

시맨틱 웹 서비스 기술을 위한 수행 온톨로지

오 지 훈* · 최 병 석** · 정 영 식*** · 주 수 종**** · 한 성 국****

요 약

DAML-S/OWL-S, WSMF 및 BPELAWS와 같은 웹 서비스 기술방식은 웹 서비스의 기능적(functionality) 측면만을 표현하고 있기 때문에, 웹 서비스(Web Service)의 개념적, 의미적 기능을 표현하기에는 한계가 있다. 본 연구에서는 웹 서비스의 기능적 측면과 의미적 측면을 통합적으로 표현하기 위해서 수행 온톨로지(Competence Ontology)를 구축하여 웹 서비스를 기술하기 위한 새로운 방법을 제시한다. 또한 시스템 내부적으로는 Data Mediator(D-Mediator)와 Control Mediator(C-Mediator)를 이용하여 서로 이질적인 서비스간의 상호연동 및 합성을 가능하게 한다.

Competence Ontology for Semantic Web Services Description

Ji-Hoon Oh* · Byeong-Seok Choi** · Young-Sik Jeong***
Su-Chong Joo**** · Sung-Kook Han****

ABSTRACT

The Web Services descriptions such as DAML-S/OWL-S, BPELAWS and WSMF focusing on the functional aspects of Web Services have limitations for the representation of the conceptual and semantic capabilities of Web Services, although WSMF is based on ontology and can represent the goal of Web Services. This paper proposes the new description formalism based on the competence ontology that can represent both functional and semantic aspects of Web Services. This paper also presents the integration and the composition of Web Services by means of Data Mediator(D-Mediator) and Control Mediator(C-Mediator) to mediate compositional in compatibility between heterogeneous Web Services.

키워드 : 지식표현(Knowledge Representation), 온톨로지(Ontology), 시맨틱 웹 서비스 기술(Semantic Web Services Description)

1. 서 론

웹(Web)은 인프라가 풍부하고 확장성이 좋아 기존의 서비스들을 웹 환경으로 이전하려고 하는 노력이 계속 있어 왔으나 이는 쉬운 일이 아니었다. 무엇보다도 현재의 컴퓨팅 환경은 이기종이 복합적으로 존재하고 다양한 종류의 어플리케이션, 프로토콜, 포맷들이 혼재하는 매우 복잡한 상태이기 때문에 이들을 하나로 통합하기 위해서는 어느 정도 정형화된 규약이 필요하다. 더불어 보안 문제가 지속적으로 제기되는 환경에서 방화벽을 경유해 데이터와 메시지를 전달할 방법도 찾아야 한다. 경우에 따라서는 원격지에 있는 서비스 객체나 API를 사용할 방법도 강구해야 한다. 그러므로 웹의 개념이 점점 확장되면서 웹이 단순한 정보 인프라로서의 역할을 넘어 기존 정보 인프라 환경에서 제공할 수 없었던 다양한 서비스 제공이 가능한 웹 서비스(Web Services)의 개념이 등장하였다[5, 10, 11]. 현재의 웹 서비스 표준 기술들은 개발자들에 의한 단편적인 서비스

연결 및 서비스에 대한 사용은 가능하지만 서비스에 대한 상호작용이나 복합적인 해석이 불가능하며, 일반인들에 의한 해석이나 이용 또한 거의 불가능하다. 즉, 구문적인 확장은 가능하지만 의미적인 요소나 의미적인 확장이 거의 불가능한 형태인 현재의 웹 서비스 구조는 한계를 지니고 있다. 이러한 문제점 때문에 RPC(Remote Procedure Call) 용도나 단순한 서비스 구현 정도에만 이용되고 있다[1, 4]. 이러한 문제점을 해결하기 위해 현재까지 많은 표준안들이 제출되어 왔으나 이러한 표준 기술들은 대부분이 표준화가 완료되지 않거나, 일부 업계만이 지원하고 있는 제한적 표준이 많으므로 표준화 완료 및 기술도입을 위해서는 해결해야 할 문제점이 많다.

웹 서비스의 모든 잠재력을 이끌어내기 위해서는 웹 서비스들에 대한 적절한 기술(description) 방법이 개발되어야 한다. DAML-S/OWL-S, WSMF(Web Service Modeling Framework) 그리고 BPELAWS(Business Process Execution Language for Web Services)와 같은 웹 서비스 기술방식은 웹 서비스의 기능적(functionality) 측면만을 표현하고 있기 때문에, 웹 서비스의 개념적, 의미적 기능을 표현하기에는 한계가 있다. 따라서 이들 표현 방식으로는 웹 서비스에서 가장 중요한 의미적인 웹 서비스 발견, 동적 웹

* 준 회원 : (주)유애텍 종합기술연구소 선임연구원
** 정 회원 : 군장대학교 컴퓨터응용학부 교수
*** 종신회원 : 원광대학교 컴퓨터및정보통신공학부 교수
**** 정 회원 : 원광대학교 전기전자 및 정보공학부 교수
논문접수 : 2003년 12월 3일, 심사완료 : 2004년 6월 3일

서비스 합성등이 불가능하기 때문에 시맨틱 웹의 궁극적인 목표를 실현할 수가 없다. 본 논문에서는 웹 서비스의 기능적 측면과 의미적 측면을 통합적으로 표현하기 위해서 수행 온톨로지(Competence Ontology)를 구축하여 웹 서비스를 기술하기 위한 새로운 방법을 제시하고, 이를 응용하여 웹 서비스 합성 시스템을 통해 수행 온톨로지의 유효성을 검증한다. 또한 시스템 내부적으로는 D-Mediator(Data Mediator)와 C-Mediator(Control Mediator)를 이용하여 서로 이질적인 서비스간의 상호연동 및 합성을 가능하게 한다.

본 논문의 구성은 2장에서 DAML-S/OWL-S, WSMF 그리고 BPEL4WS의 웹 서비스 기술방식을 설명하고, 3장에서는 웹 서비스 기술을 위한 수행 온톨로지의 요구사항과 구성에 대하여 서술한다. 4장에서는 수행 온톨로지를 응용한 웹 서비스 합성 방법에 대하여 기술한다.

2. 웹 서비스 기술 방식 분석

2.1 DAML-S/OWL-S

DAML-S/OWL-S는 사용자가 웹 서비스를 자동적으로 위치(location), 선택(selection), 합성(composition), 감독(monitoring)할 수 있게 하는 적절한 온톨로지(ontology)를 제공하는 것이다. 이에 따라 에이전트(agent)는 자동적으로 서비스 기술(description)을 판독하고 이러한 서비스들을 자동적으로 접근할 있는 것이다. DAML-S/OWL-S에서 서비스에 대한 온톨로지를 구조화하는 데는 다음과 같은 서비스 프로파일(Service Profile), 서비스 모델(Service Model), 서비스 그라운드(Service Grounding)등 3가지 핵심적인 지식 유형이 중요한 잣대가 되며, 이들 각각은 “서비스가 하는 일은 무엇인가?”, “서비스가 어떻게 동작하는가?”, “서비스에 어떻게 접근할 수 있는가?”에 대한 답변이 된다[8].

이러한 프로파일, 모델, 그라운드 등의 세부적인 내용은 각각의 Service 클래스마다 매우 광범위해서 일률적으로 표현할 수는 없다. 그러나 이들이 기본적으로 제공하는 핵심 유형은 존재한다[1, 7, 8]. DAML-S/OWL-S는 프로세스 제어(Process Control)와 서비스 온톨로지(Service Ontology)를 분리하고 있다. 프로세스 제어 온톨로지(Process Control Ontology)가 현재 정의되고 있는 과정이기 때문에 현 상태의 DAML-S/OWL-S는 서비스 감독을 수행하지 못한다. 그럼에도 불구하고 프로세스 제어 온톨로지에 대한 정의는 DAML-S/OWL-S 명세에서 명시적으로 고려되고 있다. DAML-S/OWL-S에 대한 중요한 결여요소 중의 하나는 예외 처리와 결함 처리에 대한 명확히 정의된 접근법을 제공하지 않는다는 것이다.

2.2 WSMF

WSMF(Web Service Modeling Framework)는 자동적으로 웹 서비스를 발견(discovery), 선택(selection), 중재(mediation), 합성(composition)하여 복합 서비스를 만들기 위한

포괄적인 프레임워크(architecture)를 제공하는 것을 목적으로 한다. WSMF는 컴포넌트들의 강력한 분리(strong decoupling)와 모든 서비스들이 특정 요소에 구애받지 않고 자유롭게 의사교환이 가능하게 하는 강력한 중재(strong mediation) 서비스가 기본이 된다[9]. 이런 WSMF의 기본 원리를 만족하기 위해서는 중재자(mediators)가 프레임워크의 핵심요소이며, 다음과 같이 4개의 주요 요소로 구성된다 [1, 9].

- **온톨로지(ontologies)** : WSMF 명세(specification)의 다른 구성요소(element)들에 의해 사용되는 용어(terminology) 들을 제공한다. 또한 용어를 재사용 가능하게 하고, 동일하거나 연결된 용어를 참조하는 컴포넌트들 사이의 상호운용을 지원하기 위해서 WSMF 명세의 다른 요소들에 의해 사용되는 용어를 정의한다.
- **수행능력 저장소(capability repositories)** : 웹 서비스가 수행해야 하는 문제(problem) 들을 정의한다. 이러한 문제들은 서비스를 제공하기 위해 웹 서비스 요구사항을 명세한 선행 조건(precondition)과 서비스 실행후의 미치는 영향을 기술하는 후행 조건(postcondition)으로 구성된다.
- **웹 서비스 기술(web services descriptions)** : 웹 서비스의 다양한 측면을 정의한다. 웹 서비스 기술은 웹 서비스명(web service name), 수행능력 참조(capability reference), 선행 조건과 후행 조건, 입력 데이터와 출력 데이터로 구성된 블랙 박스로 기술된다.
- **중재자(mediators)** : 이질적인 자료구조, 비즈니스 로직, 메시지 교환 프로토콜, 서비스 호출 등에 관련된 상호운용의 문제를 해결한다. 중재자들은 서비스 요청자와 제공자 사이의 완전한 상호운용이 가능하도록 필요한 연산을 수행하여 서비스 요청자와 제공자 사이의 이질성에 대해 중재하여 준다.

WSMF는 시맨틱 웹 서비스 합성을 위해 요구되는 특성들의 핵심 집합을 제공하고 있다. 그러나 현재의 WSMF는 그 명세가 지금 정립되고 있는 과도기이기 때문에, 차세대 웹 서비스를 위한 적절한 프레임워크가 되기 위해서는 웹 서비스의 특성을 고려한 다양한 기술요소들이 추가로 정의되어야 할 필요가 있다.

2.3 BPEL4WS

BPEL4WS(Business Process Execution Language for Web Services)는 비즈니스 프로세스와 비즈니스 상호작용 프로토콜에 대한 형식 명세를 위한 언어를 제공한다. 이는 웹 서비스 호출, 데이터 조작, 오류 보고, 프로세스 종료와 같은 다른 동작들을 하나로 만들어서 복합적인 프로세스를 생성할 수 있다. 비즈니스 프로세스(business process)는 다음과 같은 두 가지 방법으로 기술한다[6].

- 실행 가능한(executable) 비즈니스 프로세스의 구현: 기존 서비스들을 조합하여 새로운 서비스를 정의한다.
- 실행 불가능한(non-executable) 추상 프로세스 기술: 비즈니스 프로토콜은 프로토콜에 포함된 각 부분들의 내부를 노출시키지 않고 메시지 교환 원리를 명세하는 프로세스 기술(descriptions)을 사용한다.

BPEL4WS는 기본적으로 비즈니스 프로세스의 “합성(composition)”을 구현하는 언어이다. BPEL4WS는 웹 서비스 호출, 데이터 조작, 오류 보고, 프로세스 종료와 같은 다른 동작들을 하나로 만들어서 복잡한 프로세스를 정의하고, 비즈니스 프로세스와 비즈니스 상호작용 프로토콜에 대한 형식 명세를 제공한다. 이렇듯 BPEL4WS에서의 웹 서비스 기술(description)은 비즈니스 프로세스와 상당부분 관련되어 있다. 하지만 현재의 BPEL4WS는 서비스 기술에 관한 어떤 의미(semantics)도 제공하지 않기 때문에 자동화된 시맨틱 웹 서비스 합성에 대한 대부분의 요구사항을 만족하지 못하고 있다. 즉, BPEL4WS는 현재 웹 서비스 프로세스를 기술적으로 합성하기 위한 언어이지, 자동화되고 지능적인 시맨틱 웹 서비스의 합성을 위한 기술요소는 아니다.

3. 웹 서비스 기술을 위한 수행 온톨로지

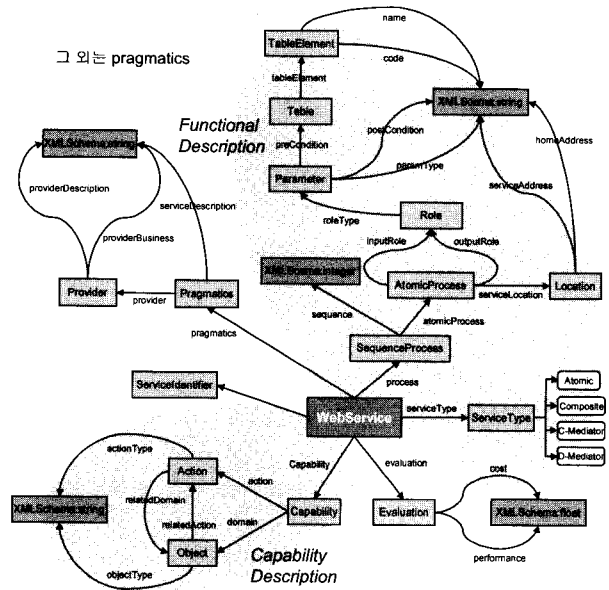
3.1 웹 서비스 기술 요구 사항

기존의 웹 서비스 기술체계인 WSDL(Web Service Description Language)[12]은 웹 서비스에 대한 입력 파라미터, 출력 파라미터, 서비스 제공자 정보, 서비스 위치 정보와 같은 서비스에 대한 기능기술만을 제공할 뿐 자동화된 웹 서비스의 발견, 실행, 합성, 상호운용을 지원하기에는 한계가 있다. WSDL로는 웹 서비스에 대한 의미 정보를 기술하는 것이 불가능하기 때문에 서비스 구문뿐만 아니라 의미까지 기술이 가능한 온톨로지 기반의 웹 서비스 인터페이스 기술이 필요하다.

시맨틱 웹[2]의 경우에는 웹 서비스를 발견하고, 이를 서비스의 소비자와 제공자를 연결하는 매개로 제공하는 방식이 기존의 문서 지향적인 웹 철학을 그대로 이용하고 있다. 시맨틱 웹 기술이 점점 발전하게 되면, 아마도 머지않아 키워드를 이용하지 않고 직접 콘텐츠를 바탕으로 웹 리소스에 접근이 가능하다. 가장 중요한 웹 리소스(web resource)는 서비스(service)이다. 대개 서비스라고 말하는 것은 단순히 정적인 정보만을 웹 사이트를 통해서 제공하는 것을 의미하는 것이 아니라, 제품을 팔거나 물리적인 장치를 구동하는 것과 같은 실질적인 행위가 이루어지도록 하는 것을 말한다. 웹 서비스의 활용을 위해서는 소프트웨어 에이전트(software agent)가 해석할 수 있는 의미적인 서비스의 기술이 필요하며, 여기에 접근할 수 있는 수단이 필요하다[1, 4].

3.2 수행 온톨로지 구성

현재 운영되고 있는 웹 서비스 제공 포털 사이트와 IBM과 Microsoft에서 운영되고 있는 UDDI Business Registry (UBR)를 대상으로 현재 공개되어 운영되고 있는 웹 서비스들에 대한 서비스 행위정보(Action), 서비스 객체정보(Object) 등과 같은 서비스에 대한 의미 정보를 추출하여 수행 온톨로지를 구축한다.



(그림 1) 수행 온톨로지 관계 구성

(그림 1)은 본 논문에서 구축한 수행 온톨로지에 대한 관계 구성을 나타낸 것이다. 이렇게 구축된 웹 서비스 온톨로지는 행위 요소를 의미하는 Action과 행위 요소가 적용되는 도메인을 의미하는 Object와 같은 웹 서비스에 대한 의미 기술과 입출력 파라미터, 입력 파라미터에 대한 프리컨디션과 출력 파라미터에 대한 포스트컨디션등과 같은 웹 서비스에 대한 기능적 기술, 이외의 웹 서비스를 기술하기 위해 요구되는 기본 정보들로 구성되어 있다. 예를 들면, 이메일 전송 서비스를 온톨로지를 이용하여 (그림 1)과 같이 의미 정보 모델링을 한다면, 서비스에 대한 동작 행위 요소인 Action 타입으로써 ‘전송하다’ 또는 ‘보내다’의 의미를 지니는 ‘Send’를 취하게 되고 서비스 대상이 되는 객체요소인 Object 타입으로써 ‘Email’을 취하게 된다. 이외에 기능적 기술은 이메일 전송 서비스가 서비스되고 있는 위치정보(WSDL 위치), 서비스를 제공하고 있는 서비스 제공자 정보, 그리고 서비스 실행을 위한 입출력 파라미터들의 정보 등이 온톨로지 상에 기술된다.

위와 같이 모델링 된 수행 온톨로지를 기술하기 위해 온톨로지 기술언어으로써 DAML+OIL[3]을 이용하였고, 온톨로지 입력은 Protégé-2000을 이용하였다. (그림 1)에서 사각형 요소는 클래스(class)로, 화살표는 속성(property)을 의미한다. 이를 바탕으로 추출된 클래스들과 속성들, 이에 따른

인스턴스들을 입력한다.

합성될 서비스의 타입을 기술하여 주는 ServiceType 클래스 (컨셉)의 경우 인스턴스로서 Atomic, Composite, C-Mediator, D-Mediator 중에 반드시 하나의 값(value)을 가져야 한다. Atomic 타입은 합성 서비스가 아닌 단일 서비스임을 의미하고, Composite 타입은 서비스간의 합성의 작업을 통해 만들어진 합성 서비스임을 의미한다. 그리고 C-Mediator 타입과 D-Mediator 타입은 파라미터를 매칭할 때 이용되는 서비스 타입으로써, C-Mediator는 서비스 출력 파라미터 중에서 자신이 필요한 출력 파라미터만 추출하여 주는 서비스이며, D-Mediator는 임의의 서비스에 대한 출력 파라미터의 타입과 매칭하고자 하는 서비스의 입력 파라미터의 타입을 형변환(conversion)하여 주는 서비스이다.

4. 수행 온톨로지를 응용한 웹 서비스 합성

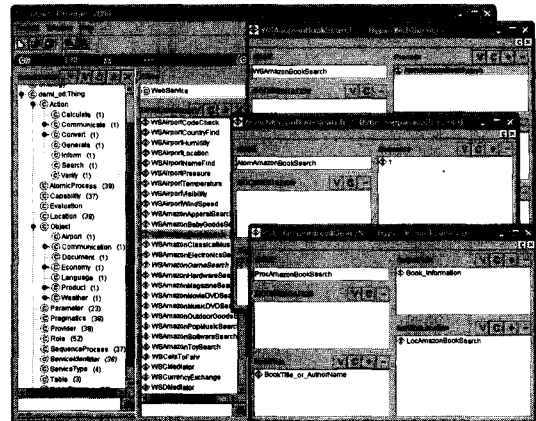
4.1 웹 서비스 도메인 기술

온톨로지 기반의 의미 모델링은 다루고자 하는 도메인에서 사용되는 용어들을 정의하고 그들 사이의 관계를 규정하여 그 과정을 구현하는 것을 의미한다. 이 과정에 포함되는 것으로 개념들의 클래스화, 그 개념들의 관계를 클래스의 계층적인 구조로 정립, 클래스들의 속성과 그 속성에 존재하는 다양성과 제한요소 정의, 마지막으로 인스턴스들의 생성등이 있다. 이를 통해 얻어진 공동의 단어들과 공유된 이해를 바탕으로 상호운용성 의사 소통의 일관성이 가능해진다. (그림 2)는 Protégé-2000 상에서 입력된 수행 온톨로지를 기술한 DAML+OIL 문서의 일부분을 나타내고 있다.

```
<WSOntology_daml:Capability rdf:ID = "CapAmazonMagazineSearch">
  <WSOntology_daml:domain rdf:resource = "#InsMagazine"/>
  <WSOntology_daml:action rdf:resource = "#InsSearch"/>
</WSOntology_daml:Capability>
:
<daml_oil:ObjectProperty rdf:ID = "serviceLocation">
  <rdfs:comment>Web service address with optional home address
  </rdfs:comment>
  <daml_oil:domain rdf:resource = "#AtomicProcess"/>
  <daml_oil:range rdf:resource = "#Location"/>
  <rdf:type rdf:resource = "http://www.daml.org/2001/03/daml+oil#
  UniqueProperty"/>
</daml_oil:ObjectProperty>
```

(그림 2) 온톨로지로 기술된 웹 서비스

DAML+OIL을 이용해 기술된 수행 온톨로지를 이용하여 의미 기반 웹 서비스 선택(selection)과 합성(composition)을 가능하게 하였을 뿐만 아니라 웹 서비스에 대한 기능적 기술을 통하여 입출력정보, 입출력 조건사항, 서비스 위치 정보도 구축된 온톨로지를 기반으로 추출 가능하게 구축한다. (그림 3)은 Protégé-2000 상에서 수행 온톨로지의 입력화면이다.



(그림 3) 수행 온톨로지 입력화면

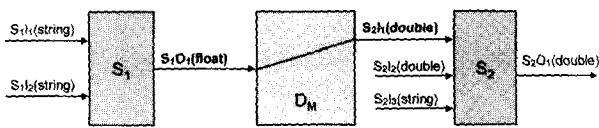
4.2 웹 서비스 합성과 Mediator

서로 이질적인 2개 이상의 서비스를 합성할 경우 2가지 사항을 고려하여야 한다. 첫째, 서로 상이한 데이터 타입간의 매칭을 시도하려 할 경우 발생하는 타입충돌 문제이다. 예를 들면, 임의의 서비스의 출력 파라미터의 타입은 'float' 타입인데, 이를 'string' 타입이 요구되는 입력 파라미터에 파라미터 매칭을 할 경우 이는 타입충돌의 문제를 발생한다. 둘째, 어떤 서비스의 출력 결과가 1개 이상일 경우 입력 파라미터로 매칭할 파라미터를 추출하는 문제이다. 서비스의 모든 결과를 매칭할 경우는 이 사항을 고려할 필요는 없지만, 만약 반환된 서비스의 출력 결과의 필요한 일부분만을 매칭할 경우는 이를 추출하는 프로세스를 추가할 필요가 있다. 그러므로 본 논문에서는 전자의 타입 충돌 문제를 해결하기 위해 서로 상이한 두 타입간의 형변환을 작업을 수행하는 D-Mediator(Data-Mediator)와 후자의 필요한 출력 파라미터 추출을 하기 위해 C-Mediator(Control-Mediator)를 구현하였다.

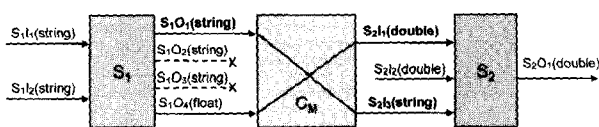
(그림 4.a)는 서로 다른 두 서비스간의 파라미터 매칭시 D-Mediator에 의해 서비스 S₁의 'float' 타입인 출력 파라미터 S₁O₁을 서비스 S₂의 'double' 타입인 입력 파라미터 S₂I₁으로 형변환하는 메커니즘을 보여주고 있다. 예를 들어, 두 국가의 통화간의 환율을 계산하는 서비스 S₁의 출력 파라미터인 환율(float 타입)과 환율(double 타입), 환전금액(double 타입)등을 입력 파라미터로 받아 환전하는 환전 서비스 S₂를 합성을 수행할 경우, (그림 4.a)와 같이 타입 충돌 문제가 발생한다. 이때 D-Mediator가 S₁의 S₁O₁(float 타입)을 S₂의 S₂I₁(double 타입)으로 형변환을 수행하여 줌으로써 이런 문제를 해결하여 준다. 만약 숫자형 string 타입이 아닌 일반 string 타입을 int나 float으로 형변환을 시도할 경우, 이는 허용되지 않는 형변환이므로 적절한 예외처리가 수행된다.

반면, (그림 4.b)는 C-Mediator에 의해 서비스 S₁의 출력 파라미터 S₁O₁, S₁O₂, S₁O₃, S₁O₄ 중에서 S₁O₁과 S₁O₄만을 추출하여 서비스 S₂의 입력 파라미터 S₂I₁과 S₂I₃로 각각 매

청되는 매커니즘을 보여주고 있다. 예를 들어, 저자명(string 타입)과 서명(string 타입)을 입력 파라미터를 입력받아 도서를 검색하여 출판일(string 타입), 출판사(string 타입), 도서가격(float 타입)등을 출력 파라미터로 지니는 아마존 도서검색 서비스 S_1 과 환율(double 타입), 환전금액(double 타입)등을 입력 파라미터로 받아 환전하는 환전 서비스 S_2 를 합성을 수행할 경우, 타입 충돌 문제뿐만 아니라 파라미터 추출문제도 발생한다. 이때는 (그림 4.a)와 같이 D-Mediator가 수행되기 이전에 출력 파라미터들 중 S_2 의 입력 파라미터들과 매칭할 파라미터들을 S_1 에서 추출하는 C-Mediator가 수행된다. (그림 4.b)에서 아마존 도서검색 서비스(S_1)의 S_{1O_1} 과 S_{1O_4} 만을 환전서비스(S_2)의 S_{2I_1} 과 S_{2I_3} 로 각각 매칭을 수행한다. 이때 4개의 출력 파라미터들 중에서 S_{1O_1} 과 S_{1O_4} 만을 추출해 내기 위해 C-Mediator가 수행된다. 이후 S_{1O_1} 과 S_{2I_3} 를 매칭할 경우 동일한 타입이므로 D-Mediator가 수행될 필요가 없지만, S_{1O_4} (도서가격: float 타입)와 S_{2I_3} (환전금액: double 타입)을 매칭할 경우 상이한 두 타입간의 적절한 형변환을 위해서 D-Mediator가 수행된다. D-Mediator는 앞에서 언급하였듯이 C-Mediator가 수행된 후에 실행된다.



(그림 4.a) D-Mediator



(그림 4.b) C-Mediator

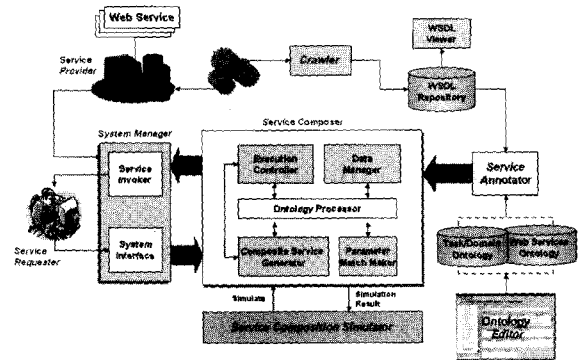
본 절에서 설명한 D-Mediator와 C-Mediator는 향후 합성된 서비스의 프로세스 기술의 일관성을 기하고, 합성된 서비스를 재사용할 때 좀더 용이 할 수 있도록 웹 서비스 형태로 구현하였다. D-Mediator와 C-Mediator가 웹 서비스 합성 프로세스에 기술될 때 'serviceType' 속성으로 각각 D-Mediator와 C-Mediator 타입을 지닌다.

4.3 웹 서비스 합성 시스템

수행 온톨로지를 적용한 웹 서비스 합성 시스템은 Service Ontology(서비스 온톨로지부), Service Composer(서비스 합성부), System Manager(시스템 관리부), WSDL Crawler(WSDL 확인부) 이렇게 4개의 모듈로 구성되고, 시스템 구조는 (그림 5)와 같다.

Service Ontology(서비스 온톨로지)는 웹 서비스 합성할 때 제공되는 웹 서비스에 대한 의미적 기술 정보와 기능적 기술 정보를 제공하여 주는 역할을 한다. 수행 온톨로지에서

제공되는 정보들과 WSDL Repository(WSDL 저장소)에 저장되어 있는 웹 서비스 위치정보를 조합하여 Service Annotator(서비스 주석기)가 추후 Service Composer 파트에서 웹 서비스 합성시 요구되는 여러 의미 정보를 제공한다.



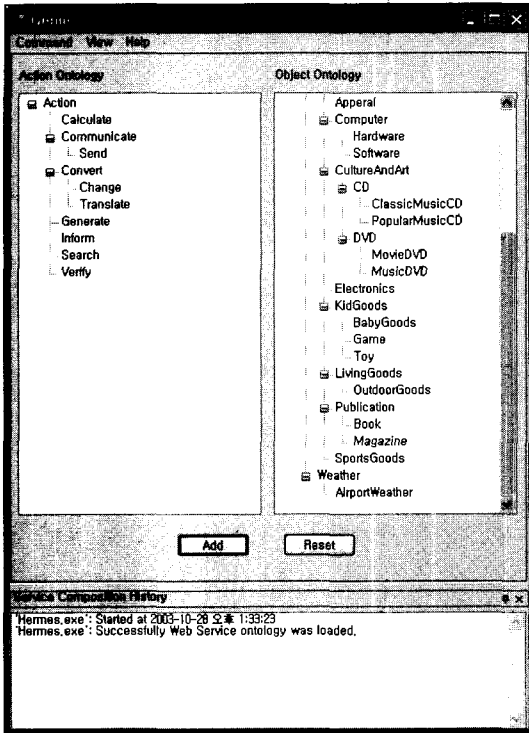
(그림 5) 웹 서비스 합성 시스템의 구조

Service Composer(서비스 합성) 모듈은 Service Annotator에서 넘어온 해당 서비스 정보들을 바탕으로 실제 서비스 합성을 하여 주는 가장 핵심적인 부분이다. Service Composer 모듈 내부에는 서비스 실행의 제어자 역할을 하여 주는 Execution Controller(실행 제어자), 서비스 합성시 합성될 서비스 간에 파라미터 매칭시 파라미터 타입들을 관리 및 변환하는 Data Manager(데이터 관리자), Execution Controller에서의 서비스 실행 정보를 바탕으로 실제 서비스들을 합성하기 위한 Composite Service Generator(합성 서비스 생성기) 및 Data Manager에서 합성될 서비스들의 파라미터 매칭 가능 여부를 기초로 실제 서비스들간의 파라미터 매칭을 수행하는 Parameter Match Maker(파라미터 매칭기)로 구성된다. 또한, 실제 서비스들 간의 합성 작업을 수행하기 전에 서비스 합성이 가능한 조합인지를 먼저 모의적으로 테스트해보기 위한 Service Composition Simulator(서비스 합성 시뮬레이터)등을 두어 효율적으로 서비스를 합성한다.

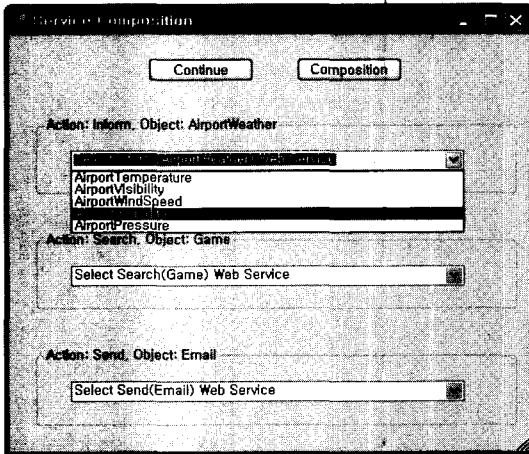
System Manager(시스템 관리자) 모듈은 Service Annotator에서 넘어온 서비스 위치 정보와 Service Composer 상에서의 서비스 매칭정보를 바탕으로 실제 서비스를 실행한다. 이는 서비스를 호출해 주기 위한 Service Invoker와 호출될 서비스로 넘겨줄 서비스 입력 파라미터를 입력하는 System Interface(시스템 인터페이스)로 구성되어 있다. WSDL Crawler(WSDL 확인) 모듈은 합성 시스템에서 지원되는 웹 서비스의 상세 내용, 즉 WSDL 파일을 확인한다.

웹 서비스를 합성하기 위해서는 합성할 서비스를 선택해야 한다. (그림 6)과 같이 로딩된 메인화면 상에서 합성할 서비스의 Action과 해당 Object를 선택한 후 'Add' 버튼을 클릭하면 웹 서비스 합성기는 선택된 Action과 Object에 해당되는 웹 서비스들을 웹 서비스 온톨로지를 분석한 후 이를 추출하여 합성 서비스 목록에 추가한다. '합성'이라는 의

미가 최소 2개의 서비스를 조합할 때를 의미하기 때문에 서비스를 합성하기 위해서는 합성 서비스 목록에 최소 2개 이상의 서비스가 추가되어야 한다.



(그림 6) 웹 서비스 합성 시스템의 메인 화면

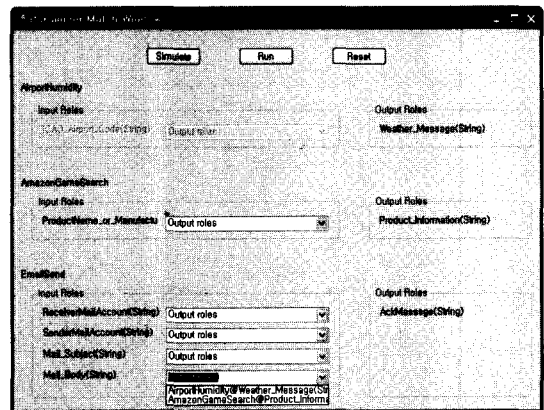


(그림 7) 합성될 서비스 선택화면

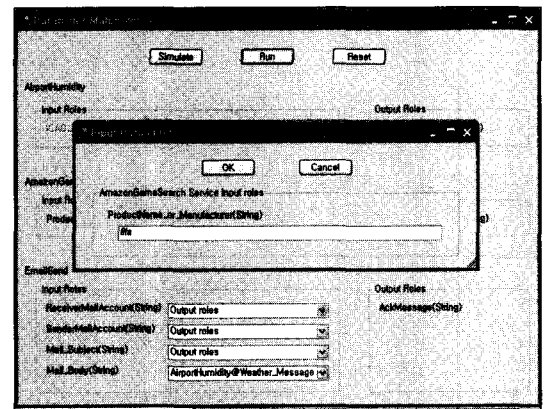
합성될 서비스를 계속 추가하기 위해서는 'Continue' 버튼을 클릭한다. 만약 합성될 서비스들을 모두 추가하였다면 'Composition' 버튼을 클릭하여 추가된 서비스들에 대한 서비스 합성작업을 시작한다. (그림 7)은 의미적으로 'Inform' Action과 'AirportWeather' Object에 해당하는 서비스, 'Search' Action과 'Game' Object에 해당하는 서비스, 그리고 'Send' Action과 'Email' Object에 해당하는 서비스들을 합성하는 예를 나타낸 것이다. 화면상의 컴보박스를 클릭하

여 보면 웹 서비스 온톨로지에서 추출된 서비스 목록을 확인할 수 있다.

(그림 7)에서 'Composition' 버튼을 클릭하면, (그림 8)과 같은 서비스 파라미터 매칭화면이 나타난다. (그림 8)에서는 공항의 습도정보를 제공하는 서비스인 AirportHumidity 서비스, Amazon 사이트에서 게임 제품을 찾아주는 Amazon GameSearch 서비스, 그리고 e-mail을 전송하는 서비스인 EmailSend 서비스의 입력 파라미터와 출력 파라미터를 매칭시키기 위한 화면이다. 여기서는 EmailSend 서비스의 'EmailBody'라는 입력 파라미터에 AirportHumidity 서비스의 출력 파라미터인 'Weather_Message'를 매칭시킨다. 추후 서비스 입력창에서 위의 'EmailBody'를 입력하는 텍스트 박스에는 AirportHumidity 서비스의 출력 파라미터인 'Weather_Message' 결과가 자동으로 입력된다. 'Simulate' 버튼을 클릭하면 위에서 설정된 파라미터 매칭정보가 적절한지에 대해서 테스트가 가능하다. 파라미터 매칭 설정이 끝났다면, 'Run' 버튼을 눌러 실제 서비스 제공자측에서 제공하고 있는 원격지의 웹 서비스를 실행할 수 있다.



(그림 8) 파라미터 매칭화면



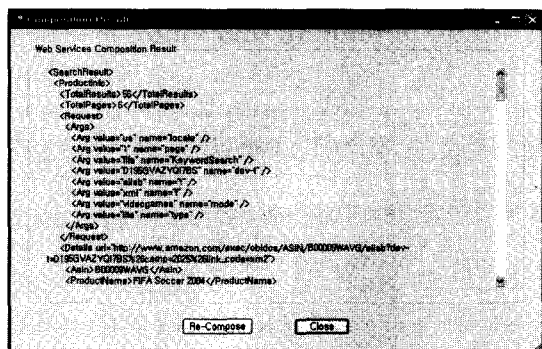
(그림 9) 파라미터 매칭 정보가 없는 입력 파라미터 입력화면

(그림 9)는 웹 서비스를 실행하기 위한 실제 파라미터 입력단계를 보여주고 있다. 이 단계에서 데이터를 입력하면,

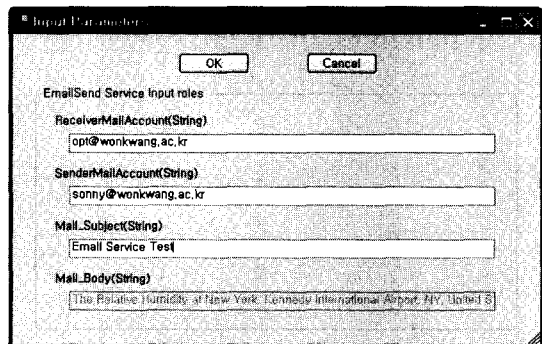
입력된 데이터를 서비스가 위치한 원격지로 SOAP 메시지 형태로 전송한다. 서비스 호스트에서는 요청된 서비스를 실행하여 그 결과 값을 다시 SOAP 메시지 형태로 서비스를 호출한 클라이언트로 반환한다.

파라미터 매칭이 설정되지 않은 서비스 경우는 데이터를 입력 후 (그림 10)과 같이 서비스 실행 결과를 바로 확인해 볼 수 있지만, 파라미터 매칭이 설정된 서비스 경우에는 출력 파라미터가 후에 실행될 서비스의 입력 파라미터로 매칭이 되었기 때문에 (그림 11)과 같이 다음 서비스 과정에서 파라미터 매칭이 적용된 입력 파라미터 입력 텍스트 박스에 매칭된 서비스의 출력 파라미터 결과값이 자동으로 입력된 후 데이터 수정을 할 수 없도록 비활성화 처리된다.

지금까지 살펴본 바와 같이 본 논문에서는 웹 서비스의 의미적 요소들을 온톨로지 기술함으로써 기존의 서비스 기술방식이 한계를 드러냈던 서비스에 대한 의미 정보기술을 가능하게 하였다. 또한, 웹 서비스간의 의미적 상호운용을 지원하여 자연스럽게 내부 또는 외부의 이질적인 어플리케이션간의 통합 서비스를 제공하고 새로운 비즈니스 시스템과의 통합을 위해 D-Mediator(Data-Mediator)와 C-Mediator(Control-Mediator)와 같은 중재자를 설계 및 구축하였다. 이를 통해 이질적인 웹 서비스들간의 통합을 자동화하여 기존에 비해 훨씬 빠르고, 유연하며, 효율적인 상호운용이 가능하게 하였다.



(그림 10) 파라미터 매칭 정보가 없는 서비스 실행 결과화면



(그림 11) 파라미터 매칭 정보가 있는 입력 파라미터 입력화면

5. 결 론

현재의 웹 서비스는 개발자에 의한 단편적인 서비스 연결 및 서비스에 대한 사용은 가능하지만 서비스에 대한 상호 작용이나 복합적인 해석이 불가능하며, 일반인들에 의한 해석이나 이용 또한 거의 불가능하다. DAML-S/OWL-S, WSMF 및 BPELWS 등의 웹 서비스 기술방식은 웹 서비스의 기능적(functionality) 측면만을 표현하고 있기 때문에, 웹 서비스의 개념적, 의미적 기능을 표현하기에는 한계가 있다. 따라서 이들 표현 방식으로는 웹 서비스에서 가장 중요한 의미적인 웹 서비스 발견, 동적 웹 서비스 합성등이 불가능하기 때문에 시맨틱 웹의 궁극적인 목표를 실현할 수가 없다. 본 논문에서는 웹 서비스의 기능적 측면과 의미적 측면을 통합적으로 표현하기 위해서 수행 온톨로지를 설계하여 웹 서비스를 기술하기 위한 새로운 방법을 제시하였고, 이를 토대로 웹 서비스 합성 시스템을 구축하여 수행 온톨로지의 유효성을 검증하였다. 또한 수행 온톨로지를 적용한 웹 서비스 합성 시스템은 웹 서비스간의 의미적 상호운용을 지원함으로써 자연스럽게 기업 내외부의 이질적인 어플리케이션간의 통합 서비스를 제공하며, 새로운 비즈니스 파트너간의 시스템과의 통합도 자동적으로 이루어지게 한다. 뿐만 아니라 소프트웨어 통합을 자동화함으로써 기존에 비해 훨씬 빠르고, 유연하며, 효율적으로 웹 서비스간의 상호운용을 가능하게 하였다.

참 고 문 헌

- [1] McIlraith, Sheila, "Semantic Enabled Web Services," XML-Web Services ONE Conference, June, 2002.
- [2] Tim Berners-Lee, James Hendler, Ora Lassila, "The Semantic Web", Scientific American, May, 2001.
- [3] McGuinness, D., Fikes, R., Hendler, J. and Stein, L., "DAML+OIL : an ontology language for the Semantic Web," *IEEE Intelligent Systems*, Vol.17, No.5, pp.72-80, 2002.
- [4] Roman Dumitru, "Semantic Web Services Composition using SHOP2 in the Open Agent Architecture," Technical University of Cluj Napoca, 2003.
- [5] Microsoft MSDN, XML Web Service, <http://msdn.microsoft.com/library/default.asp?url=/nhp/Default.asp?contentid=28000442> .
- [6] Sanjiva Weerawarana, Francisco (Paco) Curbera, "Business Process with BPELWS", <http://www903.ibm.com/developerworks/kr/preut.jsp?url=http://www106.ibm.com/developerworks/webservices/library/wsbpelcol2/&origin=ws>, 2002.

- [7] Denker, G., Kagal, L., "Security annotation for DAML Web Services," 2003.
- [8] The DAML Service Coalition, "DAML-S: Semantic Markup for Web Services," <http://www.daml.org/services/damls/0.9/>, 2003.
- [9] Fensel, D. Bussler C., "The Web Service Modeling Framework. First International Semantic Web Conference," Sardinia, Italy, June, 2002.
- [10] Darwin, "The Essential Guide to Web Services," <http://www.darwinmag.com/read/010102/essential.html>, 2002.
- [11] W3C, Web Service Specification, <http://www.w3.org/2002/ws>.
- [12] W3C, Web Service Description Language Specification, <http://www.w3.org/TR/wsdl>.



오 지 훈

e-mail : opt@wonkwang.ac.kr
 2002년 원광대학교 컴퓨터및정보통신
 공학부(공학사)
 2004년 원광대학교 컴퓨터공학과
 (공학석사)
 1999년~현재 원광대학교 시맨틱 웹 연구실
 연구원

2004년~현재 (주)유엠텍 종합기술연구소 선임연구원
 관심분야 : 온톨로지, 시맨틱 웹 서비스, 지능형 e-비즈니스



최 병 식

e-mail : bschoi@kunjang.ac.kr
 1981년 조선대학교 과학교육과(교육학사)
 1987년 고려대학교 과학교육과
 (교육학석사)
 2004년 원광대학교 컴퓨터공학과
 (공학박사)

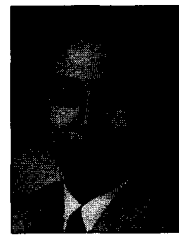
1996년~현재 군장대학교 컴퓨터응용학부 교수
 관심분야 : 온톨로지, 웹 서비스, 차세대 인터넷 기술



정 영 식

e-mail : ysjeong@wonkwang.ac.kr
 1989년 고려대학교 전산학 석사
 1993년 고려대학교 전산학 박사
 1997년~1998년 미시간 주립대학교
 전산학과 객원교수
 1993년~현재 원광대학교 컴퓨터및정보
 통신공학부 교수

관심분야 : 분산병렬처리, 그리드컴퓨팅, LBS



주 수 종

e-mail : scjoo@wonkwang.ac.kr
 1986년 원광대학교 컴퓨터공학과(공학사)
 1988년 중앙대학교 컴퓨터공학과 System
 S/W(공학석사)
 1992년 중앙대학교 컴퓨터공학과 System
 S/W(공학박사)

1993년~1994년 University of Massachusetts at Amherst,
 Dept of EECS 연구교수
 2002년~2004년 UC at Irvine, Dept of EECS 연구교수
 1990년~현재 원광대학교 전기전자 및 정보공학부 교수
 1996년~현재 ISO/IEC JTCl/SC22 - Korea 전문위원
 관심분야 : 멀티미디어DB, 분산 실시간 컴퓨팅, 분산 객체 모델



한 성 국

e-mail : skhan@wonkwang.ac.kr
 1979년 인하대학교 전자공학과(공학사)
 1983년 인하대학교 전자공학과(공학석사)
 1988년 인하대학교 전자공학과(공학박사)
 1990년~1992년 University of
 Pennsylvania 방문교수

2003년~2004년 University of Innsbruck와 DERI 연구교수
 1984년~현재 원광대학교 전기전자 및 정보공학부 교수
 관심분야 : 온톨로지, 시맨틱 웹 서비스, 지식표현 및 추론,
 자연언어처리