

# WML Page를 .NET 모바일 웹 페이지로 변환하는 웹 서비스 구현

홍인숙<sup>\*</sup>, 김윤중<sup>\*\*</sup>, 김승연<sup>\*\*\*</sup>

## 요 약

본 논문에서는 WML 페이지를 닷넷 모바일 웹 페이지로 변환하는 웹 서비스를 설계·구현하였다. 웹 서비스는 플랫폼과 프로그래밍언어의 제약 없이 누구나 사용 가능하다는 장점을 가지고 있다. 본 논문에서는 별도로 제공되는 변환 시스템을 제공하는 웹 서비스 제공자와 이를 사용하는 웹 서비스 소비자를 함께 구현하였다. 웹 서비스를 호출하는 웹 서비스 소비자를 구현함으로써 클라이언트(개발자)들은 별다른 작업 없이 웹 서버를 통해 웹 서비스를 호출하여 닷넷 모바일 웹 페이지로 변환된 결과를 받아 볼 수 있다. 또한 웹 서비스의 구현 방법을 고려하지 않고도 이를 사용하여 새로운 응용프로그램을 구현할 수 있다.

## An Implementation of the Web Services that Convert WML Page Into .NET Mobile Web Page

In-Suk Hong<sup>\*</sup>, Yoon-Joong Kim<sup>\*\*</sup>, Seung-Yeon Kim<sup>\*\*\*</sup>

## ABSTRACT

In this paper, we describe a design and implementation of Web Services that convert WML(Wireless Markup Language) Page into .NET Mobile Web Page. The Web Services has an advantage that is available everybody without restriction of platform and programming language. In this paper, we implement the Web Services Provider providing of converting system and the Web Services Consumer using this system. The Clients can get results of .NET Mobile Web Page without difficulty through Web Server because of implementation of Web Services Consumer using Web Services. Also the clients can develop new application using Web Services without respect of its implement process.

**Key words:** Mobile(모바일), Compiler(컴파일러), WML, .NET(닷넷)

## 1. 서 론

무선 통신의 급격한 성장세와 함께 모바일 서비스는 점차 필수적인 것으로 여겨지고 있으며 이에 따라

웹 콘텐츠 개발도 증가되고 있다. 현재 무선 통신 언어들은 무선 단말기의 브라우저에 따라 다르기 때문에 개발자들은 다양한 단말기의 브라우저에 맞게 웹 콘텐츠를 재 작성해야 한다[1]. 별도의 변환 시스템 없이 각각의 브라우저에 맞는 웹 콘텐츠를 생성하는 것은 사용자의 만족도가 가장 높은 콘텐츠를 제작할 수 있지만 단말기의 특성에 따라 별도의 콘텐츠를 중복 제작해야하는 단점을 지니고 있다. 이는 투자비용의 증가와 함께 개발 회사의 개발 기간의 증가를 초래한다. 이러한 문제점을 해결하고자 여러 모바일 비즈니스를 추진하는 기업들이 기존에 있는 콘텐츠를 원하는 포맷으로 변환하여 주는 '컨버팅 솔루션'

※ 교신저자(Corresponding Author) : 홍인숙, 주소 : 대전광역시 유성구 덕명동 산 16-1번지(305-719), 전화 : 042) 821-1143, E-mail : ishong@hanbat.ac.kr

접수일 : 2003년 11월 4일, 완료일 : 2004년 9월 10일

<sup>\*</sup> 준회원, 한밭대학교 대학원 컴퓨터공학과 박사과정

<sup>\*\*</sup> 중신회원, 한밭대학교 교수

(E-mail : yjkim@hanbat.ac.kr)

<sup>\*\*\*</sup> 한밭대학교 교수

(E-mail : sykim@hanbat.ac.kr)

을 개발 중에 있다. 그러나 이러한 상용 제품들을 사용하기 위해서는 솔루션을 다운로드 받아 개발자의 컴퓨터마다 설치하여야 한다. 이때 솔루션이 개발된 플랫폼과 이를 사용하는 개발자의 플랫폼은 서로 같아야 한다. 만약 다르다면 추가적인 비용이나 시간을 들여 플랫폼 환경을 다시 구축해야만 한다. 기존의 솔루션을 이용하여 새로운 애플리케이션을 구현하는 것 또한 쉽지가 않다.

본 논문에서는 기존에 개발하였던 변환 시스템[2]을 웹 서비스로 제공함으로써 변환 시스템이 개발된 플랫폼에 상관없이 다양한 종류의 클라이언트에게 동일한 서비스를 제공할 수 있도록 하였다. 또한 이를 사용하기 위해 클라이언트들은 솔루션을 따로 설치할 필요 없으며 어디서든 웹을 통하여 사용할 수 있도록 하였다.

본 연구에서는 변환 시스템을 SOAP(Simple Object Access Protocol)[3]기반의 웹 서비스로 제공한다. 변환 시스템은 WML 페이지를 변환하여 닷넷 모바일 웹 페이지를 출력한다. 닷넷 모바일 웹 페이지[4]는 콘텐츠를 요구하는 클라이언트들의 환경에 맞게 자동으로 렌더링하여 제공하는 장점을 가지고 있다. 또한 웹 서비스를 요청하고 응답 받는 과정을 처리하는 웹 서비스 소비자를 구현하여 여러 클라이언트(개발자)들이 별다른 과정 없이 간단히 웹 서비스를 이용할 수 있도록 하였다.

본 시스템은 그림 1과 같이 클라이언트와 웹 서비스 소비자 그리고 웹 서비스 제공자로 구성으로 되어 있다. 첫 번째 모바일 콘텐츠 개발자인 클라이언트는 변환하고자 하는 웹 페이지를 업로드 한다. 두 번째 웹 서비스 소비자는 웹 서비스를 요청하고 웹 서비스 결과를 수신 받는 웹 사이트 서버이다. 본 연구에서는 웹 서비스를 요청하기위해 업로드 된 웹 페이지를 바이너리 메시지 포맷 인 DIME(Direct Internet

Message Encapsulation)[5]를 이용하여 웹 서비스 제공자에게 전송한다. 또한 웹 서비스 제공자로부터 수신 받은 결과를 클라이언트에게 제공한다. 세 번째 웹 서비스 제공자는 변환 시스템을 호출하여 웹 페이지를 변환하고 그 결과를 웹 서비스 소비자에게 반환한다.

본 논문의 2장에서는 관련연구에 대해 기술하고 3장에서는 웹 서비스 소비자의 구현 기술에 대해 기술하며 4장에서는 변환 웹 서비스 제공자의 구조와 구현 기술에 대해 기술한다. 5장에서는 실험 결과를 기술한다. 마지막으로 6장에서는 본 논문의 결론과 향후 추가로 연구될 과제를 제시한다.

## 2. 관련연구

### 2.1 개발사례

누구든지 언제 어느 곳에서나 무선 인터넷을 이용할 수 있게 됨으로서 다양한 무선 콘텐츠의 요구가 급증하였다. 이에 여러 모바일 비즈니스를 추진하는 기업들은 기존에 있는 콘텐츠를 원하는 포맷으로 변환하여 주는 ‘컨버팅 솔루션’을 개발 중에 있다.

상용 제품으로는 SML이라는 중간언어를 사용하여 원하는 문서의 포맷으로 변환시키는 유니위스(UNIWIS)사의 ‘Mobile Converter’와 중간 언어의 변환 없이 직접 변환시키는 모바일 단말 언어 간 자동 변환 솔루션인 바우바우닷컴의 ‘BXT (Baubau Cross Translator)’ 등이 있다[6,7]. 그리고 기존에 연구되었던 논문에는 “무선전용 다중 언어의 번역을 지원하는 변환기의 구현”[8]이 있다. 위 논문에서는 두 개의 변환기를 구현하여 c-HTML과 m-HTML의 콘텐츠를 WML로 직접 변환하고 WAP[9] 기반의 웹 콘텐츠 개발자에게 제공한다.

위와 같은 솔루션을 사용하기 위해서 개발자들은 솔루션이 개발된 환경에 맞게 사용 환경을 맞추어야 하며 추후 변경된 사항들에 대해서는 추가적으로 구입을 해야 하는 번거로움과 함께 추가적인 비용도 증가하게 된다.

본 논문에서는 기존에 개발하였던 변환 시스템을 웹 서비스로 제공함으로써 플랫폼에 독립적이며 웹을 통해 어디에서든지 쉽게 사용할 수 있는 편리성을 제공하고자 한다. 추후 변경되거나 다른 기능이 추가되어도 이를 사용하는 클라이언트는 부가적인 비용

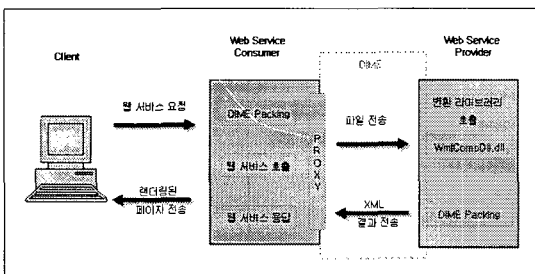


그림 1. 전체 시스템

이나 시간을 들이지 않고 사용할 수 있다. 또한 새로운 애플리케이션을 구현할 경우에 서비스 구현 방법을 고려하지 않고도 서비스 기능을 재사용할 수 있다.

### 2.2 웹 서비스(Web Services)

웹 서비스는 이기종 플랫폼에 탑재된 서로 다른 애플리케이션들 간에 데이터 통신기능을 이용하여 작업을 자동화할 수 있는 서비스 통합 기술이라 말할 수 있다[10]. 기기와 프로그램에 상관없이 SOAP을 통해 원하는 기능을 원격에서 수행할 수 있게 된다. 즉, SOAP을 통해 원격 컴퓨터에 구현된 기능을 마치 자신의 기능처럼 사용할 수 있게 된다.

웹 서비스 아키텍처는 그림 2와 같이 서비스 제공자 (Service Provider), 서비스 소비자 (Service Consumer), 서비스 중개자 (Service Broker)와 같은 세 가지 역할들의 상호작용을 기반으로 한다. 서비스 제공자는 비즈니스 관점에서 보면 서비스 소유자이며 아키텍처 관점에서 보면 서비스가 운영되는 플랫폼이다. 서비스 소비자는 비즈니스 관점에서 특정 서비스를 요구하는 비즈니스이며 아키텍처 관점에서는 서비스를 찾고 호출하는 어플리케이션이다. 서비스 중개자는 서비스 제공자가 출판하는 서비스 기술 (Service Description)들을 관리하고 서비스 소비자에게 서비스 검색 서비스를 제공한다.

그림 3에서처럼 웹 서비스를 검색 및 액세스하는 일련의 과정을 살펴보면 다음과 같다.

- ① 웹서비스 제공자가 웹 서비스를 만든다.
- ② WSDL (Web Service Description Language) 로 정의한 다음 웹 서비스 디렉토리에 게시한다.

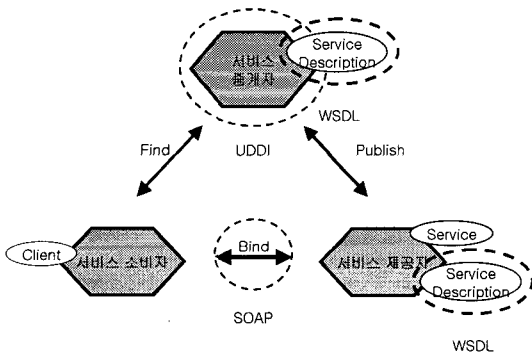


그림 2. 웹 서비스 아키텍처

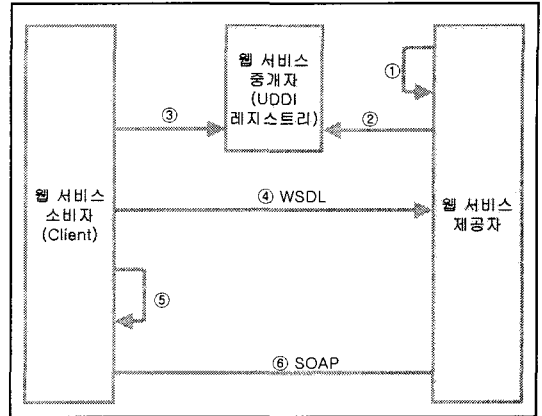


그림 3. 웹 서비스 일련과정

- ③ 이용자가 웹 서비스 디렉토리로 질의한다.
- ④ 이에 대한 응답은 요청된 서비스의 WSDL 기술자(웹 서비스 설명)를 포함한다.
- ⑤ 이용자가 WSDL 기술자의 정보를 사용하여 프록시 클라이언트를 생성한다.
- ⑥ 웹 서비스를 호출한다(SOAP).

일반적으로 웹 서비스 활용을 위해서는 먼저, 웹 서비스를 찾고 원하는 웹 서비스를 찾았다면 그 이용 방법을 알아 실제로 이용하는 과정을 거치게 된다. 웹 서비스를 찾기 위해서는 UDDI(Universal Description, Discovery and Integration)를 사용하고 서비스 이용방법을 알기위해 WSDL을 사용한다. 그리고 실제로 이용하기 위한 객체간의 통신규격으로 SOAP이 사용되고 있다[11-14].

### 2.3 WSDL(Web Service Description Language)

SOAP 메시지 집합 및 해당 메시지가 교환되는 방법을 설명하는 XML 문서이다. WSDL은 메시지 콘텐츠를 설명할 뿐만 아니라 서비스를 사용할 수 있는 위치 및 서비스와 대화하는데 사용되는 통신 프로토콜을 정의한다[15].

### 2.4 SOAP(Simple Object Access Protocol)

분산된 환경에서의 정보를 교환하기위한 프로토콜로 HTTP의 대중성과 XML의 유연성을 기반으로 한다. SOAP은 서비스 제공자와 사용자 사이의 교환 내용을 최소화한 경량(Light Weight) 프로토콜이며

최근 1.2 버전이 W3C Recommendation으로 승인되었다. SOAP은 액세스 요구나 리턴되는 결과 값으로 XML를 사용함으로써 특정의 포맷의 제약이 없고, 유연성 높은 범용의 액세스 기능을 제공한다[3].

### 3. 변환 시스템의 웹 서비스 소비자 구현

웹 서비스 소비자는 웹 서비스 제공자가 제공하는 서비스를 사용하는 모든 응용프로그램과 웹 사이트 등을 포함한다.

본 연구에서는 변환 웹 서비스를 사용하는 웹 사이트를 ASP.NET을 이용하여 웹 서비스 소비자로 구현하였다.

구현된 웹 서비스 소비자는 그림 4와 같이 웹 서비스 요청 기능과 웹 서비스 응답 처리 기능을 가진다.

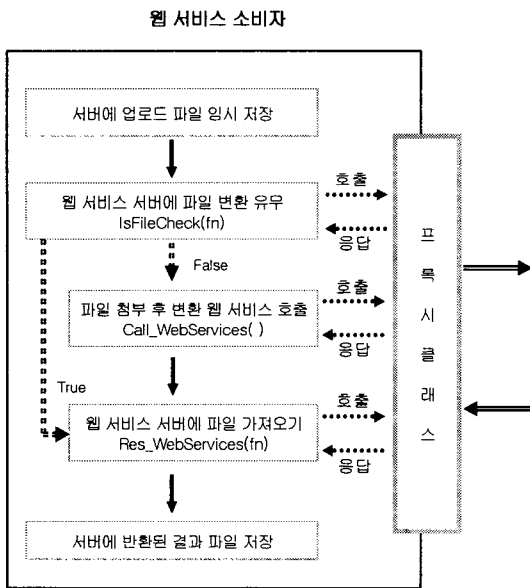


그림 4. 웹 서비스 소비자

#### 3.1 프록시 클래스

웹 서비스를 사용하기 위해서 웹 서비스 소비자는 프록시 클래스를 생성해야 한다. 웹 서비스의 코드를 알지 못해도 웹 서비스 로직을 구현할 수 있게 해주는 것이 바로 프록시 클래스이다. 프록시 클래스는 원격 웹 서비스를 참조하는 것을 가능하게 하고 웹 서비스가 제공하는 메시지를 정의하여 반환하는 데이터가 로컬에서 생성된 것처럼 응용프로그램 내에

서 웹 서비스의 기능을 사용 가능하게 한다. 프록시 클래스에서 정의된 메서드는 웹 서비스와 주고받은 SOAP 메시지를 처리하여 웹 서비스 제공자와 통신한다. 프록시 클래스는 WSDL과 WSDL.exe 도구를 이용하여 생성하거나 비주얼 스튜디오 닷넷 환경에서 제공하는 웹 참조를 통해 자동으로 생성할 수 있다[13].

본 연구에서는 두 번째 방법으로 그림 5와 같이 비주얼 스튜디오 닷넷 환경에서 제공하는 웹 참조 추가 대화상자를 통해 프록시 클래스(sProxy)를 생성하였다. 생성된 프록시 클래스는 파일의 유무를 체크하는 메소드 FileCheck()와 파일을 첨부하여 변환 서비스를 요청하고 응답받기 위한 메소드 WmlComp()와 메소드 CreateDimeFile()을 정의한다.

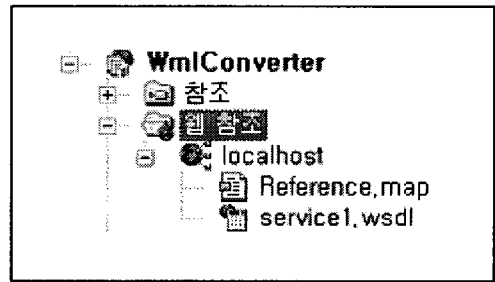


그림 5. ASP.NET 환경에서의 웹 참조

#### 3.2 웹 서비스 호출 및 응답

웹 서비스 소비자는 웹 서비스 요청 시 다음과 같은 기능들을 포함하고 있다.

- 1) 먼저 클라이언트가 업로드 한 WML 페이지를 서버에 저장한다.
- 2) 저장된 파일에 대한 변환 작업의 여부와 함께 변환된 결과 파일이 있는지를 체크한다. 아래에 있는 그림 6과 같이 메소드를 호출한다.

```
localhost.Service1 svc = new localhost.Service1();
TextBox2.Text = svc.fileCheck(fileName);
```

그림 6. IsfileCheck() : 웹 서비스 요청 1

- 3) 처음 요청한 파일이라면 Microsoft.Web.Services에서 상속받은 SoapContext와 Dime.

DimeAttachment 클래스의 객체를 생성하여 SOAP 메시지에 파일을 첨부한 후 메서드 (WmlComp())를 사용하여 웹 서비스를 요청한다. 요청하는 방법은 그림 7과 같다.

```

localhost.Service1Wse sProxy = new localhost.Service1Wse();

DimeAttachment attachment = new DimeAttachment
("text/plain", TypeFormatEnum.MediaType, input_filename);

sProxy.RequestSoapContext.Attachments.Add(attachment);

//웹 서비스 호출
downloadfile.Text = sProxy.wmlComp();
    
```

그림 7. Call WebServices() : 웹 서비스 요청 2

4) 만약 이미 변환되었던 파일이라면 변환시스템을 호출하는 메서드(wmlComp())가 아닌 웹 서비스 서버에 있는 파일을 가져오는 메서드(CreateDimeFile())를 호출한다. 그림 8과 같이 클라이언트가 요청한 파일 이름을 매개 변수로 넘겨준다.

```

WmlConverter,localhost.Service1Wse svc
= new WmlConverter,localhost.Service1Wse();

//웹 서비스 프로바이더에서 제공하는 CreateDimeFile() 호출
Svc.CreateDimeFile(inFN);
    
```

그림 8. Res WebServices(): 웹 서비스 요청 3

웹 서비스 요청이 성공적으로 이루어지면 웹 서비스 제공자는 변환된 파일을 반환한다. 웹 서비스 소비자는 최종적으로 닷넷의 모바일 웹 페이지가 첨부된 SOAP 메시지를 수신 받는다. 그림 9는 응답 SOAP 메시지의 일부를 나타낸 것이다.

```

<wmlCompResponse xmlns="http://203.230.96.33/WebService1/">
  <wmlCompResult>out1.txt</wmlCompResult>
</wmlCompResponse>
    
```

그림 9. 응답 SOAP Message 일부

웹 서비스 제공자로부터 DimeAttachment 객체를 통해 반환된 결과를 메모리에 저장한 다음 모니터에 첨부된 파일 내용을 출력한다. 그리고 클라이언트가 변환된 파일을 원하는 장소에 다운로드 할 수 있도록 다운로드 기능도 제공한다.

#### 4. 변환 시스템의 웹 서비스 제공자 구현

웹 서비스 기술은 웹 서비스 소비자 (Web Services Consumer)의 요청에 따라 원격 시스템에서 특정 기능을 수행하고 그 결과를 응답해 주는 서비스이다.

본 연구에서 구현한 웹 서비스 제공자는 웹 서비스 소비자에게 무선 모바일 웹 페이지 변환 서비스를 제공한다. 즉 웹 서비스 제공자는 웹 서비스 소비자로부터 WML 파일을 전달받아 변환 시스템을 호출하고 반환된 결과 값인 닷넷의 모바일 웹 페이지를 요청한 웹 서비스 소비자에게 재전송한다.

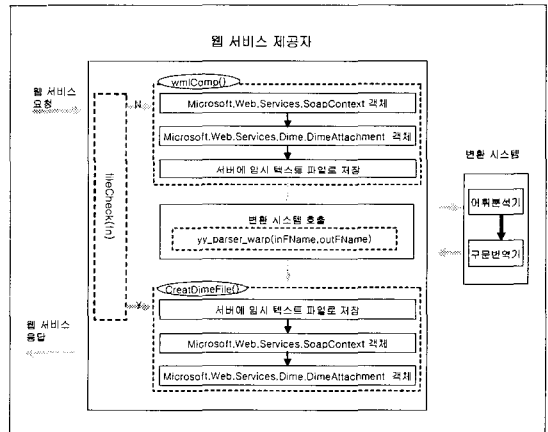


그림 10. 변환 웹 서비스 제공자

변환 시스템의 웹 서비스 제공자는 그림 10과 같이 세 개의 웹 메서드 (FileCheck() · WmlComp() · CreateDimeFile())로 구성하였다.

FileCheck()는 서비스로 요청한 파일이 이미 한번 요청된 적이 있는 파일인지의 여부를 확인하는 메서드이다. 만약 전에 요청되었던 파일이라면 변환시스템은 호출하지 않고 저장되어있는 결과 파일을 가져온다. 서버에 저장되어있는 변환된 결과 파일은 일정한 기간동안 파일에 대한 접근이 없는 경우 자동으로 삭제된다.

WmlComp()는 두 가지 기능을 하는 메서드이다. 첫 번째 기능은 Microsoft.Web.Services.SoapContext 클래스와 Microsoft.Web.Services.Dime.DimeAttachment 클래스의 객체를 생성하여 웹 서비스 소비자로부터 웹 페이지를 전달받는다. 전달받

은 SOAP 메시지와 첨부된 파일을 분석한 후 서버에 저장한다. 두 번째 기능은 첨부된 파일을 매개변수로 변환 시스템을 호출한다. 변환 시스템을 호출하기 위해 그림 11과 같이 Import 시킨다.

```
[DllImport(@"C:\inetpub\wwwroot\WebService\WmlCompDll.dll")]
public extern static void yparse_error(string inputFile, string outFile);
```

그림 11. 변환 시스템 호출을 위한 선언

CreateDimeFile()은 변환 시스템으로부터 변환된 파일을 Dime.DimeAttachment 클래스의 객체를 사용하여 SOAP 메시지에 첨부하고 요청한 웹 서비스 소비자에게 전송하는 메서드이다.

별도로 제공되는 변환 시스템은 WML 페이지를 매개변수로 받아 어휘분석기 모듈과 구문 지향 번역기 모듈을 사용하여 parsing과 변환과정을 거쳐 이에 대응되는 닷넷 모바일 웹 페이지를 반환하는 것이다. 변환 시스템은 LEX와 YACC[16]로 구현된 시스템이다.

## 5. 실험 및 결과

### 5.1 구현환경

본 논문에서 구현한 웹 서비스 시스템의 환경은 표 1과 같다.

구현된 시스템의 성능을 알아보기 위해서 HTML3.2 브라우저를 지원하는 PocketPC 애플레터와 WML 브라우저를 지원하는 Phone.com의 UPSDK 4.0 시뮬레이터, 그리고 HP IPAQ 5420 기종의 PDA를 사용하여 테스트 하였다. 또한 구현한 변환 웹 서비스의 기능을 테스트하기 위해 동작 과정을 살펴보았다.

표 1. 구현환경

	O/S	웹서버	서비스 플랫폼	구현 언어
웹 서비스 제공자	Windows XP Professional	IIS5.1	VS.NET & WSE2.0	ASP.NET & C#
변환 시스템				VS.NET · Lex & Yacc
테스트 환경		IIS5.1	VS.NET & M MIT	1.ASP.NET 2.WML

### 5.2 실험 및 결과

웹 서비스 형태로 제공되는 변환 시스템의 입력은 WML 파일이고, 출력은 변환된 ASPX 파일이다. 본 변환 시스템이 동작되는 과정은 다음과 같다.

- 1) 먼저 클라이언트는 그림 12와 같이 WML 페이지(selc.wml)를 웹 서비스 소비자의 변환 웹 페이지에 업로드 시킨다.
- 2) 웹 서비스 소비자는 업로드 된 파일을 SOAP 메시지에 첨부한 후 웹 서비스 제공자를 호출한다.
- 3) 웹 서비스 제공자는 수신 받은 파일을 가지고 변환 라이브러리를 호출하고 변환된 결과를 웹 서비스 소비자에게 전송한다.
- 4) 웹 서비스 소비자는 호출한 웹 서비스 제공자로부터 응답 받아 다시 클라이언트에게 전송하여 그 결과를 직접 볼 수 있도록 그림 13과 같이 화면에 보여준다.

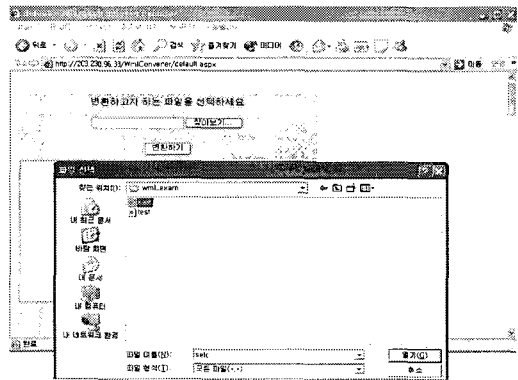


그림 12. 웹 페이지 : 업로드 화면

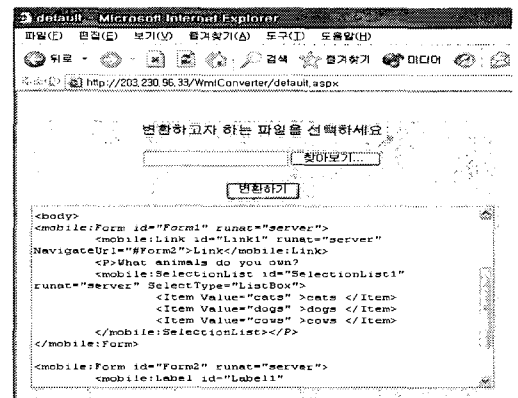


그림 13. 변환 결과 화면

5) 클라이언트는 화면에 출력된 결과를 확인하고 필요한 경우 원하는 컴퓨터에 다운로드 받는다.

본 논문에서 구현한 웹 서비스를 통해 WML 페이지를 보내고 그 결과 닷넷의 모바일 웹 페이지가 전송되어 오는 것을 확인하였다.

그림 14는 실험 대상이 된 WML 페이지를 시뮬레이터로 실험한 결과이다. 그림 15는 구현된 시스템의 출력 결과인 out.aspx를 WML 시뮬레이터를 통해 실행시킨 결과를 나타낸 것이다. 그림 14와 그림 15를 비교한 결과 같은 페이지가 동작되는 것을 알 수 있다.

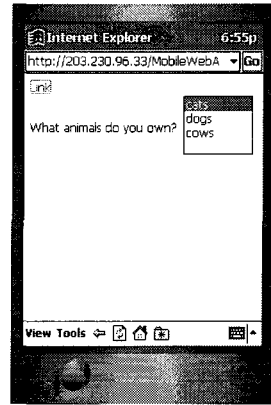


그림 17. Pocket Pc에서의 out.aspx 실행 화면



그림 14. selc.wml

그림 15. out.aspx

다음은 변환된 닷넷 모바일 웹 페이지를 서버에서 돌아가도록 구동시킨 후 Internet Explorer(그림 16)와 Pocket PC 에뮬레이터(그림 17) 그리고 PDA(그림 18)를 사용하여 각각 접속한 결과를 나타낸 것이다. 닷넷 모바일 웹 페이지는 요청하는 디바이스에 맞는 output을 생성하여 결과를 보내주기 때문에 또한번의 변환 과정이 필요 없다.

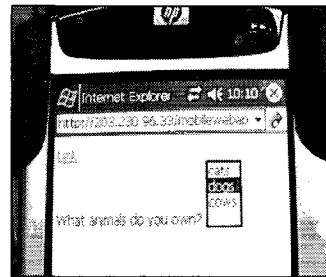


그림 18. PDA에서의 out.aspx 결과화면

## 6. 결론

현재 개발되어 사용되는 변환 솔루션은 직접 개발자의 컴퓨터에 설치하여 사용하는 응용 프로그램 형태로 되어있다. 이러한 솔루션을 사용하기 위해서는 솔루션이 개발된 플랫폼과 이를 사용하는 개발자의 플랫폼이 같아야 한다. 또한 태그의 확장으로 인한 기능의 추가 시 업그레이드 작업이 필요하거나 그에 따른 추가적인 비용과 시간이 든다는 단점을 가지고 있다.

SOAP 기반의 웹 서비스 기술은 플랫폼이나 프로그래밍 언어에 독립적이기 때문에 언제 어디서든 누구나 사용할 수 있다는 장점을 지니고 있다. 또한 웹 서비스를 사용하는 웹 서비스 소비자의 형태나 개발 플랫폼에 상관없이 웹을 통해서 언제든지 사용이 가능하다. 본 연구에서는 웹 서비스가 가지고 있는 장점을 이용하여 WML 페이지를 닷넷의 모바일 웹 페이지로 변환하는 시스템을 웹 서비스로 구현하였다. 또한 웹 서비스를 요청하고 응답 받는 과정을 처리하는 웹 서비스 소비자를 구현함으로써 클라이언트는

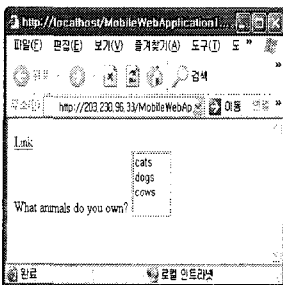


그림 16. Internet Explorer에서의 out.aspx 실행 화면

간단하게 파일을 업로드하고 다운로드받을 수 있는 편리함을 제공하였다.

본 논문에서 구현한 웹 서비스를 통해 WML 페이지를 보내고 그 결과 닷넷의 모바일 웹 페이지가 전송되어 오는 것을 확인하였다. 또한 웹 서비스의 변환 기능을 검증하기 위해 변환 전 WML 페이지와 변환 후 닷넷의 모바일 웹 페이지를 WML 시뮬레이터를 통해 확인해 본 결과 같은 동작을 하는 것을 볼 수 있었다.

### 참 고 문 헌

[1] 한동일, 김미연, 김기수, "무선 인터넷 환경을 지원하기 위한 콘텐츠 변화 미들웨어 시스템 연구," 한국멀티미디어학회, 2001, pp.42-45.

[2] 홍인숙, 김윤중, 김승연, "wmlTo.NET 모바일 웹 페이지 변환 시스템 설계," 한밭대학교 정보통신전문대학원 논문집, 제2-1호, 2003, pp. 177-183.

[3] SOAP, <http://www.w3.org/TR/2002/WD-soap12-part1-20020626>

[4] Andy Wigley, Peter Roxburgh, 문봉재 역, 「모바일 디바이스에서 닷넷 애플리케이션 구축하기」, 정보문화사, 2003.

[5] Specification : Direct Internet Message Encapsulation (DIME) Version 17-June-2002. (<http://www.ibm.com/developerworks/ws-dime/>)

[6] <http://www.mnetsoft.com>

[7] <http://baubau.net/main.htm>

[8] 최지원, 김기천, "무선전용 다중 언어의 번역을 지원하는 변환기의 구현," 정보처리학회, 제9-C권, 2002, pp.293-296.

[9] 홍준호, 송건철, 김정석, 「about WAP」, 영진.COM, 2001.

[10] 정부연, "웹 서비스의 개념과 관련 기업에 미치는 영향," 정보통신정책 제14권 7호, 2002. 4.

[11] 변광준, "웹 서비스 기술과 전망," 한경 Enterprise IT Directions Track E, 2002. 4.

[12] 김성익, 오찬주, 최영미 "웹 서비스 기술과 응용 사례," 멀티미디어 학회, 특집 차세대 주요 IT 기술, 2003. 12, pp.126-138.

[13] 리스 베슈라 외, 「Professional ASP.NET Web Services」, 정보문화사, 2003.

[14] 이한수, 「웹 서비스 실전 프로그래밍」, 한빛미디어, 2002.

[15] Web Services Description Language (WSDL) 1.1 W3C Note Version 15-March-2001, (<http://www.w3c.org/TR/wsdl>)

[16] John R. Levine, Tony Mason, Doug Brown, 「Unix Programming Tools: lex & yacc」, Reilly & Associates, 1992.



#### 홍 인 숙

1999년 한밭대학교 전자계산학과(학사)  
 2004년 한밭대학교 컴퓨터공학과(석사)  
 현재 한밭대학교 컴퓨터공학과 박사과정

관심분야 : 모바일, 웹서비스, 컴파일러, 웹 응용프로그램



#### 김 윤 중

1981년 충남대학교 전자공학과 육과(학사)  
 1983년 충남대학교 전자공학과(석사)  
 1989년 충남대학교 전자공학과(박사)  
 현재 한밭대학교 교수

관심분야 : 컴파일러, 음성인식, 웹 서비스, 모바일 프로그램, 웹 응용 프로그램



#### 김 승 연

1980년 광운대학교 전자공학과(학사)  
 1982년 한양대학교 전자계산학과(석사)  
 1990년 한양대학교 전자공학과(박사)  
 현재 한밭대학교 교수

관심분야 : 운영체제, 보안, 컴퓨터 네트워크