을 이용한 브라우저 상에서 볼 수 있도록 하는 과정을 말하며 처리 모듈의 실행 위치를 고려하여 설계를 나눌 수 있다. 그림 4은 처리 모듈이 클라이언트에 위치한 경우이다.



[그림 4] 사용자 에이전트 구현을 위한 모델

이 모델의 특징은 자바의 이식성을 기반으로 서버쪽의 중요 모듈을 클라이언트가 다운로드 받아 수행한다는 것이다. 자바 애플릿의 기능을 이용하여 좀 더 풍부한 디자인이 가능할 수 있고 응용 프로그램으로 제작되어지는 경우 자바의 java.awt와 같은 패키지나 스윙(Swing)등을 통해 세련되고 풍부한 인터페이스 표현이 가능하다는 것이 장점이 있다. 여기서 프록시 서버는 단순히 요청되고 응답되는 정보를 넘겨주게 되게 된다. 또한 WML 파싱이 클라이언트 상에서 이루어지기 때문에 분산 컴퓨팅을 이용한 효율적인 점도 생각할 수 있다.

제안한 모델과 달리 이미지와 텍스트 처리 모듈을 서버측에 위치한 경우가 있을 수 있는데 이는 구현에 있어 용이 할 수 있으나 모든 처리과정을 서버가 담당해야 하는 부담이 있고, 또한 서버에서 제공하는 기능을 클라이언트가 활용하기에는 한정한 것이 문제가 될 수 있다.

이미지 변환은 요청된 이미지를 서버로부터 가져와 WAP 브라우저가 인식할 수 있는 GIF 파일로 변환하는 과정으로 구성되어 있다. 이때 사용되는 이미지 변환 프로그램은 상용화되거나 배포된 그래픽 처리 유틸리티로 대신해서 사용할 수 있다. 변환된 GIF 파일은 다른 형태의 파일보다 압축율이 높기 때문에 적은 용량을 지니고 있는 브라우저를 고려해야 한다. 만약 브라우저가 인식할 수 있는 크기가 넘게 되면 변환과정을 수행하지 않고 단순히 링크 정보를 이용하여 대처 할 수 있다. 여기서 제안된 사용자 에이전트 프로그램 내의 WAP 브라우저는 유선상에서 사용되며 클라이언트 내에서 실행되므로 비교적 메모리의 제한을 받지 않게 되므로 실제 무선 단말기 보다는 많은 그림을 보여줄 수 있

도록 설계할 수 있다. 그러나 실제 서비스될 내용에 대해 검증하는 과정이므로 실제 무선 단말기의 적은 용량을 고려하였다.

4. 결론

본 문에서는 이미 기존에 제작된 HTML 기반의 문서들을 일정한 규칙에 의해 WAP 단말기에서 브라우징하기 적합한 언어인 WML로 바꾸어주는 실시간 변환시스템과 WML로 이루어진 콘텐츠가 실제 WAP 장비에서 브라우징 할 수 있는 효과를 나타내고 이를 일부 수정 편집할 수 있는 WAP 사용자 에이전트를 실시간 환경을 고려하여 설계하고 구현하였다. 현재 급증하는 무선 인터넷 시장을 고려한다면 앞으로도 많은 무선 콘텐츠가 생성될 것이고, 활성화에 따른 무선 인터넷 영역과 유선 인터넷 영역의 공유영역은 증가하리라 기대된다. 본 논문에서 제안하고 구현한 변환 플랫폼과 사용자 에이전트 시스템은 무선편환을 위해 웹 문서물 재구성의 과정 없이 그대로 사용하고 그 내용을 확인할 수 있는 특징을 가지고 있다. 향후 연구 과제로서 무선 인터넷 환경에서 텍스트와 이미지 뿐 아니라 다양한 유선상의 멀티미디어 데이터의 전송과 처리에 관해 연구 하는 것이다.

참고문헌

- [1] [WAE] Wireless Application Environment Specification, WAP Forum, November 4, 1999. URL: <http://www.wapforum.org/>.
- [2] [WAP] Wireless Application Protocol Architecture Specification, WAP Forum, April 30, 1998. URL: <http://www.wapforum.org/>.
- [3] [WDP] Wireless Datagram Protocol Specification, WAP Forum, November 5, 1999. URL: <http://www.wapforum.org/>.
- [4] [WML] Wireless Markup Language Specification, WAP Forum, November 4, 1999. URL: <http://www.wapforum.org/>.
- [5] [WMLScript] WMLScript Language Specification, WAP Forum, November 4, 1999. URL: <http://www.wapforum.org/>.
- [6] [WSP] Wireless Session Protocol Specification, WAP Forum, November 5, 1999. URL: <http://www.wapforum.org/>.
- [7] [WTA] Wireless Telephony Application Specification, WAP Forum, November 8, 1999. URL: <http://www.wapforum.org/>.
- [8] [WTAI] Wireless Telephony Application Interface Specification, WAP Forum, November 8, 1999. URL: <http://www.wapforum.org/>.
- [9] [WTP] Wireless Transaction Protocol Specification, WAP Forum, November 6, 1999. URL: <http://www.wapforum.org/>.
- [10] [WTLS] Wireless Transport Layer Security Specification, WAP Forum, November 5, 1999. URL: <http://www.wapforum.org/>.
- [11] [XML] Extensible Markup Language 1.0, W3C, October 6, 2000. URL: <http://www.w3.org/>.
- [12] RFC2616, Hypertext Transfer Protocol—HTTP/1.1, URL: <http://www.w3.org/Protocols/rfc2616/rfc2616.html>.