

# 도서관 서비스를 위한 다중-에이전트 기반의 시맨틱 웹 서비스 시스템

## The Multi-agent based Semantic Web service System for the Library Service

충북대학교 전기전자컴퓨터공학부, 첨단정보기술연구센터(AITrc)

Kyoung Soon Hwang, Keon Myung Lee

School of Electric and Computer Engineering, Chungbuk National University, AITrc

E-mail : hks@aicore.chungbuk.ac.kr, kmlee@cbnu.ac.kr

### 요 약

이 논문에서는 시맨틱 웹 서비스의 기술들을 이용하여 에이전트들이 상황을 파악하고 작업 계획을 동적으로 수행할 수 있는 다중-에이전트 기반의 시맨틱 웹서비스 시스템을 설계하고 도서관 서비스 업무에 적용해 구현 하였다.

#### 1. 서론<sup>1)</sup>

웹 서비스는 분산된 서비스 지향 구조 (SOA: Service Oriented Architecture)의 기반으로 복잡하고 다양한 비즈니스 프로세스를 인터넷과 같은 네트워크상에서 공개하고, 설치하고 실행가능하게 한다[1]. 이와 같은 분산된 웹 환경에서는, 에이전트 기술과 시맨틱 웹 서비스 기술의 결합함으로써 이질적인 환경에 비즈니스 프로세스 통합을 수행하는데, 적응적이고 유연한 서비스를 제공할 수 있다. 또한, 분산된 웹 서비스 환경에서는 서비스의 주체가 사람이 아닌 기계가 될 수 있다. 따라서 기계가 문제를 해석(Interpret)하고 해(Solution)를 제공할 수 있도록 지식을 표현하고 획득 할 수 있는 방법을 제공할 수 있어야 한다.

한편, 시맨틱 웹은 기계가 문서의 판독이 용이하게 의미를 부여하여 데이터 간에 연관 관계를 잘 정의되어 있다. 따라서 에이전트가 문서의 의미를 적절하게 처리할 수 있도록 한다. 하지만, 좀 더 나은 수준의 자동화와 지능화를 이루기 위

해서는 의미가 부여된 데이터뿐만 아니라 데이터에 대한 추론을 규정하는 규칙들을 표현할 수 있어야 하며, 특정 도메인에 대한 용어들 간에 의미적 연관 관계와 추론을 위한 온톨로지가 필요하다. 이와 같이 기술된 온톨로지로부터 에이전트들은 의미있는 데이터를 획득할 수 있다.

하지만, 위에서 언급은 웹서비스, 시맨틱 웹, 그리고 에이전트 기술들은 새로운 것들이 아니다. 이 연구에서는 시맨틱 웹 서비스의 기술을 이용하여 에이전트들이 상황을 파악하고 작업계획을 동적으로 수행할 수 있는 다중-에이전트 기반의 시맨틱 웹서비스 시스템을 설계하고 도서관 서비스 업무를 예를 들어 구현하였다.

이 논문은 다음과 같이 구성된다. 2절에서는 에이전트 기반의 시맨틱 웹 서비스에 대하여 설명한다. 3절에서는 구현된 도서관 서비스 시스템에 대하여 기술하고, 4절에서는 결론을 맺는다.

#### 2. 다중-에이전트 기반의 시맨틱 웹 서비스

웹 서비스의 구조(그림 1)는 서비스를 제공하는 '서비스 제공자'와 서비스를 필요로 하는 "서비스 수요자" 그리고 이들 사이를 중개하는 "서비스 중개자"로 구성된다. 이들 사이에는 서비

본 연구는 첨단정보기술연구센터를 통하여 과학기술부/한국과학재단의 지원을 받았음.

스를 공개하고 그 서비스를 검색한 다음, 제공자와 수요자 사이에 서비스를 결합하는 3가지 기능의 세 요소(제공자, 수요자, 중개자) 사이에 일어난다[1]. 한편, 웹 서비스는 XML을 기반으로 구축되어 있기 때문에 문서에 포함된 의미를 처리하는 데 한계가 있다.

시맨틱 웹 서비스(Semantic Web Services)는 시맨틱 웹과 웹 서비스가 갖고 있는 특징을 결합한 것으로[4], 메타데이터의 개념을 통하여 문서에 의미를 부여하고, 이를 이용하여 에이전트가 의미 있는 정보를 자동으로 추출함으로써 동시에 서비스를 처리할 수 있다. 시맨틱 웹은 서비스 사용자들에 의해 요구되는 서비스들을 탐색(find)하고 공개(publish)하고, 그리고 결합(bind)하는 과정을 구조적으로 자동화하기 위하여 에이전트들을 이용한다.

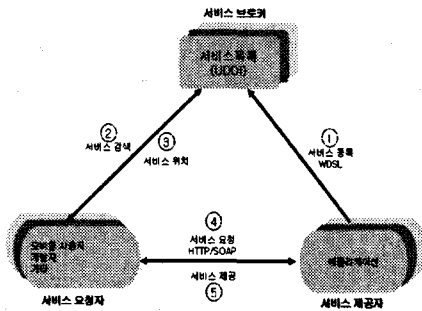


그림 1 웹 서비스 구조 (SOA : Service Oriented Architecture)

W3C에서는 MOM (Message Oriented Model), SOM (Service Oriented Model), 그리고 ROM (Resource Oriented Model)을 포함한 에이전트 기술을 가지는 웹 서비스 구조를 제공하고 있다. 여기서, 에이전트의 개념은 웹 서비스와 연관된 실제 자원을 식별하기 위해 적용된다. 즉 서비스는 추상적인 기능을 제공하는 반면 에이전트 메시지를 주고받아 실제 작업을 수행하는 물리적인 개체(소프트웨어의 일부로서 에이전트)이다[5]. 따라서 추상적인 개념의 서비스는 에이전트의 역할(기능)로 구현될 수 있다.

다중-에이전트 기반의 시맨틱 웹 서비스를 구현하기 위해서는, 에이전트가 서비스와 다른 에이전트들과 통신을 해야 한다. 이때, 통신이 설정된 에이전트 간에는 정보를 공유하기 위하여 상태정보를 저장하는 상태정보(SI : State Information)테이블이 생성된다. 상태정보 테이블의 필드에는 태스크 번호, 태스크 순서 번호,

발신 에이전트 번호, 목적지 에이전트 번호, 확인응답 번호, 그리고 제어 필드가 포함된다. 제어 필드에는 연결을 설정, 종료, 그리고 데이터 전송모드를 포함한다. 에이전트 간에 통신이 종료되면 상태정보 테이블도 삭제된다.

에이전트 간에 통신은 다음과 같은 태스크들로 이루어진다. 다른 에이전트에 메시지를 보내는 send message, 다른 에이전트로부터 메시지를 기다리는 receive message, 메시지를 보내고 확인 응답을 보내는 Request-response, 메시지를 보내고 응답을 기다리는 Solicit-response, 그리고 에러를 알리는 Notify error등이 있다.

또한 에이전트와 서비스들 사이에 통신은 SOAP 프로토콜을 사용한다. 왜냐하면, SOAP는 웹 서비스의 핵심기술로서 분산 환경에서 정보를 교환하기 위한 목적으로 고안된 XML 기반의 경량의 프로토콜로, 수요자와 공급자 사이의 메시징 프로토콜을 정의하기 때문이다. 에이전트와 서비스 사이에 통신 메시지 형식은 다음과 같다.

- Message Header

- From : 보내는 주소
- To : 받는 주소
- Protocol : 프로토콜 버전
- Session : 섹션 식별자
- Version : 버전
- Order : 프로토콜의 메시지 순서

- Message Body : 메시지 내용

```
<MessageHeader>
  <form>http://203.255.71.95/#agent_collection </form>
  <to>http://203.255.71.95/#service_onto</to>
  <protocol>ASC</protocol>
  <session>loaninfo</session>
  <version>1.0</version>
</MessageHeader>
<MessageBody>
  <loaninfo>itloaninformation</loaninfo>
  <userid>2004150001</userid>
  <accessionno>EM00095695</accessionno>
  <loantype>0001</loantype>
  <loandate>2004.10.20</loandate>
  <returndate>2004.11.4</retrundate>
  <postponecnt>0<postponecnt>
  <remark>CD00095695</remark>
</Messagebody>
...
```

그림 2 에이전트와 서비스 간의 통신 메시지

또한, 시맨틱 웹에서는 문서에 부여된 의미를



된다.

웹 서비스 부분에서, 서비스 제공자는 도서검색, 참고도서추천, 도서구입신청, 도서구입, 도서정리, 도서 반납.연체자 통보, 사용자의 수강신청 정보 검색, 담당교수의 강의계획서 정보 검색, 그리고 각종 통계와 같은 업무 등을 제공하게 된다. 우선 서비스 제공자는 레지스트리(UDDI)에 서비스를 등록하고, 레지스트리에서는 서비스 제공자에 의해 등록된 정보가 저장된다.

스케줄링 에이전트는 서비스 요구자인 작업에이전트가 UDDI 레지스트리로 서비스를 검색하고 서비스 중개자인 UDDI 레지스트리로부터 그 결과를 전달받아 실제 서비스 제공자에게 서비스를 요청하고 그 결과를 받아 사용자에게 전달하는 역할을 한다.

### 3.2 가상 시나리오

본 논문에서 시스템 구현을 위한 가상의 참조도서추천 서비스에 적용하여 구현하였다. 서비스 흐름은 [그림 5]에서 볼 수 있다

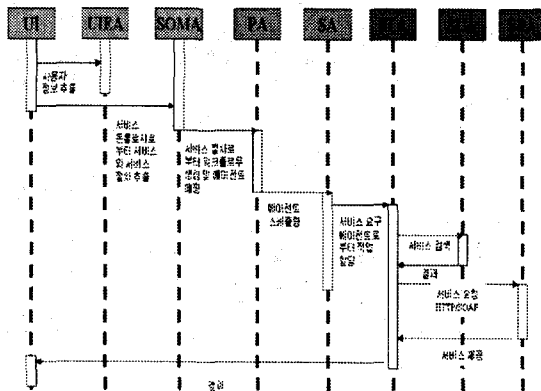


그림 5 서비스 흐름도

- 서비스 사용자가 사용자 인터페이스를 이용하여 서비스를 요청한다.
- 사용자 관리 에이전트로부터 사용자 프로 파일을 이용하여 사용자 정보를 추출한다.
- 온톨로지 에이전트로부터 서비스 종류와 서비스 절차를 생성한다. 예를 들어, 서비스 절차는 사용자 정보로부터 도서를 검색하고 (관련 도서정보와 일치하는 도서가 없는 경우), 신규 도서신청 생성(사용자와의 상호작용을 통해), 결제서류를 생성(도서구입이 결정되면), 그리고 도서구입신청서 생성한다.

- 계획 에이전트는 서비스 절차 정보를 이용하여 에이전트 저장소로부터 에이전트에 대한 워크프로우를 생성한다.
- 스케줄링 에이전트는 작업흐름에 맞게 에이전트들에게 작업을 할당한다. 즉 에이전트들의 action() 메서드를 초기화 및 실행한다. 또한, 에이전트 간에 통신이 일어나면 에이전트들 사이에 정보를 공유하기 위한 상태정보 테이블을 생성, 유지, 그리고 삭제 관리한다.
- 작업 에이전트들은 UDDI 레지스트리로부터 서비스를 검색하여 서비스 제공자에게 서비스를 요청하고 그 결과를 반환 받아 사용자에게 제공한다.

### 4. 결론

제안된 시스템은 시맨틱 웹 서비스의 기술을 이용하여 에이전트들이 상황을 파악하고 작업계획을 동적으로 수행할 수 있는 다중-에이전트 기반의 시맨틱 웹서비스 시스템을 설계하고 가상의 도서관 서비스 업무를 예를 들어 구현하였다. 한편, 이 논문은 특정 도메인을 대상으로 작업을 수행하는데 적합하게 구현되었다. 따라서 향후연구과제로는 넓은 범위에 지식을 표현하고 공유할 수 있는 온톨로지 언어와 추론을 위한 틀을 표현에 관한 연구가 필요하다.

### 5. 참고문헌

[1] Tyler Jewell, David Chappell, "java web services", O'REILLY, 2002  
 [2] Nicholas Gibbins, Stephen Harris, Nigel Shadbolt, "Agent-Based Semantic Web Services", journal of web Semantics, 2003.11  
 [3] Anurpriya Ankolekar, Mark Burstein, "DAML-S : Semantic Markup for Web Service",  
 [4] Michael K. Smith, Chris Welty, Deborah L. McGuinness, "http://www.w3.org/TR/2004/REC-owl-guide-20040210/  
 [5] Chun-Che Huang, Tzu-Laing(Bill), Tseng, Roger R. Gung and Hsuan-shao Chung, "Agent-Based Web Services solution to Collaborative Product Design", international journal of knowledge-based and Intelligent Engineering System 9 (2005) 63-79