
SOAP 기반의 유/무선 웹 서비스 시스템 설계 및 구현

박종훈*

Design and Implementation of Wire/Wireless Web Service System based on SOAP

Jong-Hoon Park

요 약

본 논문에서는 웹 서비스 기술을 이용하여 동적으로 가격비교를 위한 SOAP기반 유/무선 웹 서비스 시스템을 설계 및 구현하였다. 본 웹 서비스 시스템은 크게 상품검색, 상품정렬, 상품주문과 UDDI 검색 브라우저를 지원하는 웹 서비스 클라이언트 모듈과 상품 검색/주문 원격 프로시저와 UDDI 게시 브라우저를 포함하는 서버모듈과 무선 환경에서 상품 검색/주문을 지원하는 모바일 클라이언트 모듈로 나누어지며, SOAP 표준 환경기반으로 상호 정보를 교환하도록 구현하였다. 따라서 사실표준 환경인 SOAP, WSDL, UDDI 표준을 따르는 모든 환경에서 이식가능하며, 플랫폼에 독립적으로 지원할 수 있다.

ABSTRACT

This paper designs and implements wire/wireless web service system based on SOAP for price comparison dynamically using web service technologies in real time. This web service system composes web service client module including functions of product search, product sort, product order and UDDI retrieval browser, and server module including functions of remote procedure call for product search/order and functions of UDDI publish browser, and mobile client module supporting functions for product search/order in wireless environments. This web service system can exchange product information based on SOAP messages and support independence of platform and flexible portability in environment conforming to SOAP, WSDL and UDDI standards.

키워드

Web Service, SOAP, WSDL, UDDI, Standard

1. 서 론

초창기 처음부터 같은 프로토콜 환경에서 시스템을 설계할 경우는 문제되지 않지만, 비즈니스 사업규모가 커지고 기업간의 전자상거래가 활발해짐에 따라 서로 다른 시스템간의 연동이 중요하게 되었다. 즉 적은 비용으로 다른 시스템을 통합시키는 분산 컴포넌트 기반의 컴퓨팅 기술인 웹 서비스 기

술이 필요하다. 여기서 사용하는 프로토콜은 XML 기반의 SOAP(Simple Object Access Protocol)이고, 웹 서비스와 클라이언트간의 프로토콜인 SOAP는 플랫폼 중립적인 프로토콜을 말한다. XML 문서를 인식하고 처리할 수 있는 모든 플랫폼에서 SOAP를 이용할 수 있기 때문에 웹 서비스와 클라이언트의 개별적인 개발이 가능하다. 따라서 웹 서비스는 다양한 확장성과 잠재성을 갖고 있는 기술이기 때

문에 현재 다양한 분산 환경 분야에서 웹 서비스 기술을 연구하고 있다[1]~[4].

본 논문에서는 웹 서비스 적용 기술분야로 가격 비교 시스템을 대상 모델로 선정하여 설계하였다. 현재 가격비교 사이트에서는 관련 쇼핑몰 회사들이 가격비교 사이트에 가입하여 수동으로 상품정보와 가격을 입력하여야 하며, 또한 상품가격을 자동으로 실시간 가격변동을 지원할 수 없는 문제점이 있다. 이러한 문제점을 해결하기 위해 가격비교 사이트에서 쇼핑몰의 상품정보를 실시간으로 받아 가도록 설계하는 것이 필요하다. 또한 웹 서비스 기술 중에 하나인 UDDI 레지스트리에 현재 구축된 다수개의 다양한 서버 정보를 게시하거나 검색하는 UDDI 브라우저 기능이 필요하다. 그리고 시스템의 다양한 환경을 제공하기 위해서는 웹 환경과 모바일 응용 환경 등에서 이용할 수 있어야 한다.

이와 같은 필요성들을 해결하기 위해, 본 논문에서는 웹 서비스 기술을 이용하여 가격비교를 위한 SOAP기반 유/무선 웹 서비스 시스템을 설계 및 구현하였다. 본 논문의 웹 서비스 시스템은 크게 상품검색, 상품정렬, 상품주문과 UDDI 검색 브라우저 기능을 지원하는 웹 서비스 클라이언트 모듈과 상품 검색/주문 원격 프로시저와 UDDI 게시 브라우저 기능을 포함하는 서버모듈과 무선 환경을 지원하는 모바일 클라이언트로 나누어지며, SOAP 표준 환경기반으로 상호 정보를 교환하도록 구현하였다. 따라서 사실표준 환경인 SOAP, WSDL, UDDI 표준을 따르는 모든 환경에서 이식가능하며, 플랫폼에 독립적으로 지원할 수 있다.

II. 관련기술

1. 웹 서비스 개념

웹 서비스에 대한 정의를 표준 기관(W3C)과 각 업체(IBM, SUN, Microsoft)들의 정의를 종합하여 설명하면 웹 서비스란 XML을 기반으로 하는 공개 표준들을 이용해서 인터넷 기반의 분산 컴퓨팅 환경에서 원격 메소드를 호출하여 사용할 수 있는 재사용 가능한 소프트웨어 컴포넌트를 일컫는다. 웹 서비스를 이용하면 웹을 통한 원격 프로시저 호출(RPC : Remote Procedure Call)이 가능해 진다. 웹 서비스는 자기 기술적이며, 퍼블리싱이 가능하고, 웹이나 개방 인터넷 표준을 기반으로 하는 네트워크라면 어떤 곳에서도 설치할 수 있으며, 호출도 가능한 모듈 응용 프로그램이다. 웹서비스 기술과 관련된 표준들은 다음절에서 설명한다[1]~[4].

2. SOAP(Simple Object Access Protocol)

SOAP은 원격 응용프로그램 간의 정보를 교환하는데 필요한 구조를 표준화한 것이다. SOAP 1.1 스펙은 2000년 5월 8일 W3C에 노트(Note)로 공식화되었으며, 2003년 현재 SOAP1.2가 Recommendation 상태이다. SOAP의 구조는 XML 문서형태를 가지고 있고, 전송 프로토콜로서 운반되기 위한 방법에 대해서 언급하고 있다[5].

SOAP은 XML 언어를 이용한 분산 환경에서의 정보교환을 위한 프로토콜이다. SOAP은 XML로 구성되어 있기 때문에 XML을 이해할 수 있는 모든 시스템은 SOAP을 통해서 통신할 수 있다. 즉 이종의 플랫폼 응용프로그램 간에도 정보를 교환할 수 있다. SOAP은 단순한 정보 전달용 XML 메시지로 구성되기도 하고, 다른 분산 컴퓨팅 프로토콜(CORBA, RMI, DCOM)처럼 원격 프로시저 호출(RPC)을 하는 XML 메시지로 구성될 수도 있다 [1],[4],[6].

3. WSDL(Web Service Description Language)

WSDL은 IBM, Microsoft 등의 회사들이 웹 서비스 시스템 기능을 명세화시키는 방법을 표준화하기 위해 만들어졌으며, W3C에서 2001년 3월15일에 WSDL 1.1 Note로 처음 공식화되었으며, 2003년 이후 현재는 버전 2.0(Working Draft)으로 되어 있다.

웹 서비스 제공자는 웹 서비스의 기능, 즉 프로시저의 이름, 인자의 종류, 반환형의 종류, 전송프로토콜의 종류, 웹 서비스의 종점(URL) 등을 WSDL 문서로 작성해서 웹 서비스를 소비자가 언제든지 다운로드 받을 수 있도록 웹서버에 저장한다. 그리고 WSDL 문서의 다운로드 URL을 UDDI 레지스트리에 공개해야 한다. 웹 서비스 소비자인 클라이언트 응용프로그램은 UDDI 레지스트리에서 웹 서비스 명세서인 WSDL 문서를 다운로드 받을 수 있는 URL을 얻은 후 실제로 해당 URL에서 WSDL 문서를 다운로드 받는다. WSDL 문서를 해석한 후 클라이언트 응용프로그램은 웹 서비스 시스템의 사용법을 알게 되고, SOAP 메시지를 생성하여, 전송할 수 있게 된다[1],[3],[7].

4. UDDI(Universal Description, Discovery and Integration)

처음에는 UDDI 레지스트리를 제안한 Microsoft, IBM, Ariba 3개 회사가 UDDI 초안 스펙을 작성하였으나, UDDI 레지스트리에서 정보를 공유하고, 상호검색 할 수 있는 산업표준을 정의하기 위해 UDDI.ORG가 이후에 발족되었다. 이들은

2002년 6월 UDDI 버전 3.0을 발표하면서, 표준화에 대한 작업 일체를 ebXML로 잘 알려진 OASIS에 일임하였으며, 현재 운영되고 있는 스펙은 버전 2 또는 버전3 스펙을 가지고 구현하고 있다[8],[9].

UDDI 레지스트리는 각종정보들을 생성, 저장, 검색할 수 있는 XML 기반의 자료저장장치(S/W+H/W)를 말한다. UDDI 레지스트리의 클라이언트가 UDDI 레지스트리에 접근해서 정보를 저장하고, 찾기 위해서는 SOAP 메시지를 이용한다. UDDI 레지스트리는 지역별, 업종별로 회사정보를 분류하여 특정회사를 쉽게 검색할 수 있도록 구성되며, 마치 전화번호부와 비슷한 방식으로 웹 서비스에 대한 정보들을 화이트페이지, 옐로우 페이지, 그린페이지로 분류되어 있다[10],[11].

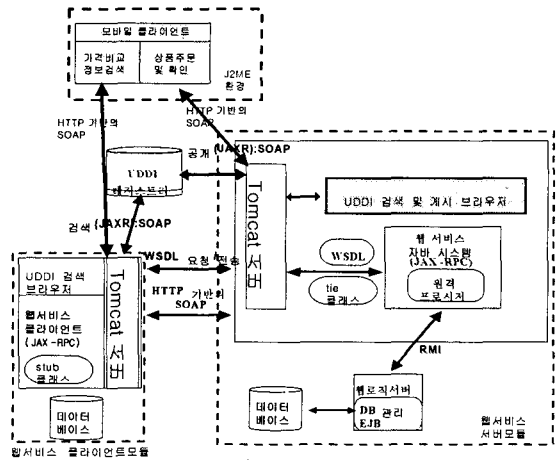


그림 1. 웹 서비스 시스템 설계 구조
Fig. 1 Design structure of web service system.

III. 유/무선 웹 서비스 시스템 설계 및 구현

1. 시스템 개발도구 및 개발 환경

본 시스템에서의 개발환경으로 운영체제는 MS 윈도우 2000, 자바환경은 j2sdk1.4.1_03이며, EJB 환경을 위해 BEA SYSTEM사의 Weblogic 서버를 사용하고, 데이터베이스 도구로는 MySQL을 이용하였으며, MySQL 사용자 인터페이스 도구로는 MySQL-Front를 사용하였다[12],[13],[14]. 모바일 클라이언트 환경으로는 J2ME wireless tool kit 1.0.4와 ksoap을 사용하여 애플리케이션을 개발하였다[15],[16],[17]. 웹 서비스 개발 도구로는 Sun사에서 배포하는 JWSDP(Java Web Services Developer Pack)을 이용하여 웹 서비스를 개발한다[18]. 본 시스템의 서버로는 Apache Group에서 제작한 Tomcat 서버를 사용하였다.

2. 시스템 설계 및 구현

본 논문에서는 다음 그림 1에서 바탕색이 있는 모듈들을 설계 및 구현하였다. 본 논문의 웹 서비스 시스템은 크게 웹 서비스 클라이언트 모듈, 서버모듈과 모바일 클라이언트로 나누어지며, SOAP 표준 환경기반으로 상호 정보를 교환하도록 설계 및 구현하였다. 따라서 표준인 SOAP 환경을 따르는 모든 환경에서 이식가능하며, 플랫폼에 독립적으로 지원할 수 있다.

1) 웹서비스 클라이언트와 서버 설계

웹 서비스 클라이언트 모듈은 사용자로부터 제품 검색과 가격비교를 위한 다양한 쇼핑기능과 주문을 입력 받는 기능과 UDDI 검색 브라우저로 구성하였다. 서버모듈은 웹 서비스 자바시스템, DB

관리 EJB와 UDDI 검색 및 게시 브라우저로 구성하였다. 서버 모듈은 여러 개의 온라인 상점이 존재할 경우 여러개의 모듈이 동시에 존재할 수 있다. 웹 서비스 클라이언트 모듈의 UDDI 검색 브라우저에서는 각각의 상점에 대한 모듈을 인식하기 위해 IBM의 UDDI 레지스트리에서 각 상점의 WSDL의 위치를 인식하여 각각의 WSDL 문서를 다운로드해서 웹 서비스 클라이언트와 스텝 클래스를 설정하도록 함으로서 시스템 확장이 용이하도록 한다.

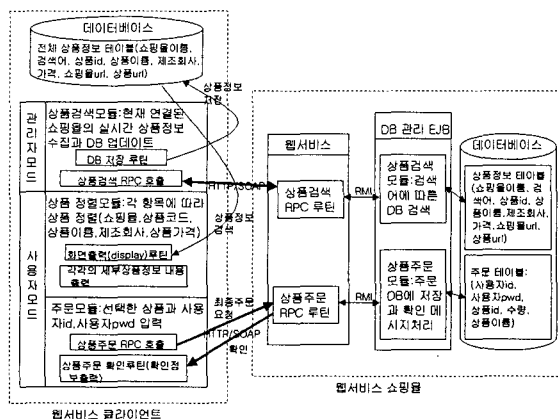


그림 2. 웹서비스 클라이언트와 쇼핑몰의 웹 서비스 처리 흐름도

Fig 2. Process flow diagram of web service client and shopping mall.

그림 2는 웹 서비스 클라이언트, 웹 서비스와 DB관리 EJB와의 처리 흐름과 작업 모듈들과의 관

계를 나타내고 있으며, 웹 서비스 쇼핑몰은 여러 개가 존재할 수 있다. 웹 서비스 클라이언트와 일대 다 관계로 쉽게 확장 가능하도록 웹 서비스를 구성할 수 있다.

2) UDDI 브라우저 설계

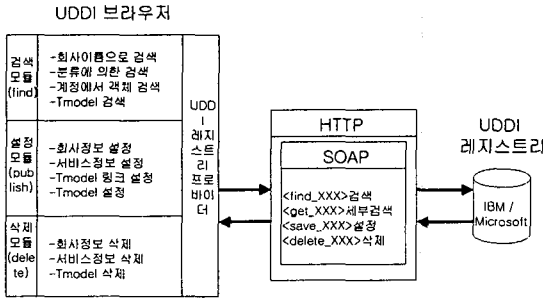


그림 3. UDDI 브라우저의 설계 모듈
Fig 3. Design module of UDDI browser.

UDDI 브라우저는 검색모듈, 게시모듈과 삭제모듈로 구성되어 웹서비스 서버모듈과 웹서비스 클라이언트 모듈에서 함께 동작한다. 클라이언트 모듈, 서버모듈, UDDI 레지스트리와는 SOAP 표준환경기반으로 상호 정보를 교환하도록 설계하였다. UDDI 표준 데이터 구조의 비즈니스 엔티티, 비즈니스 서비스, 바인딩 템플릿, 테크니컬 모델 등에 대한 게시와 검색에 대한 기능을 실행한다.

위 그림 3에서 설계된 UDDI 브라우저 각각의 모듈을 UDDI 스키마 구조와 함께 연결하여 내부구조를 설명하면 다음과 같다.

(1) 검색모듈(Find)

회사이름으로 검색은 회사이름을 입력한 후 먼저 전체적인 회사정보인 <businessEntity>의 하부구조에 대하여 검색한다. 계정에서 객체검색은 사용자가 만든 객체인 <businessEntity>의 하부구조와 <tModel>의 하부구조에 대하여 검색한다. tModel 검색은 단어 검색 또는 전체 tModel 검색을 하면 <tModel>의 하부구조에 대하여 검색한다.

(2) 설정모듈(Publish)

회사정보설정은 <businessEntity>의 하부구조에 대하여 설정할 수 있다. 서비스 정보와 tModel 링크 설정은 회사이름으로 검색한 후 <businessService>의 하부구조와 tModel 이름과 tModel key 값을 설정한다. tModel 설정은 <tModel>의 하부구조에 대하여 설정한다.

(3) 삭제모듈(Delete)

회사 정보 삭제는 회사이름을 검색하여 <businessEntity>의 전체 정보를 삭제한다. 서비스 정보 삭제는 회사이름을 검색하여 <businessEntity>의 하부구조인 여러 개의 <businessService>들을 검색하여 선택 삭제할 수 있으며, 또한 <businessServices>의 전체 삭제도 가능하다. tModel 삭제는 tModel의 키값을 이용하여 관련 <tModel> 전체를 삭제할 수 있다.

3) 모바일 클라이언트 설계

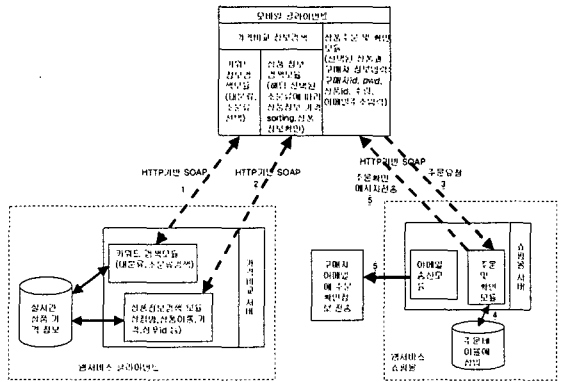


그림 4. 모바일 클라이언트의 웹 서비스 처리 흐름도
Fig 4. Process flow diagram of web service for mobile client.

모바일 클라이언트는 가격비교 정보검색과 상품 주문 및 확인 모듈로 구성된다. 가격비교 사이트인 웹서비스 클라이언트에서는 모바일 처리에 대한 웹서비스 모듈로는 키워드 검색모듈과 상품정보 검색모듈로 구성된다. 모바일 처리 요청을 위한 웹서비스 쇼핑몰에서는 주문 및 확인 모듈과 이메일 송신모듈로 구성된다. 다음 그림 4에서는 모바일 클라이언트에서의 처리 흐름도를 순서적으로 보여 주고 있으며, 점선으로된 화살표는 SOAP 기반 프로토콜을 사용한다.

- 1) 먼저 모바일 클라이언트에서는 상품의 대분류, 소분류를 검색하기 위해 키워드 검색어를 웹서비스 클라이언트에게 요청한다. 웹서비스 클라이언트에서는 요청한 키워드를 데이터베이스에서 검색하여 모바일 클라이언트에게 전송한다. 전송 받은 모바일 클라이언트는 수신된 키워드를 이용하여 메뉴를 구성한다.
- 2) 모바일 클라이언트에서 선택된 키워드에 따라 상품정보 검색을 웹서비스 클라이언트에게 요청한다. 웹서비스 클라이언트는 키워드에 따라

데이터베이스에서 가격으로 소팅하여 순서적으로 모바일 클라이언트에게 전송한다. 모바일 클라이언트에서는 정보를 수신하여 화면에 출력한다.

- 3) 모바일 클라이언트에서는 출력된 상품정보 중에서 구매자가 구매할 상품을 선택한 후, 주문절차에 따라 구매자 정보와 수량, 이메일 정보를 입력한 후 쇼핑몰 서버에 주문 요청한다.
- 4) 쇼핑몰 서버에서는 주문 및 확인 모듈에서 주문 요청을 수신하여 처리한다. 주문 요청된 정보를 데이터베이스의 주문 테이블에 삽입한다.
- 5) 성공적으로 주문이 되었을 경우 이메일 전송모듈을 이용하여 확인 메시지를 전송하고, 모바일 클라이언트에도 동시에 확인 메시지를 전송한다.

3. 실행 결과

1) UDDI 브라우저 실행 결과

다음 그림 5는 UDDI 브라우저 정보 설정 기능으로 회사정보설정, 서비스 정보와 tmodel 링크설정과 tmodel 설정기능이 있으며, 이 중에서 회사정보를 설정하는 입력 화면을 보여주고 있다. 회사정보를 입력한 후 publish 버튼을 클릭하면 IBM의 UDDI 테스트 레지스트리에 정보가 저장된다.

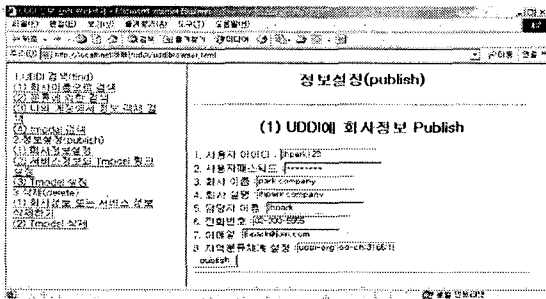


그림 5. 비즈니스 정보 게시를 위한 입력화면

Fig 5. Input screen for publication of business infomation.

다음 그림 6은 IBM의 UDDI 테스트 레지스트리에 회사정보가 저장된 화면을 XML 형식으로 보여주고 있다. UDDI 표준 명세에 따라 비즈니스 엔터티 요소의 회사명이 저장되어 있고 다음 contact 요소에 담당자, 이메일 등이 입력되어 있다.

2) 웹 서비스 클라이언트 실행 결과

웹 서비스 가격비교 클라이언트에서 UDDI 레지스트리를 검색하여 웹 서비스 정보를 얻는다. 이

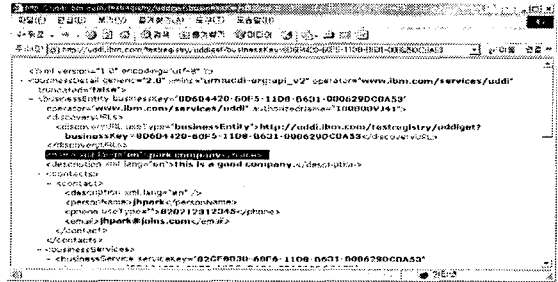


그림 6. UDDI 레지스트리에 저장된 비즈니스정보
Fig 6. Business information stored in UDDI registry.

정보를 이용하여 웹 서비스 쇼핑몰과 바인딩 하여 상품정보 검색을 실행하며, 또한 상품정렬과 상품주문에 대한 순서적인 실행 결과를 보여준다.

- (1) 다음 그림 7은 실제로 요청 SOAP 메시지와 응답 SOAP 메시지를 확인하기 위해 아파치 그룹에서 지원하는 TCPMonitor를 이용하여 SOAP 메시지를 확인하였다[19]. 왼쪽 영역은 클라이언트에서 요청 SOAP 메시지를 보내는 화면이고, 오른쪽 영역은 웹 서비스 쇼핑몰 서버에서 결과를 응답하는 SOAP 메시지이다.

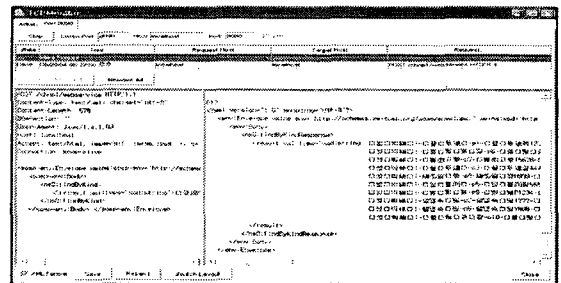


그림 7. TCPMonitor를 이용한 상품검색 요청과 응답 결과

Fig 7. Result of product search request and response using TCPMonitor.

- (2) 다음 그림 8은 검색어를 “모니터”로 하여 데이터베이스를 검색한 후 상품정렬하기에서는 쇼핑몰이름, 상품코드, 상품이름, 제조회사, 상품가격 정렬하기 중에서 상품가격을 기준으로 정렬하였을 경우의 화면을 보여준다. 또한 장바구니 기능을 추가하여 여러 개의 상품을 동시에 주문할 수 있도록 하였다.

- (3) 다음 그림 9는 주문하기 버튼을 클릭한 후 올바르게 상품이 신청된 경우 구매자에게 신청한 내용을 보여주는 화면이다. 관련 쇼핑몰 서버

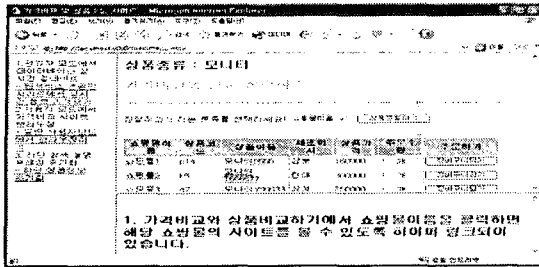


그림 8. 상품가격으로 정렬한 화면
Fig 8. Screen sorted by price in user mode.

모듈에서 주문 승인에 대한 정보를 클라이언트로 전송하면, 웹 서비스 클라이언트 모듈에서 확인 메시지를 출력되며, 또한 구매자의 이메일로도 확인 메시지가 전송된다.

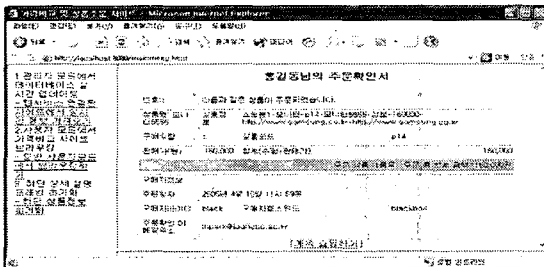


그림 9. 구매자에게 신청한 내용을 보여주는 확인 화면
Fig 9. Confirmation screen showing order product.

3) 모바일 클라이언트 실행 결과



(1) (2)
그림 10. 분류항목에 따른 상품정보 출력
Fig 10 Output of product information by classification items.

(1) 그림 10의 (1)은 대분류, 소분류 키워드에 따라 선택 메뉴를 보여주고 있으며, 현재 'mouse'를 키워드로 선택한다. 이것은 SOAP 메시지를 통하여 가격비교 사이트인 웹서비스 클라이언트에게 요청한다. 그림(2)은 웹서비스 클라이언트로부터 'mouse' 키워드를 갖는 상품정보를 수신하여 상품정보를 가격순서로 출력한 화면이다.

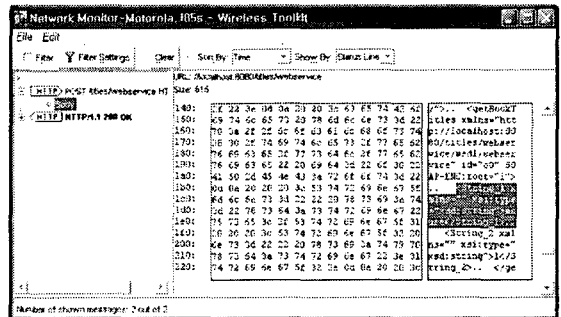


그림 11. 무선 툴킷을 이용한 SOAP 메시지 확인
Fig 11. Confirmation of SOAP message using wireless toolkit.

(2) 위 그림 11은 무선 툴킷을 이용하여 모바일 클라이언트로부터 "mouse"라는 키워드를 웹서비스 클라이언트에게 전송하는 SOAP 메시지를 확인한 화면이다. 파란색으로 반전된 영역은 "mouse" 키워드가 전송되는 부분을 보여주고 있다.



(1) (2)
그림 12. 주문자 정보 입력과 주문확인
Fig 12. Input of orderer information and confirmation of order.

(3) 그림 12의 (1)는 해당 상품 주문 시에 주문자의 정보를 입력하는 화면이다. 입력 정보로는 사용자 아이디, 패스워드, 상품 아이디, 수량, 주문날짜, 이메일 정보를 입력한 후 해당 쇼핑몰 서버로 SOAP 메시지를 전송한다. 쇼핑몰 서버는 주문요청을 수신하여 주문 정보를 데이터베이스의 주문테이블에 입력한 후 주문확인 메시지를 모바일 클라이언트와 주문자 이메일로 메시지를 동시에 전송한다. 위 그림(2)은 모바일 클라이언트에 수신된 주문확인 메시지를 보여주는 화면이다. 출력정보로는 쇼핑몰 이름, 사용자 아이디, 사용자 패스워드, 상품정보, 주문날짜 등이 출력된다.

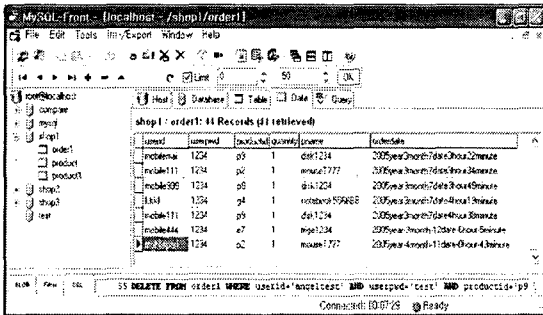


그림 13. 데이터베이스에 주문 정보가 저장된 화면
Fig 13. Screen stored with order information in database.

(4) 위 그림 13는 모바일 구매자가 상품을 선택하여 해당 쇼핑몰1의 MySQL 데이터베이스에 주문을 신청한 화면을 보여주고 있다.

4. 고찰

가격비교 사이트에서 쇼핑몰의 상품정보와 가격을 실시간으로 업데이트 가능하며, 또한 인터넷 브라우저를 통한 가격비교 사이트와 무선 환경의 모바일 클라이언트에서 각항목별로 정렬이 가능하도록 다양한 환경을 지원하였다. 상품주문도 실시간으로 직접 주문가능하며, 그에 대한 확인메시지와 이메일 전송도 지원하였다. 그러나 상호 호환적인 웹 서비스 환경을 지원하기 위해, 쇼핑몰의 데이터베이스 내에 상품주문과 상품정보 스키마가 표준적으로 통일이 되어야 하며, 검색어에 대한 키워드도 일치되어야 웹 서비스 가격비교 클라이언트와 모바일 클라이언트에서 일관된 방법으로 웹 서비스가 가능하다.

본 시스템의 설계 기본 방향은 다음과 같다. 가격비교 사이트에서 쇼핑몰의 상품정보를 실시간으로 수신하거나, 주문을 할 수 있도록 지원하고, UDDI 레지스트리에 현재 구축된 다수개의 다양한 서버 정보를 게시하거나 검색하는 UDDI 브라우저 기능을 지원한다. 마지막으로 시스템 환경의 유연성을 제공하기 위해서 웹 환경과 모바일 응용 환경을 이용할 수 있어야 한다.

따라서 이러한 설계 기본 방향을 만족시키기 위해 본 논문에서는 웹 서비스 기술을 이용하여 가격비교를 위한 SOAP기반 유/무선 웹 서비스 시스템을 설계 및 구현하였다. 본 시스템은 사실표준 환경인 SOAP, WSDL, UDDI 표준을 따르는 모든 환경에서 이식가능하며, 플랫폼에 독립적으로 지원할 수 있다.

향후 추가 확장해야 할 부분은 SOAP 메시지에서의 보안 분야와 WSDL을 이용한 자동화된 바인딩기술 분야가 필요하다. 본 시스템의 적용 가능한 응용분야로는 가격비교를 이용한 전자상거래 기술, 그리드 컴퓨팅, 본사와 지사간의 정보관리 시스템, 모바일 응용 등으로 다양하게 적용할 수 있을 것으로 생각된다.

참고문헌

- [1] Patric Cauldwell and et al., Professional XML Web services, wrox, 2002.
- [2] Adam Freeman and Allen Jones, MicroSoft. Net XML Web Services Step by Step, MicroSoft Press, 2003.
- [3] Harvey M. Deitel and et al., Java Web Services for Experienced Programmers, Prentice Hall, 2003.
- [4] Simeon Simeonov and et al., Building Web Services with Java, SAMS, 2002.
- [5] W3C, SOAP Version 1.2(part0,part1,part2) (Recommendation), <http://www.w3.org/2002/ws/>, 2003.
- [6] W3C recommendation, Extensible Markup Language(XML)1.0(third edition), <http://www.w3.org/TR/REC-xml/>, 2004.
- [7] W3C, WSDL Version 2.0(Working Draft), <http://www.w3.org/2002/ws/>, 2003.
- [8] IBM UDDI Business Registry V3 beta, <https://uddi.ibm.com/beta/registry.html>
- [9] MicroSoft UDDI Business Registry Node, <https://uddi.beta.microsoft.com/>

IV. 결 론

- [10] UDDI standard spec, <http://www.uddi.org/specification.html>
- [11] UDDI Version 3.0 UDDI Spec Technical Committee Specification, <http://uddi.org/pubs/uddi-v3.00-published-20020719.htm>, 2002.
- [12] Bea systems, Bea weblog platform 8.1, http://www.bea.com/framework.jsp?CNT=homepage_main.jsp&FP=/content.
- [13] MySQL, "MySQL reference manual", <http://dev.mysql.com/doc/>
- [14] Star Tools, "Welcome to MySQL-Front", <http://www.mysqlfront.de/>
- [15] ksoap Documentation, <http://ksoap.objectweb.org/software/documentation/index.html>
- [16] J2ME wireless toolkit Version 1.0.4._02, Sun Microsystems, Sep. 2004.
- [17] Burdick and Robert, Wireless Application Programming With J2Me, MacGraw-Hill, 2003.
- [18] Java Web Services Developer Pack, <http://java.sun.com/webservices/downloads/webservicespack.html>, 2003.
- [19] Apache Group, Using the Axis TCPMonitor tool, <http://ws.apache.org/axis/java/user-guide.html>.

저자소개

박종훈(Jong-Hoon Park)



1995년 광운대학교 전자계산기공학과(박사)
1995년~1998년 한국전산원 선임연구원
1999년~현재 중부대학교 컴퓨터멀티미디어학과 교수

※ 관심분야 : XML 웹서비스, XML 검색