

효율적인 모바일 웹서비스 처리기 설계

박중국, 오수열^o

국립목포대학교 컴퓨터공학전공

e-mail: {jkpark, syoh}@mokpo.ac.kr

Design of Efficient Mobile Web Services System

Jong Kook Park, Su Ryul Oh
Computer Eng., Mokpo National University

요 약

웹 서비스를 구현하기 위해서는 표준화된 WSDL코드를 가지고 SOAP 통신을 해야 한다. 또한 웹 통신을 위해 Tomcat과 AXIS 라이브러리를 필연적으로 사용해야한다. 그러나 Tomcat의 사용은 추가적인 네트워크 사용과 시스템에 프로세스를 상주시켜야 하므로 시스템 자원을 사용하는 문제가 있다. 본 논문에서는 표준화된 웹서비스 처리기를 개선하여 설계함으로써 웹 서비스 구현 시 네트워크 자원과 시스템 자원이 절약하도록 하였다.

1. 서론

1999년도의 후반까지의 통신망의 발달은 정보전송의 고속화에 중점을 두었으나, 2000년대에 들어서면서 고속화와 언제 어디서나 정보에 접근이 가능하도록 하는 정보 접근의 용이성이 큰 화두가 되고 있다. 개인 휴대폰 PDA 등 이동성이 있는 기기들의 보급이 활성화 되면서 정보 접근의 용이성이 요구됨에 따라 휴대폰이나 PDA 등에 모바일 서비스가 가능한 응용 시스템이 필요하게 되었다[1].

웹서비스 기술 연구는 웹 분야 산업의 환경 변화에 대한 적절한 대응 및 전략적 비전을 제시하며 경쟁력 있는 시스템 개발 및 시장 경쟁력 확보를 위하여 현재 그 중요성이 부각되고 있다. 즉, 웹서비스가 단순히 웹 분야에 국한된 응용 분야를 가지고 있지 않고 C-Commerce (Collaborative-Commerce)를 비롯하여, 시스템 통합 및 컴포넌트 소프트웨어 산업 분야, 웹 기술 통합이 예상되는 정보 가전, 내장형 SW 분야, 전자 정부 등에 다양한 형태의 웹서비스 이용이 가능하리라 예상되기 때문에 웹서비스 시스템의 필요성은 증대되고 있다[1,7,9,10].

웹서비스는 “XML(eXtensible Markup Language) 기술 기반으로 인터페이스와 바인딩을 정의, 기술, 검색할 수 있고 URI에 의하여 식별할 수 있는 소프트웨어 응용이며, 인터넷 기반 프로토콜 위에서 XML 기반의 메시지를 이용하여 다른 소프트웨어 응용과 직접적으로 동작할 수 있는 소프트웨어 응용”을 일컫는다[2,3].

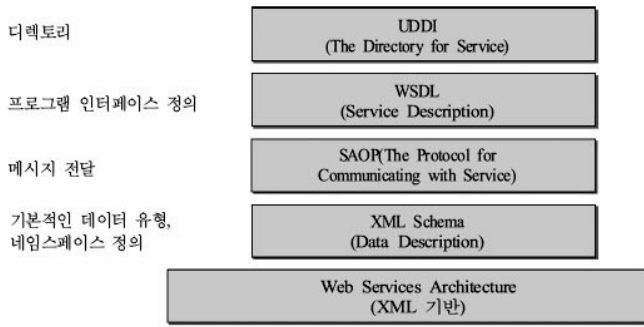
2. 웹서비스 표준 기술

웹서비스의 핵심 기술 표준으로는 SOAP (Simple Object Access Protocol), WSDL (Web Service Description Language), 그리고 UDDI(Universal

Description Discovery and Integration)를 말한다[2,3].

SOAP은 웹 환경 하에서 정보 제공자와 소비자 사이에서 정보를 제공하는 통신 프로토콜로서 정보기술은 XML로 하게 된다. WSDL은 웹서비스의 서비스를 정의하는 언어로서 해당 서비스의 이름, 입력변수, 출력 변수에 대한 정의를 할 수 있다. 이를 통하여 웹서비스 이용자는 서비스 제공자측에서 제공하는 서비스에 대하여 어떻게 호출할 수 있는지 알게된다. 이러한 기술은 과거 분산 객체 환경에서의 CORBA나 DCOM의 IDL(Interface Definition Language)의 기술과 매우 유사하다. 또한, UDDI는 해당 기관에서 제공하는 웹서비스를 사용자에게 알리는 등록의 기능과 서비스 이용자는 어떠한 웹서비스가 제공되는지 검색할 수 있는 기능을 제공한다. 하지만 웹 서비스는 UDDI가 없어도 가능하다[2].

결국, 웹 서비스는 그림1과 같은 구성 요소를 가지며, 다양한 하드웨어와 소프트웨어의 문제를 해결하기 위해 SOAP, UDDI, WSDL과 같은 개방형 표준 기술을 기반으로 서비스 제공자(Service Provider), 서비스 저장소(Service Registry) 및 요청자(Service Requester)를 유기적으로 결합하여 구현된다.



(그림 1) 웹서비스 구성요소

SOAP, UDDI, WSDL의 최근 표준화 동향은 다음과 같다 [2].

• SOAP[5,8]

SOAP(Simple Object Access Protocol)은 마이크로 소프트사와 IBM사에 의해 급속하게 보급되고 있는 표준으로, 클라이언트의 작업 요청과 시스템의 응답을 XML 문자열로 포장하고 전송 프로토콜로는 HTTP를 사용한다. 즉, SOAP은 HTTP와 XML의 결합으로서 분산 환경하에서 정보의 상호 교환을 가능케하는 간단한 프로토콜이다.

• UDDI[6]

UDDI(Universal Discovery Description & Integration)의 목표는 웹 상에서 이용가능한 전자 서비스들에 대하여 분산형 웹 기반 글로벌 레지스트리를 만드는 것이다. UDDI를 개발하게된 직접적인 동기는 서로 연관된 이질적인 시스템, 인터페이스, 프로토콜들을 인식하지 않은 상황에서 중개 소프트웨어와 중개자들을 통해 이용자들이 서비스들을 탐색하고 상호작용할 수 있는 전자 마켓플레이스의 개념을 구현하는데 있다. 오늘날 웹 서비스 레지스트리를 위한 실질적인 표준 규격으로 기능을 하고 있는 UDDI는 Ariba, IBM, 마이크로 소프트사 등에 의해 프로젝트 형태로 추진되고 있다. UDDI 버전 2 레지스트리는 서비스 탐색이 쉽고 비용대 효과면에서 우수하여 웹 서비스를 증진시키는 장점이 있다. e-비즈 레지스트리를 더욱 정교하고 효율적으로 만들기 위해 UDDI 버전 3을 위한 작업들이 현재 진행중이며 여기에서는 고도화된 질의 기능, 캐쉬, 보안, 세계화 등과 같은 기능들이 포함되어 있다. UDDI는 웹 서비스의 필수 사항이 아니고 서비스를 등록하는 역할을 하므로 본 연구에서는 UDDI가 없이도 웹 서비스를 할 수 있도록 설계한다.

• WSDL[2]

WSDL(Web Services Description Language)은 웹 서비스의 IDL(Interface Definition Language) 버전이며 특정 웹 서비스의 방법과 프로토콜, 데이터 포맷들을 더욱 상세하게 정의하는 일종의 스크립트이다. WSDL은 XML 포맷으로 구성되고 HTTP를 통해서 전달될 수 있으며 인터페이스를 정의하는 IDL에 해당한다. 현재는 실질적 (de facto) 산업표준

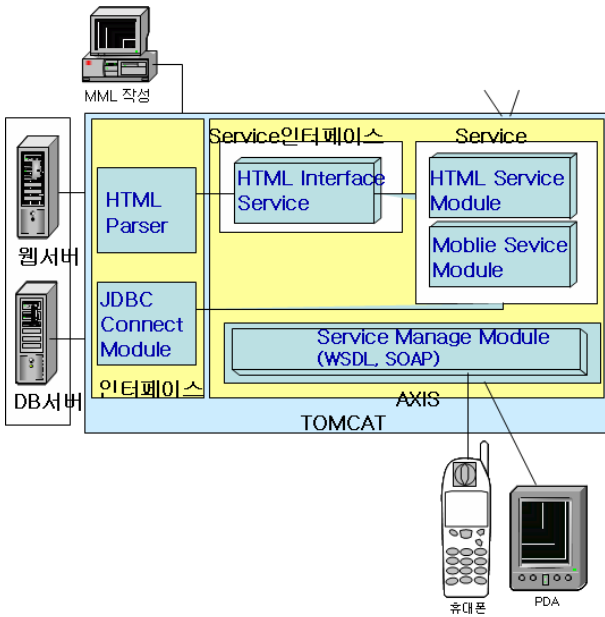
으로 자리잡고 있으며 산업계에서 표준화된 서비스 인터페이스를 정의할 수 있도록 한다. 또한 UDDI 레지스트리와 함께 사용되어 호환되는 서비스들의 동적인 검색과 바인딩을 가능하게 해주며, 이질적인 어플리케이션들에 대해 정규화된 기술도 가능하게 해준다.

3. 웹서비스 시스템 설계

3.1 웹서비스 시스템 아키텍처

본 연구에서 제안하는 시스템에서는 기존의 웹 콘텐츠 및 웹서비스를 제공하기 위한 에디터 기능 및 서비스 전환 기능, 데이터 베이스 서비스와 JDBC를 통한 직접 연결을 사용한 데이터베이스 서비스 기능, 웹 및 모바일 상에서 SOAP 서비스 지원, 모바일 및 웹 콘텐츠 동시지원 등의 기능을 갖는다. 또한 시스템의 확장성 및 유지보수의 용이성, 소프트웨어의 안정성 및 사용되는 기술의 표준성을 준수하기 위하여 Apache 그룹의 AXIS와 Tomcat을 기반으로 한다. AXIS는 Web Service에 필요한 메시지 핸들러 및 메시지 체인 등 웹 서비스 관리의 기능을 제공하며, SOAP을 사용하기 위한 플랫폼(platform)으로서 업계에서 널리 사용되고 있다. Tomcat은 Java Servlet, JSP(Java Server Page)의 공식 참조 구현(Official Reference Implementation)인 오픈소스 서블릿 엔진이다. 본 연구에서 구현한 표준웹 서비스 시스템은 기존에 검증된 Tomcat과 AXIS 서비스 플랫폼 위에서 모바일 서비스를 지원하기 위한 서비스 모델을 제시한다. 설계하는 시스템의 기본 구조 및 신호방식은 Tomcat 서버를 통하여 기본적인 웹 및 모바일 서비스를 지원하며, 모바일 브라우저(WAP 및 Windows Me등) 및 웹 브라우저를 통해 전달되는 Web Service 등의 SOAP 서비스 요청 신호를 AXIS 라이브러리에 전달한다. AXIS 라이브러리로 전달된 SOAP Service 요청 신호는 AXIS에서 관리되고 있는 모바일 또는 웹 서비스 루틴으로 전달되며, 서비스 결과는 Tomcat 서버를 통하여 해당 브라우저로 전달된다.

그림 2는 표준에 입각하여 구현한 시스템 구성도이다. 서버 관리자는 전용툴을 사용하여 기존의 제공되는 웹 콘텐츠를 Parsing하기 위한 MML(Made Markup Language)을 작성한다. MML은 모바일 서비스 요청시 기존의 웹서비스 콘텐츠를 동적으로 Parsing하여 관련 모델 및 클라이언트로 전송하여 서비스 요청 양식 제공이나 서비스 결과 제공시에 사용된다. MML을 통하여 클라이언트로 전달된 양식을 Tomcat을 통하여 전달받은 사용자는 클라이언트측(Client Side)에서 서버측(Sever Side)으로 SOAP 메시지를 요청하며, 요청된 SOAP 메시지는 Tomcat을 통하여 AXIS로 전달되어지며, AXIS는 Service Management 기능을 통하여 해당 서비스로 SOAP Message를 인터페이스 한다. Service를 제공하기 위한 모델들은 AXIS에서 제공하는 WSDL을 모듈을 통하여 인터페이스 되어진다.



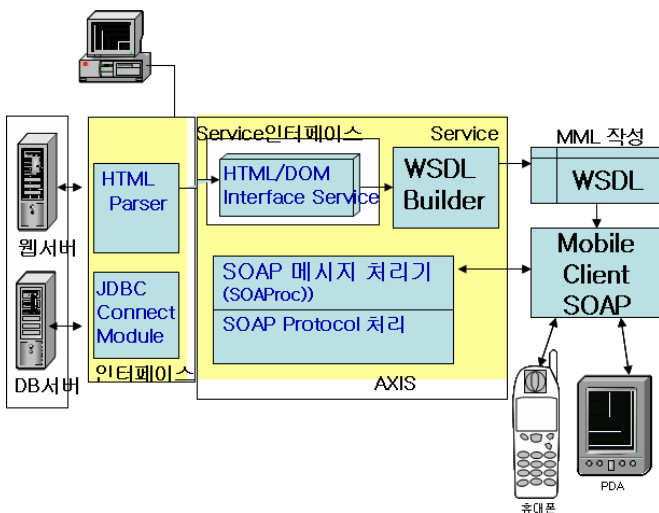
(그림 2) 웹시스템 구조

SOAP 및 분산환경 서비스를 사용함으로써 가벼운 클라이언트 (Thin Client) 구조를 갖게 되며, 이로 인한 서비스 제공의 편의성이 증대된다.

3.2 SOAPProc 설계

클라이언트에서 서버쪽으로 요청(request)이 될 때나 서버에서 클라이언트에게 요청에 대한 응답(reponse)을 할 때 사용하는 것이 xml 형식으로 된 SOAP message이다. 여기서는 SOAP message의 구성과 클라이언트에서 서버쪽으로 요청이 될 때 처리되는 과정과 클라이언트의 요청에 대한 서버의 응답을 할때 어떻게 SOAP message로 변화하는 지에 대해 설명한다.

표준 웹 서비스 시스템과 달리 SOAP 메시지 처리기가 개선된 시스템의 구조는 그림 3과 같다.

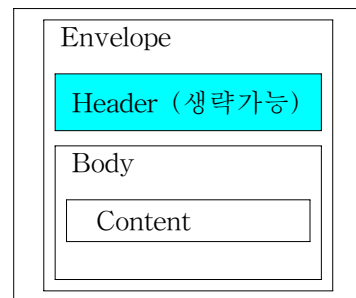


(그림 3) SOAPProc를 활용한 웹 서비스 구조

그림 3은 Tomcat이 없는 것이 그림 2와 가장 대표적인 개선사항이다. Tomcat이 없이 시스템을 구성하기 위해 톰캣이 제공하는 WSDL 생성 부분을 WSDL Builder가 제공한다. WSDL Builder은 WSDL 코드를 생성하는데 이는 HTML Interface Service 부에서 제공하는(서비스 개발자가 HTML 코드에서 모바일 기기에 서비스할 콘텐츠를 추출함) HTML 코드를 가지고 자동으로 WSDL 코드를 생성한다. 또한 서비스 인터페이스 부분에서도 기존의 HTML 인터페이스부에서 HTML코드로 보여지던 것을 HTML 코드를 자동 분석하여 DOM 트리(tree) 형태로 보여 줌으로써 인터페이스 및 서비스 코드 추출 시간과 정확성을 매우 단축시켰다. 또한 그림 2의 서비스 Management Module에서 제공되는 기능은 SOAP 프로토콜 처리기를 구현하여 톰캣이 제공하는 기능을 대신하도록 하였다.

3.3 SOAP 메시지 구성

SOAP message의 구성은 그림 4와 같이 크게 4개로 나눌 수 있다. 첫 번째, Envelope라고 하는 Root Tag로서 SOAP Message가 시작됨을 나타내는 것이 있다. 두 번째, Header라고 하는 것이 있는데 이것은 가외의 정보를 나타내기 위한 것으로 생략이 가능하다. 세 번째, Body가 있는데 이것은 주 내용이 들어가는 Tag이며 다양한 형태의 태그들이 들어간다. 마지막으로 Content가 있다. 여기에는 서비스에 대한 요청이나 응답이 들어간다.



(그림 4) SOAP의 구조

4. 결론

본 연구에서 제안한 시스템은 기존의 웹 콘텐츠 및 웹 서비스를 제공하기 위한 에디터 기능 및 서비스 전환 기능, 데이터 베이스 서비스와 JDBC를 통한 직접 연결을 사용한 DB 서비스 기능, 웹 및 모바일 상에서 SOAP 서비스 지원, 모바일 및 웹 콘텐츠 동시지원 등의 기능을 갖는다. 특히 본 논문의 결과로 Tomcat을 별도로 설치하지 않아도 웹 서비스를 할 수 있게 되어 네트워크 자원을 절약할 수 있고, 항상 상주하여야하는 시스템 프로세스의 수를 줄여 보다 효율적인 시스템 운영을 할 수 있다. 본 논문의 결과물로서 HTML Parser에 DOM 트리를 적용함으로써 웹 서비스 클라이언트 모듈 개발의 효율, 개발의 편리성,

개발 시간 감소 효과를 가져올 수 있을 것이다.

JDBC 인터페이스는 웹과 연동되지 않은 DataBase의 내용을 모바일 기기로 직접 전송할 수 있으므로해서 웹에 올리기 전의 데이터 베이스 내용을 서비스 할 수 있으므로해서 특정인에게 좀더 빠르고 고급스러운 정보 서비스가 가능할 수 있을 것이다.

본 논문의 활용분야는 M(Mobile)- Commerce, 모바일 웹 서비스, Mobile ERP 구축 등 모바일 기기를 활용하는 분야에서 활용이 가능하다.

향후 연구 과제로는 제안된 알고리즘을 해석하고 구현하여 웹 서비스 클라이언트 모듈 개발의 효율, 개발의 편리성, 개발 시간 감소 등 자원의 절약과 성능 향상을 입증하고 모바일 서비스에 보안 프로토콜 등을 추가를 하여 상업적 활용에 부족함이 없도록 할 것이다.

참고문헌

- [1] 박길철 외2, 웹 프로그래밍 학습서, 상조사 출판, 2001년 6월. 355 쪽.
- [2] W3C, <http://www.w3.org/>
- [3] 전정호, 아파치 AXIS 경험담, <http://www.whiterabbitpress.com/lg/authors/tougher.html>
- [4] W3C Technical Report for AXIS, <http://ws.apache.org/axis/>
- [5] W3C Technical Report for SOAP, <http://www.w3.org/TR/soap/>
- [6] 전영호, UDDI와 SOAP ebXML, http://kr.sun.com/developers/tech_docs/pdf_java/java0208.pdf
- [7] Apach Technical Report for AXIS architecture, http://ws.apache.org/axis/_java/architecture-guide.html
<http://jakarta.apache.org/tomcat/>
- [8] Lecture Note, <http://www.jiniproject.com/lecture/xml/soap.html>
- [9] SSL, <http://www.cs.utexas.edu/users/chris/cs378/f98/hdt/project/SSSLv3.html>
- [10] 박길철, “유무선 통합을 위한 인터넷 콘텐츠 변화기의 구현에 관한 연구” 한남대학교 공학 논문집, 제34권, 2004년 2월, pp.419-424.
- [11] Gil-Cheol Park and Seok-Soo Kim, “Designing Linux Cluster Content-Aware Balancer”, II International Conference Science and Technology In 21 century, Nov. 18, 2003 ~ Nov. 21, 2003, pp. 166-167.