

---

# SOAP을 기반으로 한 마켓플레이스 에이전트 설계와 구현

최유순\* · 소경영\*\* · 신현철\*\*\* · 한성국\* · 박종구\*

Design and Implementation of SOAP-based Marketplace Agent

Yue-Soon Choi\* · Kyung-Young So\*\* · Hyun-Cheul Shin\*\*\* · Sung-kook Han\* · Jong-Goo Park\*

## 요 약

사용자에게 다양한 웹 서비스를 쉽고, 간편하게 제공하기 위한 연구가 계속되고 있다. 사용자가 요구하는 필요한 정보를 제공하기 위하여 웹용 마크업 언어인 XML이 개발되었고, 사용자들이 서비스를 효율적으로 이용할 수 있도록 UDDI 레지스트리에 WSDL을 사용하여 서비스를 정의하여 제공하고 있다. SOAP 프로토콜은 전송 매커니즘으로 HTTP를 사용하기 때문에 방화벽 양단에서 사용할 수 있다. 본 논문에서는 사용자의 요구에 정확하고, 신속한 서비스를 제공하기 위해서 마켓플레이스에 에이전트를 설치하고 SOAP을 통하여 서비스 제공자와 UDDI, 그리고 에이전트 사이의 통신을 담당하게 하였다.

## ABSTRACT

It is being continued that studies is to supply users with various web service easily and simply. XML, extensible mark up language for web has been developed, and WSDL is provided for web service to UDDI registry in order to more efficient usage. SOAP protocol can be used for firewall endpoint because it uses HTTP as transfer mechanism. In this paper, we are studied dealing with installing agent in marketplace and communicating among service supplier and UDDI. Then agent through SOAP to provide exact and prompt service to demands of users.

## 키워드

xml, agent, wsdl, uddi

---

\*원광대학교 컴퓨터공학과

\*\*\*천안외국어대학 사무정보학과

\*\*익산대학 컴퓨터학과

접수일자 : 2002. 4. 13

## 1. 서론

오늘날 시장을 주도하는 동적인 힘의 근원은 고객으로부터 시작되는 변화에 대한 욕구로부터 출발한다. 오늘날의 소비자들은 고품질의 제품과 서비스를 선택적으로 조합하여 자신만의 경쟁 전략을 구현할 수 있는 맞춤 능력을 요구하고 있다. 선도적인 기업들은 고객의 주문을 확보하기 위한 새로운 아이디어와 프로세스를 끊임없이 탐색하고, 기존 고객의 경계를 넘어서서 완전히 새로운 시장을 창출할 수 있는 시장 혁신 능력과 필요한 자원의 확보에도 역량을 집중하고 있다. 정보 통신 기술은 공급 채널을 따라서 상호 작용하는 업무 기능들을 연결하여, 다수의 독립적인 회사들을 하나의 경쟁력있는 공급 체인 시스템으로 변화시키고 있다. 무엇보다도 인터넷을 바탕으로 한 전자상거래는 시장의 구조를 급속히 변화시키고 있다.

인터넷의 사용 증가와 분산 처리 기술의 발달로 생활의 상거래 환경을 전자적으로 구현하여 기업, 개인, 정부 등 경제 주체들은 기대와 관심을 한층 고조시키고 있다. 그러나 인터넷 상에는 다양한 종류와 규모의 쇼핑몰이 만들어지고 있고 그 속에는 더욱 넓은 범위의 상품들이 각각 다른 조건과 가격으로 거래되고 있다. 이러한 상황 속에서 인터넷이나 전자상거래에 익숙해 있지 않은 많은 사용자들은 물론이고 보다 좋은 조건으로 상품을 구매하고자 하는 사용자들은 자신이 필요로 하는 상품 혹은 그것을 취급하는 상점 및 구매 조건을 찾기 위해 많은 시간과 노력이 필요로 된다. 이러한 사용자의 어려움을 도와주기 위해 대리인이라는 의미의 지능형 소프트웨어인 에이전트의 역할이 커지고 있다<sup>[1]</sup>. 에이전트를 사용하면 사용자는 스스로 여러 인터넷 상점을 오가며 상품 정보를 수집, 비교하는 등의 번거로운 작업을 하지 않고도 자신의 개인 취향이나 조건에 맞는 맞춤정보를 얻을 수 있다<sup>[2][3]</sup>.

에이전트는 이질적인 망(heterogeneous network)에서 자신의 제어로 호스트들을 옮겨다니며 다른 호스트의 에이전트 서버나 에이전트와 상호 작용하거나 자원을 이용하면서 사용자의 작업을 수행하는 프로그램이다. 이동성과 자율성을 특징으로 하는 에이전트는 네트워크 부하를 줄일 수 있고, 불안정한 통신 환경에서 클라이언트와 서버 사이의 지속적인 연결을 유지할 필요가 없으며, 또한 클라이언트의 요구가 다양하고 수

시로 변하는 환경에서도 장점을 갖는다<sup>[4][5]</sup>. XML은 사용자의 다양한 요구를 지원할 수 있는 가장 적절한 언어이다. 다수의 상품 카탈로그를 받아서 비교표를 만들거나, 문서를 전달받아서 자동으로 정보시스템에 입력하고자 하는 경우와 같은 전자상거래를 지원하기 위해서는, 전달된 데이터나 전자문서를 컴퓨터가 즉시 이해하는 것은 중요하다.

XML은 이러한 문제를 해결할 수 있는 HTML과는 다른 마크업을 사용하여 사용자 소프트웨어에 전달될 수 있도록 하였다. SOAP은 플랫폼과 무관하게 서비스, 객체, 그리고 서버를 접근하기 위한 XML과 HTML의 사용 방법에 대하여 정의한 표준이다. 달리 말하면, SOAP은 이질적인 소프트웨어 컴포넌트간의 접착제 역할을 하는 프로토콜이다. 인터넷에서 구매를 위해 구매함수를 함수가 제공된 경우, 이 함수의 호출을 SOAP으로 표현한다. SOAP은 POST 방식으로 URL에 전달되지만, SOAP에서는 객체를 어떻게 실행하는지에 대해서는 언급을 하지 않기 때문에, 객체를 실행하고 함수를 호출하는 방법은 URL 내부의 코디에서 결정해준다. 함수 호출에 대한 응답은 Message Type 헤더의 값으로 확인할 수 있다. 응답 내용은 함수 호출 관련 엘리먼트 대신에 응답 엘리먼트에 기술된다. SOAP으로 호출과 응답이 이루어지는 전자상거래는 사용자의 요구에 적합한 웹 서비스 지원을 위해 UDDI 레지스트리를 검색하도록 한다. UDDI 레지스트리에는 다양한 서버에서 지원되는 서비스를 등록해 놓은 저장소로서 사용자의 작업을 정확하게 수행하도록 하고 있다. UDDI 레지스트리에 저장된 정보는 WSDL로 이용이 가능하다<sup>[6]</sup>.

본 논문에서는 사용자의 요구에 적합한 서비스를 지원하기 위한 방안으로 마켓플레이스에 에이전트를 두어 구현하였다. 사용자는 인터넷으로 마켓플레이스에 연결되고, 마켓플레이스는 각 제공업체들과 인터넷상의 SOAP을 통하여 정보를 전송하도록 하였다. 상품을 구매하기 위하여 구매자는 정보를 요구하게 되고, 마켓플레이스와 공급자는 좀 더 빠르고 쉽게 구매자에게 응답을 해주게 된다.

본 논문의 구성은 다음과 같다. 2장에서는 관련연구로서 기존의 에이전트 마켓플레이스를 연구하고, 3장에서는 본 논문에서 제안한 SOAP 기반의 에이전트의 구조 및 기능에 대하여 상세 설명하였고, 4장에서는

구현 방법 및 내용에 대하여 기술하였다. 5장에서는 구현 결과와 향후 연구방향을 제시하였다.

## II. 관련연구

인터넷을 기반으로 한 전자상거래는 사용자가 요구하는 정보를 정확하고, 신속하게 제공하여야 한다. 전자상거래는 단순히 물건만을 매매하는 공간이 아닌 서비스를 정의하고, 언제나 서비스를 지원해줄 수 있어야 한다.

### 2.1 직접 판매 모델(Direct Sales Model)

직접 판매 모델은 도매 기업이나 소매 기업을 중심으로 중간 유통 단계를 과감히 없애고, 고객과 직접 접촉하는 모델이다. 델컴퓨터가사 최초로 적용한 이 모델은 판매 비용을 절감하고, 고객과의 직접 접촉을 통해서 새로운 정보를 제공하거나 고객이 원하는 상품을 맞춤 형식으로 제공할 수 있다. 즉, 고객이 직접 생산 프로세스에 참여하는 효과를 얻게 되며, 주문 생산 방식에 의해서 재고 비용을 감소시키고, 현금 흐름을 개선시킬 수 있다. 이외에도 이 모델의 장점은 고객의 요구를 직접 파악할 수 있어서 제품 개발에 활용할 수 있으며, 고객의 취향이나 수요를 세밀하게 파악할 수 있기 때문에 정확한 생산 대응이 가능해진다. 그러나, 직접 판매 모델은 제조 회사 중심의 거래 체계로서, 구매자 입장에서 여러 가지 불편한 점이 있다. 구매하고자 하는 제품을 생산하는 여러 회사로부터 카탈로그를 비롯한 제품 정보를 입수하여, 종합적으로 비교하고자 하는 경우, 정보가 불충분하기 때문이다.

### 2.2 추천 시스템

추천 시스템은 전자상거래에서 핵심 기술로 각광을 받고 있다<sup>[7]</sup>. 이는 인터넷 상점의 폭발적인 증가로 인해 상점마다 다른 상점과 다른 차별화 된 서비스 기술을 요구하고 있기 때문이다. 추천 시스템은 개개인의 성향에 맞게 정보를 가공하여 사용자에게 전해준다. 여기에는 추천의 기준에 따라 비 개인화 추천방식, 선호도 기반 추천방식, 항목별 추천방식, 사람 대 사람 추천방식이 있다<sup>[8]</sup>. 비 개인화 추천 방식은 추천의 기준을 사용자들이 말하는 평균에 두는 방식으로, 개인

사용자와는 무관하게 많이 팔리는 상품을 추천하는 방식이다. 선호도 기반 추천방식은 물품의 속성과 사용자의 선호도를 비교하여 추천하는 방식이며, 항목별 추천방식은 상품을 조그마한 집합으로 구분하여 만약 사용자가 집합에 포함된 상품을 구입하면 같은 집합에 있는 상품을 추천하는 방식이다. 사람 대 사람 추천방식은 사용자와 다른 사용자간의 관계를 이용하여 상품을 추천하는 방식이다.

### 2.3 비교 쇼핑 시스템

비교 쇼핑 시스템은 지능형 시스템이 전자상거래에 응용된 분야로써 널리 활용되고 있다. 사용자가 특정 상품을 찾고자 할 때, 에이전트가 사용자를 대신하여 인터넷에 존재하는 상점을 검색하여 상점별로 해당 상품을 비교하는 시스템을 말한다. 이 기능으로 인해 사용자는 인터넷을 직접 검색하는 수고와 시간을 줄일 수 있는 장점을 얻을 수 있다. 개발된 시스템으로는 BargainFinder와 Jango 같은 시스템이 있다. 그러나 BargainFinder는 미리 정해진 인터넷 상점만 비교할 수 있다는 단점과 함께 가격 경쟁력이 약한 사이트들이 서비스 제공을 거부함으로써 시스템의 기능을 상실하고 있다. 한편 Jango는 상점을 등록할 필요 없이 사용자의 요구에 따라 상점을 분석할 수 있는 기능을 추가하여 이러한 문제를 해결하려고 하였으나, 상점의 분석 성공률이 50% 이하로 실질적으로 문제를 해결하지는 못하였다.

### 2.4 계획 에이전트 마켓플레이스

계획 에이전트는 인터넷 마켓플레이스의 여러 기능들 중에서 데이터와 소프트웨어 서비스들을 결합하고 실행하기 위해 수행순서를 만들어서 사용자가 사용하기 쉽도록 지원해 주는 역할을 한다. 구매자에게 서비스들을 즉시 실행할 수 있는 프로그램을 제공하므로 검색기와는 그 기능이 다르다. 검색기는 어떤 요구사항을 만족하는 서비스의 이름과 간단한 설명만 제공한다. 따라서 계획 에이전트는 다중데이터베이스 시스템에서의 질의 최적화기와 비슷한 역할을 수행한다. 실행 계획을 세우기 위해서 메타데이터가 이용되는데 이 정보는 서비스 공급자가 서비스를 등록할 때 만들어진다. 그러나 계획 에이전트 마켓플레이스는 이질적인 분산 환경을 극복하기 위해서 산업 표준으로 표현되어야

하는 숙제를 안고 있다.

### III. 마켓플레이스 에이전트 설계

#### 3.1 SOAP

SOAP은 플랫폼과 무관하게 서비스, 객체, 그리고 서버를 접근하기 위한 XML과 HTML의 사용 방법에 대하여 정의하고 있다. 달리 말하면, SOAP은 이질적인 소프트웨어 컴포넌트간의 접착제 역할을 하는 프로토콜이다<sup>[9]</sup>. HTTP와 XML의 동시 사용을 원하는 경우, 표준적인 방법으로 경쟁적인 기술들을 연계하기 위한 메커니즘을 SOAP은 제공한다. 이것은 HTTP와 XML을 단일 솔루션으로 결합하여 전혀 새로운 수준의 상호 운용성을 제공한다. 예를 들면, 윈도우 환경에서 비주얼 베이직으로 작성된 단말 프로그램으로부터 UNIX 환경에서 작동하는 CORBA 서비스를 손쉽게 작동시킬 수 있으며, 자바 스크립트로 작성된 단말 프로그램으로부터 대형 컴퓨터에서 작동 중인 서비스를 작동시킬 수도 있다.

SOAP은 XML-RPC로부터 시작되었다. SOAP 역시 HTTP와 XML 문서를 통한, 일련화 프로시저 호출을 사용한다.

SOAP은 UserLand와 DevelopMentor, Microsoft의 협력에 의해 만들어졌다. 최초의 공개 릴리즈는 XML-RPC를 기본으로 하고 네임스페이스와 긴 엘리먼트 이름을 사용할 수 있도록 하였다. 그러나 그 이후에 SOAP은 W3C working group으로 넘겨졌다. W3C working group은 SOAP에 여러 잡다한 것을 추가시켰고, 현재는 XML Schemas, enumerations, struct와 array의 혼합, 그리고 사용자 정의 데이터형 등을 지원한다. 그와 동시에 플랫폼에 따라 몇 가지 형태의 SOAP이 나타났다

SOAP은 표준 분산환경인 CORBA의 복잡한 자체 내의 구조와 사용법의 어려움을 해결하기 위한 방안으로 개발되었다. CORBA가 아무리 복잡한 구조를 가지고 있더라도 기본적인 개념은 서버에서 구현해 둔 서비스를 호출해서 처리된 결과값만 받아서 이용하는 것이기 때문에 SOAP에서의 호출 방법 또한 XML을 기반으로 객체의 개념을 도입하여 호출/응답 서비스가 상호 독립적이면서 상호 운용성이 뛰어나다는 장점을

가지고 있다.

SOAP에서는 HTTP 혹은 SMTP 상에서 단순히 서비스 요청/응답에 대하여 표준 XML 문서 구조를 정의하여 서비스 제공업자와 클라이언트 간에 상호 교환하도록 구성되어 있다. 서비스를 요청하는 문서에는 사용할 서비스의 정보와 입력값을 채워서 보내주면 서비스 제공업체는 이를 분석하여 해당 서비스를 수행한 후 그 결과값을 응답 문서에 넣어서 돌려주는 방식이다.

SOAP은 SOAP envelope와 body를 가지며 선택적으로 header를 가질 수 있다. 그림 1.에 SOAP의 구조

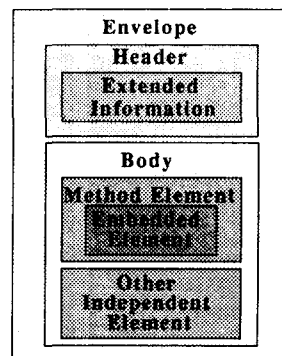


그림 1. SOAP의 기본 구조

를 나타내었다. SOAP은 무상태성이므로 어플리케이션이 저장해야할 것을 다룬다면 문제가 발생한다. 이때 세션ID를 이용해서 RPC 호출이 끝날때까지의 프로그램 상태를 기억하도록 한다. 그림 2.는 클라이언트와 서버간의 SOAP을 통한 메시지 교환을 나타내었다.

#### 3.2 UDDI(Universal Description, Discovery and Integration)

UDDI는 마이크로소프트, IBM 그리고 Ariba에 의해서 제안된 프로젝트의 결과로서 기업들에게 어느 누구나 사용할 수 있는 Universal Business Registry에 그들이 제공하는 웹 서비스에 대한 정보를 알리는 것을 가능하게 하고 있다. 만일 UDDI가 방대한 지지를 얻는다면 선두 기업들은 이것이 모든 규모의 회사들이 전세계적인 기회를 디지털 혁명에 의해서 제공받게 될 B2B 온라인 상거래의 성장을 가속화 시키는 것에 앞장설 것이라고 말하고 있다<sup>[10][11]</sup>.

UDDI Business Registry를 사용하면 협력업체에

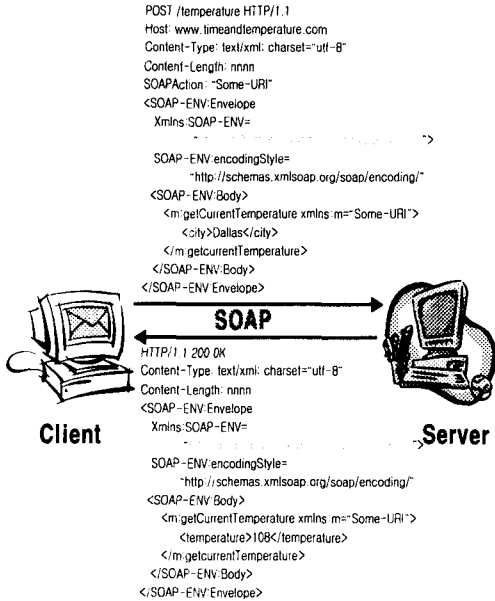


그림 2. SOAP의 메시지 전달 형태

의해 제공된 프로그래밍이 가능한 웹 서비스의 정보를 찾을 수 있다. 또한 이 레지스트리를 사용하면 모든 조직에서 제공할 웹 서비스에 대한 정보를 게시할 수 있다. 프로그래머의 인터페이스는 XML, SOAP 및 HTTP에 능숙한 프로그래머가 프로그램에서 직접 레지스트리와 상호 작용할 수 있게 하는 UDDI 레지스트리 사양의 일부로 정의되어 있다.

그림 3은 간단한 구성 요소 집합과 UDDI에 대한 이들의 관계를 보여준다. 프로그램에서의 요청 서식 설정, 요청 관리자로의 요청 전송 및 응답 수신을 가능하게 하는 세 개의 기본 개체가 나타나 있다. 요청 관리자는 모든 XML 및 SOAP 세부 정보, 인증 및 오류 관리를 전부 숨긴다<sup>[12]</sup>.

UDDI를 사용하는 주 목적은 다른 사용자에게 자신의 존재를 알리고 업무와의 상호 작용에 사용할 수 있는 웹 서비스를 제공했음을 알리는 것이다. 이러한 서비스의 예로 구매 주문, 송장, 발송 d통지 등의 업무 문서를 위한 서비스를 들 수 있다. 제공한 서비스를 관리하는데 사용하는 도구에 따라 UDDI 레지스트리를 사용하여 해당 서비스에 대한 기술 정보를 광고할 수도 있다.

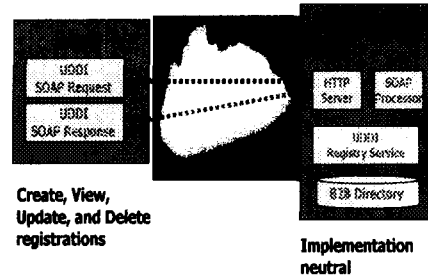


그림 3. UDDI와 SOAP

조회를 수행하여 협력업체가 제공한 서비스에 대한 정보를 찾는 것과 다른 사용자가 자신의 정보를 검색 및 연결할 수 있도록 관리하는 것으로 나눈다.

### 3.2.1 게시 작업

게시를 포함한 모든 상호 작용에는 인증된 연결이 필요하다. UDDI 레지스트리에 데이터를 게시하려면 우선 레지스트리 웹 사이트를 방문하여 데이터를 관리하는 데 사용할 특정 UDDI 운영자를 선택해야 한다. 운영자를 선택한 후에는 (여러 운영자는 각각 전체 데이터의 일부를 관리함) 게시 자격 증명에 등록한다. UDDI 게시 API를 사용하기 위해 UDDI 게시 서버에 연결하려면 대개 사용자 이름과 암호 쌍으로 이루어진 이러한 자격 증명이 필요하다.

### 3.2.2 조직 등록

우선 수행할 작업 중 하나가 주요 조직 정보를 등록하는 것이다. 조직 정보는 조직 이름과 설명 및 연락처를 비롯한 업무 또는 조직에 대한 간단한 데이터이다.

UDDI 레지스트리에 있는 주요 자료 형식이 그림 4에 나타나 있다. 그 내용을 보면, businessEntity, businessService, bindingTemplate, 그리고 tModel이 있다. businessEntity는 비즈니스에 관한 정보를 제공하면서 하나 또는 여러개의 businessService를 포함한다. 비즈니스는 서비스 제공자를 말한다. 웹 서비스에 대한 기술적이고 비즈니스적인 정의는 businessService와 businessTemplates에서 정의된다.

bindingTemplate는 한 개 이상의 tModel을 포함한다. tModel은 서비스를 위한 기술적 사양을 정의한다.

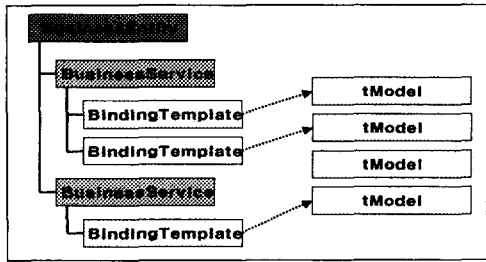


그림 4. UDDI 자료 형식

### 3.2.3 WSDL(Web Services Description Language)

통신 프로토콜 및 메시지 형식이 웹에서 표준화됨에 따라, 통신을 어느 정도 구조적인 방법으로 설명하는 것이 점차 가능해지고 중요해지게 되었다. WSDL은 네트워크 서비스를 메시지를 교환할 수 있는 통신 종점의 컬렉션으로 설명하는 XML 문법을 정의하여 이러한 필요성을 해결하고 있다. WSDL 서비스 정의는 분산 시스템에 대한 설명서를 제공하고 어플리케이션 통신과 관련된 자세한 정보를 자동화할 수 있도록 한다.

WSDL에서 종점 및 메시지의 추상 정의는 구체적인 네트워크 구축 또는 데이터 형식 바인딩과는 구분되며 이러한 특성은 추상 정의를 재사용 가능하도록 하고 있다. 즉, 메시지는 교환되는 데이터의 추상 설명이고, 포트 유형은 오퍼레이션의 추상 컬렉션이다. 특정 포트 유형에 대한 구체적인 프로토콜과 데이터 형식 지정은 재사용 가능한 바인딩을 구성한다. 포트는 네트워크 주소를 재사용 가능한 바인딩에 연결하여 정의되고, 포트의 컬렉션은 서비스를 정의한다. 따라서 WSDL 문서는 네트워크 서비스의 정의에서 다음과 같은 요소를 사용하고 있다.

**types**는 교환된 메시지를 설명하는 데 사용된 데이터 형식 정의를 제공한다.

**message**는 전송되는 데이터의 추상 정의를 나타낸다. 메시지는 논리적인 부분으로 이루어져 있으며 각 부분은 특정 형식 시스템 내의 정의와 연관되어 있다.

**portType**은 추상 작업의 집합이며 각 작업은 입력

메시지 및 출력 메시지를 참조한다.

**binding**은 특정 portType에 의해 정의된 메시지 및 작업에 대해 구체적인 프로토콜과 데이터 형식을 지정한다.

**port**는 바인딩에 대해 주소를 지정하기 때문에 단일 통신 종점을 정의한다.

**service**는 관련된 포트의 집합을 집계하는 데 사용된다<sup>[13]</sup>.

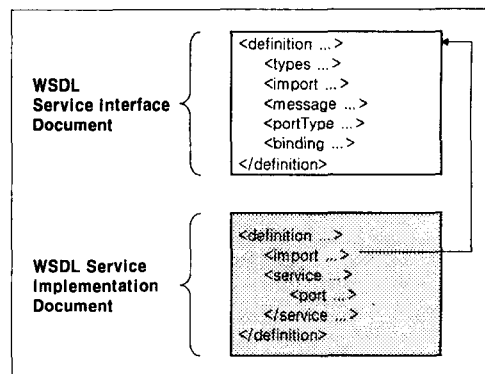


그림 5. WSDL 문서 형식

WSDL이 새로운 형식 정의 언어를 도입하는 것은 아니고, 메시지 형식을 설명하는 다양한 형식 시스템의 필요성을 인식하여 정식의 형식 시스템으로 XML 스키마 규격(XSD)을 지원한다. 그러나 현재의 메시지 형식을 포함하여 앞으로 사용하게 될 모든 메시지 형식을 설명하는 데 있어 단일 형식 시스템 문법을 기대하기 어렵기 때문에, WSDL은 확장성을 통해 다른 형식의 정의 언어를 사용할 수 있도록 하고 있다. 또한 WSDL은 일반적인 바인딩 메커니즘을 정의한다. 이 메커니즘은 특정 프로토콜이나 데이터 형식 또는 구조를 추상 메시지, 작업 또는 종점에 연결하는 데 사용되고 추상 정의를 재사용할 수 있도록 한다. UDDI에 정의되어 있는 WSDL 서비스를 검색하고, 이용하는데, WSDL 문서는 service interface와 service implementation 두 부분으로 구분된다. 그림 5.에 그 구조를 보이고 있다. Service interface 문서는 하나 또는 그 이상의 서비스를 구현하는데 사용될 WSDL 서비스를 정의하고, service interface provider에 의해 개발되고

지원된다. Service implementation 문서는 서비스 인터페이스를 구현할 서비스를 정의하고, service provider에 의해 생성되고 사용된다.

service interface는 UDDI의 tModel에 제공되므로서 서비스를 재사용할 수 있도록 정의한다. service implementation은 서비스의 인스턴스를 정의한다. 각 인스턴스는 WSDL 서비스 엘리먼트에 정의되어 있고, service implementation 문서에 있는 각 서비스 엘리먼트는 UDDI의 BusinessService에서 사용된다. WSDL 서비스가 정의될 때, service implementation이 BusinessService에 표현되기 전에 service interface는 tModel에 정의되어야 한다. 그림 6.은 WSDL과 UDDI의 맵핑을 나타낸 것이다.

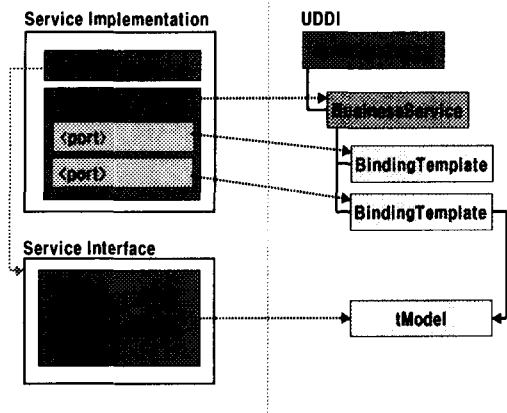


그림 6. WSDL과 UDDI의 mapping

### 3.2.4 SOAP과 UDDI, WSDL의 관계

인터넷에서 전자상거래 뿐만 아니라 다양한 웹 서비스를 지원하기 위해서는 자료 전송을 담당하는 미들웨어로 CORBA보다 훨씬 간편한 SOAP을 이용하고 있는 추세이다. SOAP을 이용하여 자료를 전송하게 하고, 서비스를 제공하는 측은 WSDL로 UDDI에 SOAP을 통하여 서비스를 등록하게 된다.

그림 7.은 기본 웹 서비스 구조를 보여주고 있다. 이 구조에서는 자동적으로 자료전송이 가능하게 하고 있다.

1. Service provider는 서비스를 제공하고, WSDL을 통해 서비스 브로커에게 서비스를 전달한다.
2. Service broker는 UDDI 레지스트리로 접근을 한

다. Service provider가 레지스트리로 알리게 되면, service requestor는 레지스트리로 요청된 서비스를 갖고 있거나, 검색하게 된다.

3. Service requestor는 UDDI를 통해 service broker에서 서비스를 검색하고, SOAP을 통해 service requestor에게 서비스를 바인딩한다.

UDDI 레지스트리는 비즈니스 이름과 바인딩 정보, 어플리케이션 인터페이스 지점, 또는 WSDL을 포함하고 있다. CORBA나 RMI같은 분산된 객체 구조와는 달리, SOAP은 단지 header와 body를 갖고 있는 프로토콜이다. Service requestor가 적당한 웹 서비스를 찾게 되면, service requestor로 SOAP 바인딩을 효율적으로 하기 위해 service broker의 레지스트리에서 정보를 사용한다. WSDL은 웹 브라우저와 웹 사이트 사이에 서로 활동을 묘사하기 위해 HTTP GET이나 POST 방식을 사용한다.

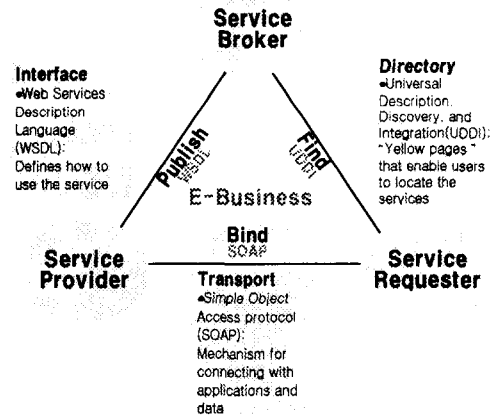


그림 7. SOAP과 UDDI, WSDL의 관계

## IV. 구현

본 연구에서는 마켓플레이스 에이전트를 구현하기 위해서 XML과 SOAP, UDDI, WSDL을 사용하였고, 응용을 위해서 마이크로소프트사의 닷넷 환경에서 C#을 이용하였다. 전체적인 구현 구성도를 보면 그림 8.과 같다.

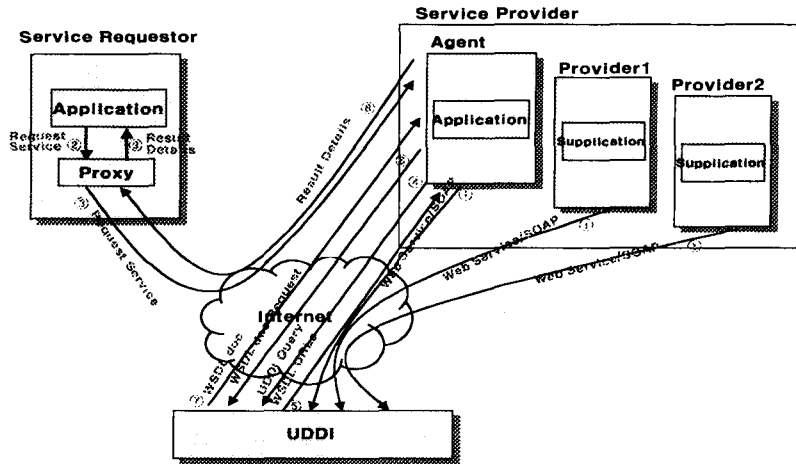


그림 8. 시스템 구성도

구성도에서는 여러 웹 서비스 제공자중의 하나를 마켓플레이스로 하고, 여기에 에이전트를 구축하였다. 여러 웹 서비스 제공자는 SOAP으로 UDDI 레지스트리에 제공할 서비스를 등록을 한다. 등록된 내용은 상품에 대한 카탈로그와 자세한 상품정보, 구매 요구서 등이 포함된다. 그래서 웹 서비스 요구자가 언제든지 서비스를 이용할 수 있도록 한다. 서비스 요구자가 특정 상품을 요구하면, 마켓플레이스에 있는 에이전트는 SOAP으로 UDDI 레지스트리에 접속해서 UDDI Query를 보낸다. UDDI 레지스트리에서는 해당 서비스에 대한 응답을 역시 SOAP을 통해서 WSDL로 된 URL을 전송한다. WSDL은 다른 웹 서비스들과 함께 쉽게 작업할 수 있도록 표준으로 에이전트에게 전달된다. 즉, 에이전트의 웹 서비스 어플리케이션은 웹 서비스를 이용하기 위해서 다른 추가적인 프로그래밍이 필요 없다.

이때 에이전트는 SOAP으로 UDDI 레지스트로부터 직접 연결하여 전달된 URL주소를 사용하여 URL이 지정한 위치에서 WSDL 문서를 검색한다. WSDL 문서는 필요한 여러 가지 웹 서비스를 어떻게 호출할 것인가를 나타낸다. 그리고 웹 서비스 제공자의 카탈로그와 상품 검색, 자세한 가격 정보 등을 사용할 수 있게 한다.

현재 에이전트는 응답을 서비스 요구자에게 보낼 필요가 있는 정보를 가지고 있다. 그 정보가 요구자에게 전달되면 요구자는 응답 목록에서 상품을 선택하

고, 자동적으로 서비스 제공자를 선택하게 된다. 에이전트는 요구서에 있는 서비스 제공자로부터 웹 서비스를 사용하기 시작하게 되고, 서비스 제공자의 WSDL 문서로부터 제공자 웹 서비스를 재차 사용할 수 있게 된다. 이러한 요구에 대한 일련의 과정은 SOAP을 통해서 이루어진다. 그러면 서비스 제공자는 에이전트를 통해서 상품 요구서를 수용하도록 한다.

선택된 상품에 대한 정보를 가지고 있는 서비스 제공자는 서비스 요구자의 주소와 지불 정보를 에이전트로부터 가져오고, 서비스 요구자에서의 어플리케이션에서는 통신하는 정보를 proxy에 일단 저장할 하게 된다. 마켓 카트나 돈 같은 저장을 요하는 자료는 세션을 통해서 처리하도록 한다. 이러한 과정은 계속적으로 진행된다. 다음 번에도 에이전트는 이러한 처리 과정을 반복할 것이고, 그래서 새로운 서비스 공급자가 UDDI 레지스트리에 서비스를 등록하게 되면 서비스 요구자는 자동적으로 변경된 카탈로그와 상품 정보를 역시 제공받게 된다.

## V. 결론 및 향후 연구방향

인터넷은 단순히 정보를 제공하는 차원이 아니라, 고급의 서비스를 제공하도록 발전되고 있다. 본 논문에서는 웹 서비스를 빠르고, 정확하게 지원할 수 있는 환경을 구축하고, 실제로 마켓플레이스를 위한 에이전



트를 설계하고 구현하였다. 웹 서비스 제공자는 제공할 서비스를 만들어서 WSDL로 정의하고, UDDI에 정의하므로서 에이전트나, 다른 웹 서비스 제공자, 일반 사용자에게 전달될 수 있도록 하고 있다. 이들간의 통신은 HTTP를 사용하는 쉽고, 간편한 프로토콜인 SOAP을 이용하였다. 향후 연구 계획으로는 UDDI를 이용한마켓플레이스 에이전트가 무선 웹에서도 그 기능을 지원할 수 있도록 무선 프로토콜을 개발하여야 하고, 그에 적합한 UDDI도 개발할 예정이다.

### 참 고 문 헌

[1] Alper Caglayan, Colin Harrison, "Agent sourcebook", Wiley & sons, Ind., 1997.

[2] 이은석, "에이전트 기술의 전자상거래 응용", 전자공학회지, Vol.26 No.1, pp61-70, 1999.

[3] Robert B.Doorenbos, Oren Etzioni, and Daniel S.Weld, "A Scalable Comparison-Shopping Agent for the World Wide Web", Proceedings of the first international conference on Autonomous agents, pp.39-48, February 1997.

[4] P.Dasgupta, L.E.Moser, and P.M.Melliari Smith, "MagNet : Mobile Agents for Networked Electronic Trading", IEEE Transaction on Knowledge and Data Engineering, Vol.11, No.5, July/August 1999.

[5] P.Kotzanilolaou, G.Katsirelos, and V.Chrissikopoulos, "Mobile agents for Secure Electronic Transactions", Recent Advances in Signal Processing and Communications, pp.363-368, world Scientific Engineering Society, 1999.

[6] Bob Cut, "Web Services : The New Web Paradigm"

[7] J.Ben Schafer, Joseph Konstan, John Rie이 "Recommender Systems in E-Commerce", Proceedings of the ACM Conference on Electronic Commerce, November 1999.

[8] 정한희, 이은석, 최중민, 한경현, 이준호, "지능형 전자상거래를 위한 온토로지 서버 구축과 개인 적용형 상품검색", 한국정보처리학회 논문지, 제7권 제

5호, 5. 2000.

[9] Skonnard, A., "SOAP : The Simple Object Access protocol", <http://www.microsoft.com/mind/0100/soap/soap.asp/>, Microsoft Internet Developer, January 2000.

[10] Siemeon Simeonov, "Introduction to UDDI", XML Journal, volume:2 Issue:10, 2001.

[11] Siemeon Simeonov, "Deeper into UDDI", XML Journal, volume:2 Issue:11, 2001.

[12] UDDI.org, "UDDI Overview", <http://www.uddi.org/>, 2000.9.

[13] Erik Christensen(Microsoft), Francisco Curbera(IBM), Greg Meredith(Microsoft), Sanjiva Weerawarana(IBM), "WSDL(Web Services Description Language) 1.1", <http://www.microsoft.com/>, 2001.

### 저 자 소 개



최유순(Yue-Soon Choi)

1986 원광대학교 컴퓨터공학과 학사  
1990 원광대학교 컴퓨터공학과 석사  
2000 원광대학교 컴퓨터공학과 박사  
과정 수료  
2001~현재 원광대학교 BK21 겸임  
교수

※ 관심분야 : 소프트웨어공학, 컴포넌트 소프트웨어, 웹프로그래밍, 웹서비스



소경영(Kyung-Young So)

1986 원광대학교 전자공학과 학사  
1990 원광대학교 컴퓨터공학과 석사  
2001 원광대학교 컴퓨터공학과 박사  
1991년~현재 익산대학 컴퓨터과  
학과 부교수

※ 관심분야 : 소프트웨어공학, XML, 전문가시스템, 웹 프로그래밍



신현철(Hyun-Cheul Shin)

1987 서울산업대학교 전자계산학과 학사

1989 광운대학교 전자계산학과 석사

2002 원광대학교 컴퓨터공학과 박사

1994~현재 천안외국어대학 컴퓨터 정보과 조교수

※관심분야 : 정보통신, 정보보안, 소프트웨어공학



한성국(Sung-Kook Han)

1979 인하대학교 전자공학과 학사

1981 인하대학교 전자공학과 석사

1988 인하대학교 전자공학과 박사

1984~현재 원광대학교 컴퓨터공학과 교수

1979~1984 대성전자통신연구소 비상임연구원

1988~현재 미르칸전자 기술고문

1989~현재 (주)비트컴퓨터 기술고문

1990.12~1992.2 미국 펜실버니아대학교 Post-Doc.

※관심분야 : 인공지능(자연언어처리), 정보공학, 인지과학, 객체지향시스템, 컴파일러, 컴퓨터교육, WBI



박종구(Jong-Goo Park)

1969 동국대학교 농업경제학과 학사

1975 동국대학교 전자정보처리학과 석사

1999 동국대학교 통계학과 박사

1981년~현재 원광대학교 컴퓨터

공학과 교수

※관심분야 : 전문가시스템, 소프트웨어공학, 소프트웨어 신뢰성 공학, 모바일 프로그래밍