

웹서비스를 위한 모바일 웹 콘텐츠 변환 시스템 개발

홍인숙*, 김윤중**, 김승연**

*한밭대학교 컴퓨터공학과 **한밭대학교 컴퓨터공학과 교수님
indy15@freechal.com yjkim@hanbat.ac.kr, sykim@hanbat.ac.kr

A development of the Conversion System of Mobile Web Contents for the Web Service

In-Suk Hong*, Yoon-Joong Kim*, Seung-Yeon Kim*

*Dept of Computer Engineering, HabBat University

**Dept of Computer Engineering, HabBat University

요 약

본 논문에서는 WML Page로부터 닷넷 모바일 Web Page로의 변환 기능을 제공하는 웹서비스 프로바이더를 설계·구현하였다. 웹서비스 프로바이더는 플랫폼과 프로그래밍언어의 제약 없이 누구나 사용 가능하다는 장점을 가지고 있다. 본 논문에서 구현한 웹서비스 프로바이더로 제공하는 변환 라이브러리는 어휘분석기 모듈과 구문 지향 번역기 모듈로 구성되어 있다. 어휘분석기 모듈은 WML Page를 입력받아 각 태그들을 분석하고 토큰을 출력한다. 구문 지향 번역기 모듈은 어휘분석기에서 출력된 토큰들을 입력받아 문법을 체크하고, parsing이 완료되면 대응되는 모바일 .NET 컨트롤로 변환하여 최종적인 닷넷 모바일 Web Page를 생성한다. 어휘분석기 모듈과 구문 지향 번역기 모듈은 각각 어휘분석기 생성기(LEX)와 파서 생성기(YACC)를 사용하여 구현하였다. LEX와 YACC의 사용은 태그를 확장할 경우 변경된 부분만 설계함으로써 유연하게 대처할 수 있는 장점을 가지고 있다.

1. 서론

현재 무선 통신 언어들이 무선 단말기의 브라우저에 따라 각기 다르기 때문에 현재 무선 단말을 위한 웹 콘텐츠를 제공하기 위해서 개발자들은 다양한 단말에 맞는 웹 콘텐츠를 제작성 해야 한다[1]. 별도의 변환 시스템 없이 각각의 단말의 종류별로 웹 콘텐츠를 생성하는 방법은 사용자의 만족도가 가장 높은 웹페이지를 보여줄 수 있다. 그러나 이것은 단말의 특성에 따라 별도의 콘텐츠를 중복 제작해야하는 단점을 지니고 있다. 이는 투자비용의 증가와 함께 개발 회사의 개발 기간이 증가하게 되는 부작용을 가져온다. 이러한 문제를 해결하기 위해 여러 기업들은 기존에 있는 콘텐츠를 원하는 포맷으로 변환하여 주는 '컨버팅 솔루션'을 개발 중에 있다. 상용제품으로는 UNIWS의 'Mobile Converter'와 바우바우닷컴의 'BXT'이 있으며 기존에 연구되었던 논문으로는 "무선전용 다중 언어의 번역을 지원하는 변환기의 구현"이 있다[2-4].

이와 같은 대부분의 연구들은 각 마크업 언어마다 서로간의 변환기를 구현해야하는 어려움이 있다. 이에 본 논문에서는 상기의 문제점을 보완하고자 WAP기반의 WML Page를 닷넷의 모바일 Web Page[5]로 변환하는 시스템을 구현하고자 한다. 또한 변환시스템을 별도의 구현 없이 누구나도 사용할 수 있도록 SOAP기반의 웹서비스[6]로 제공하고자 한다.

본 연구에서는 (그림1)과 같은 시스템을 구현하였다. 클라이언트는 웹사이트 개발자이며, 웹서비스 소비자는 웹 서비스를 요청하는 기능을 포함한다. 웹서비스 프로바이더는 모바일 웹 페이지를 변환하는 변환 라이브러리를 제공한다. 본 시스템의 기능은 다음과 같다. 첫 번째 웹사이트 개발자인 클라이언트가 웹페이지를 업로드 한다. 두 번째 웹서비스 소비자는 업로드 된 웹페이지를 바이너리 메시지 포맷인 DIME (Direct Internet Message Encapsulation)[7]을 이용하여 웹서비스 프로바이더

에게 전송한다. 마지막으로 웹서비스 프로바이더는 변환 라이브러리를 호출하여 웹페이지를 변환하고 웹서비스 소비자에게 전달한다.

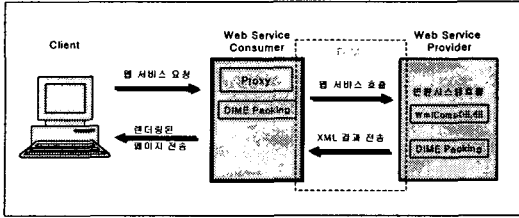


그림 1. 전체 시스템

본 논문의 2장에서는 웹서비스 프로바이더를 호출하기 위한 웹서비스 소비자의 구현 기술에 대해서 설명하고 3장에서는 변환시스템의 웹서비스 프로바이더의 구조에 대해, 4장에서는 모바일 Web Page로 변환하는 기능을 제공하는 변환 라이브러리의 구조와 구현 기술에 대해 설명한다. 5장에서는 실험 결과를 보이고 마지막으로 6장에서는 본 논문의 결론과 향후 추가로 연구될 과제를 제시한다.

2. 웹서비스 소비자 구현

본 연구에서 구현한 웹서비스 소비자는 먼저 생성된 프록시 클래스 sProxy를 사용하여 두 개의 메시지를 정의한다. 첫 번째 메시지를 사용하여 파일을 SOAP 메시지에 첨부하고 DIME으로 웹서비스를 요청하며, 두 번째 메시지를 사용하여 웹서비스 응답을 위와 같은 방법으로 받는다.

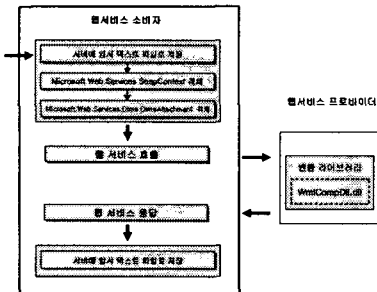


그림 2. 웹서비스 소비자

(그림2)와 같이 웹서비스 소비자는 먼저 클라이언트가 업로드 한 WML 웹페이지를 서버에 저장한다. MicrosoftWeb.Services에서 상속받은 SoapContext와 Dime.DimeAttachment 클래스의 객체를 생성하

여 업로드 파일을 SOAP 메시지에 첨부하고 변환 웹서비스 프로바이더를 호출한다. 마지막으로 호출한 웹서비스 프로바이더로부터 변환된 모바일 웹페이지를 수신 받아 클라이언트에게 전달한다.

3. 웹서비스 프로바이더 구현

본 연구에서는 웹서비스 소비자로부터 WML 파일을 입력받아 변환 라이브러리를 호출하고 .NET의 Web Page로 변환된 파일을 다시 요청한 웹서비스 소비자에게 전달하는 웹서비스 프로바이더를 구현하였다. 웹서비스 프로바이더는 먼저 Microsoft.Web.Services.SoapContext 클래스와 Microsoft.Web.Services.Dime.DimeAttachment 클래스의 객체를 생성하여 웹서비스 소비자로부터 웹 페이지를 전달받는다. 수신 받은 웹 페이지를 임시로 저장하고, 저장된 파일을 변환 라이브러리를 이용하여 변환한다. 마지막으로 그 결과 페이지를 DIME을 통해서 웹서비스 소비자에게 전송한다.

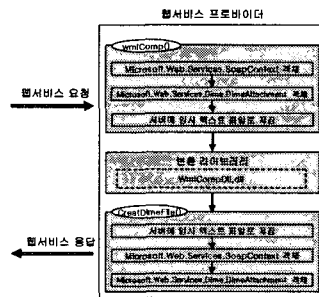


그림 3. 웹서비스 프로바이더

(그림3)과 같이 구현된 웹서비스 프로바이더는 wmlComp()와 CreateDimeFile()이라는 두 개의 웹 메시지를 제공한다. wmlComp()는 DimeAttachment 객체를 이용하여 웹서비스 요청 메시지를 수신 받고 첨부된 파일을 저장한 후 WmlCompDll.dll을 호출한다. WmlCompDll.dll은 WML Page를 닷넷의 Web Page로 변환해주는 라이브러리이다. 이에 대한 자세한 구조 및 동작에 대한 내용은 4장에서 설명한다. CreateDimeFile()은 변환된 결과를 요청한 웹서비스 소비자에게 전송하는 메시지이다.

4. 변환 라이브러리 구조

웹서비스 프로바이더에서 제공하는 변환 라이브러리의 구조는 (그림4)와 같이 어휘분석기 모듈과 구문 지향 번역기(SDD) 모듈로 구성되어 있다.

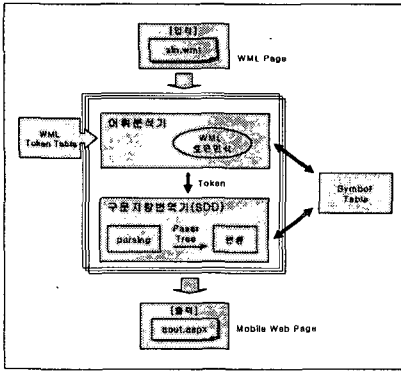


그림 4. 변환 시스템

어휘분석기 모듈은 WML 토큰 테이블과 확장 마크업 언어의 형식[8,9]을 기반으로 입력문서의 토큰을 인식한다. 각각의 태그들을 분석하고, 태그와 텍스트를 분리한 후 불필요한 태그들과 주석 등을 제거한다. 그리고 필요한 태그들만을 토큰으로 생성하여 구문 지향 번역기 모듈에게 전달한다. 구문 지향 번역기 모듈은 어휘분석기에서 받아온 토큰을 이용하여 각 태그에 해당하는 속성들과 문법을 체크(parsing)한다. parsing이 완료되면, 변환 규칙에 의해 각각을 대응되는 .NET 모바일 컨트롤로 변환하여 최종적인 Mobile Web Page를 생성한다.

4.1 어휘분석기 모듈

어휘분석기 모듈은 먼저 각각의 토큰에 대한 패턴을 정의하고 정의된 패턴에 따라 입력 문자들을 태그·속성·값을 나타내는 데이터로 분리하여 토큰을 반환하고 각각의 정보를 symbol table에 저장한다.

하나의 토큰을 인식하고자 할 때 우선 이 토큰이 주석인지 마크업인지 파싱된 데이터인지 구분되어야 한다. 예를 들어 card라는 토큰이 인식되었을 때, 마크업에 사용되었다면 deck 내의 하나의 카드를 인식하는 요소의 시작태그로 인식될 것이고, 파싱된 데이터에 사용되었다면 단지 문자 데이터로 인식될 것이다. 그리고 주석문 내에 사용이 되었다면 무시될 것이다. 어휘분석기 모듈은 하나의 토큰이 인식될 때 이 토큰이 주석 또는 마크업 그리고 데이터인지를 구분하기 위해 상태를 다섯 가지(CMT, CData, tagStart, attrStart, valueStart,)로 두어 인식한다.

어휘분석기는 모든 문자를 읽어 토큰을 생성하는 것뿐만 아니라 변환 코드 생성부분에서 사용될 토큰의 속성을 가지고 있는 symbol table을 생성한다.

4.2 구문 지향 번역기 모듈

본 시스템에서 구현한 구문 지향 번역기(SDD) 모듈은 어휘분석기 모듈에서 넘겨주는 토큰들을 받아 미리 정의된 문법에 따라 parsing table을 생성한다. 생성된 parsing table과 확장 마크업 언어의 형식을 참조하여 입력받은 문서의 유효성 여부를 판단한다. parsing이 완료되면 어휘분석기 모듈에서 만들어진 symbol table과 미리 정의한 변환규칙에 의해 입력된 문자들을 변환한다.

파서의 문법 구조는 LALR(1) 문법으로 작성되었고, 무선 마크업 언어 2.0 문서 형태 정의에 정의된 모든 엔티티들과 요소들의 속성 및 속성 값, 그리고 내용에 대하여 parsing 할 수 있도록 설계하였다.

<표1>은 문서 형태 정의 선언에 의해 무선 마크업 언어 요소 중 <anchor>요소의 XML DTD 예이다.

표 1. <anchor> 요소의 XML DTD

1	<ELEMENT ANCHOR (%INLINE: 1 DO PREV REFRESH) * >
2	<!ATTLIST ANCHOR
3	TITLE %vdata; #IMPLIED
4	>

라인1은 엔티티 %inline과 <task> 요소들을 내용으로 포함하는 <anchor>요소를 선언하고 있다. 라인 2~4에서는 속성을 나열하고 있는데 <anchor>이 가질 수 있는 속성은 title 뿐이다. 속성 title의 타입은 엔티티 %vdata 이고, 디폴트 값은 없으며, 필수 값이 아니다. 이와 같은 내용을 검증하기 위한 파서의 문법은 <표 2>와 같다.

표 2. <anchor> 요소의 grammar 의 예

1	anchor_statement : '<' T_ANCHOR attr_anchor '>' go_statement '<' '/' T_ANCHOR '>'
2	attr_anchor : attr_anchor attr_anchor1 '=' value_statement
3	1 attr_anchor1 '=' value_statement
4	attr_anchor1 : A_TITLE
5	go_statement : '<' T_GO attr_go '/' '>'
6	;

5. 실험 및 결과

5.1 구현환경

운영체제는 Windows 2000 Server를 사용하였으며, 윈도우즈용 LEX와 YACC[10,11]를 이용하여 변환시스템을 구현하였다. 본 시스템을 웹서비스로 제공하기 위해 ASP.NET 과 C#을 이용하였다.

5.2 실험 및 결과

본 연구에서 구현된 시스템의 성능을 측정하기 위

해서 Visual Studio .NET 환경에서 실험하였다. 먼저 클라이언트는 (그림 5)과 같이 WML Page (tag2.wml)를 웹서비스 소비자의 변환 웹페이지에 업로드 시킨다. 웹서비스 소비자는 업로드 된 파일을 SOAP 메시지에 첨부한 후 웹서비스 프로바이더를 호출한다. 호출된 웹서비스 프로바이더는 파일을 이용하여 변환 라이브러리를 호출하고 변환된 결과를 요청한 웹서비스 소비자에게 전송한다. 웹서비스 소비자는 변환된 결과를 다시 클라이언트에게 전송하여 (그림 6) 과 같이 화면으로 직접 보여준다. 출력된 결과를 확인한 후 클라이언트가 원하는 컴퓨터에 다운로드받을 수 있도록 하였다.

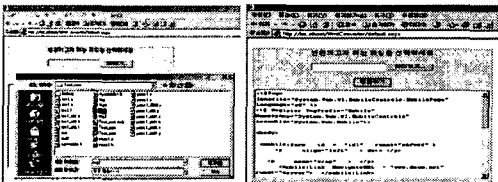


그림 5. 업로드 화면 그림 6. 변환 결과 화면

(그림7-a)는 실험 대상이 된 WML Page를 에뮬레이터로 실험한 결과이다. (그림7-b)는 구현된 시스템의 출력 결과인 out.aspx를 WML Emulator를 통해 실행시킨 결과를 나타낸 것이다. (그림7-a)와 (그림7-b)를 비교한 결과 같은 페이지가 동작되는 것을 알 수 있다.

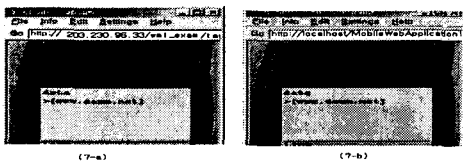


그림 7. 시뮬레이션 결과화면

6. 결론

본 논문에서는 마크업 언어 중 가장 많이 사용되고 있는 WML Page를 .NET Web Page로 변환하는 기능을 제공하는 웹서비스 프로바이더를 구현하였다. 모바일 Web Page는 요구되는 사용자의 환경에 맞게 자동으로 변환해 주기 때문에 WML로 변환하고자 할 때 별도의 변환 시스템을 구현할 필요가 없다. 또한 어휘분석기 생성기(LEX)와 파서 생성기(YACC)를 사용하여 변환 시스템을 구현함으로써

향후 태그를 확장시킬 경우 변경된 부분만 다시 설계하여 새로운 변환 시스템을 자동 생성할 수 있는 장점을 가진다.

본 연구에서 구현한 변환 시스템은 언제 어디서든 누구나 사용할 수 있도록 SOAP기반의 웹서비스 형태로 구현하였으므로 클라이언트들은 별다른 애플리케이션 설치 없이 사용할 수 있다. 또한 웹서비스를 요청하고 응답 받는 과정을 처리하는 웹서비스 소비자를 구현함으로써 클라이언트는 간단하게 파일을 업로드하고, 변환된 파일을 다운로드받을 수 있다.

추후 현재 프로토타입으로 구현되어있는 본 시스템을 더욱 구체적으로 설계하고 구현하여 성능을 향상시키고자 한다. 또한 다양한 스크립트 언어(ASP, PHP, JSP 등)와 함께 사용된 동적인 WML Page에 대한 변환 기능의 추가와 함께 웹 상에서의 보안성에 대한 문제도 고려해 보고자 한다.

[참고문헌]

- [1] 한동일, 김미연, 김기수, "무선 인터넷 환경을 지원하기 위한 콘텐츠 변화 미들웨어 시스템 연구", 한국멀티미디어학회, 2001, p42~45
- [2] <http://www.mnetsoft.com>
- [3] <http://baubau.net/main.htm>
- [4] 최지원, 김기천, "무선전용 다중 언어의 번역을 지원하는 변환기의 구현", 정보처리학회, 제9-권, 2002, p293~296
- [5] Andy Wigley, Peter Roxburgh, 문봉재 역, 「모바일 디바이스에서 닷넷 애플리케이션 구축하기」, 정보문화사, 2003
- [6] 러스 베슈라 외, 「Professional ASP.NET Web Services」, 정보문화사, 2003
- [7] Specification:Direct Internet Message Encapsulation(DIME) Version 17-June-2002 (<http://www.ibm.com/developerworks/ws-dime/>)
- [8] "Wireless Markup Language", WAP Forum, Version 11-Sep-2001, (<http://www.wapforum.org>)
- [9] 홍준호 외 2, 「about WAP」, 영진.COM, 2001
- [10] Alfred V.Aho, Ravi Sethi, Jeffrey D. Ullman, 「Compilers:Principles, Techniques, and Tools」, Addison-Wesley Pub Co, 1985
- [11] John R. Levine, Tony Mason, Doug Brown, 「Unix Programming Tools: lex & yacc」, Reilly & Associates, 1992