

시맨틱 웹 언어를 이용한 서비스 명세 (Service Description) 확장

Extension of Service Description in a Semantic Web Language

김홍기(Hong-Gee Kim)*, 김학래(Hak Lae Kim)*, 이강찬(Kang Chan Lee)**,
전종홍(Jong Hong Jeon)**

초 록

시맨틱 웹 기반의 서비스의 기술(혹은 명세)은 지능적 서비스 중개(matchmaking)가 가능할 수 있도록 언어적 표현의 유연성(flexibility)과 표현력(expressiveness)을 높일 수 있는 방향으로 진행된다. 유연성은 서비스 기술의 상세한 정도와 표현의 구조적 차이를 컴퓨터가 처리하는 방식과 관련이 있고, 표현력은 서비스를 의미적 차원으로(혹은 온톨로지에서 표현하는 개념의 수준으로) 보다 풍부하게 표현하는 것을 의미한다. 최근 전자상거래의 표준으로 자리잡고 있는 UDDI와 ebXML의 단점은 서비스 명세에 있어서 유연성과 표현력이 떨어진다는 점이다. 본 연구에서는 DAML-S를 중심으로 시맨틱 웹 기반의 서비스 명세의 확장에 필요한 언어적 구성 요소들을 분석한다. 그리고 이를 지능형 웹 서비스 구현에 적용시키기 위한 방법을 가상적 경매 사이트인 테스트옥션 닷컴(testAuction.com)을 구현한다. 마지막으로 현재 온톨로지를 이용한 서비스 기술 프레임워크에 대한 이론적 고려사항들에 대하여도 살펴본다.

ABSTRACT

A Semantic Web based service description is to enable intelligent matchmaking of services by improving representational flexibility and expressiveness in service description for Web Services. Flexibility concerns the way of how a computer handles the discrepancies in the granularity and the structure of service description between different participants of web services. Expressiveness means richness in the semantic description of services (or ontological representation of each service as a concept). The main shortcoming of currently available industry standard framework for e-commerce such as UDDI and ebXML is that they do not allow much flexibility and expressiveness in the service description. In our research we analyze the requirements of extending service description based on the Semantic Web, especially in the context of DAML-S. We also propose a method to be applied to implement intelligent web services by implementing a virtual auction site, testAuction.com. Finally, we consider some theoretical issues regarding the framework for web services using ontology.

키워드 : 시맨틱 웹, 웹 서비스 서비스 명세, 온톨로지

Semantic Web, Web Services, Service Description, Ontology

이 연구는 2003학년도 단국대학교 대학연구비의 지원으로 연구되었음.

* 단국대학교 경상학부

** 한국전자통신연구원 표준화센터 인터넷응용기술연구팀

1. 서 론

오늘날 인터넷을 통한 전자상거래는 상거래를 위한 시간, 공간, 프로세스의 개념을 바꾸어 놓았다. 상거래의 속도는 빛의 속도로 빨라지고, 시장의 참여자는 공간을 뛰어넘은 관계를 형성할 수 있으며, 비즈니스 프로세스는 자동화 됨으로써 더욱 복잡하고 다양하게 되었다. 최근 웹 서비스(Web Services)는 전자상거래의 새로운 장을 열게 해 줄 기술로 주목을 받고 있다. 즉, 웹 서비스는 비즈니스 서비스 제공자와 사용자를 연결시켜주고, 비즈니스 파트너들 사이에 서비스를 공유하게 하고, 더 나아가서 여러 서비스들이 보다 복잡한 새로운 서비스로 결합될 수 있는 방법을 제공해 준다. 웹 서비스 환경에서는 서비스의 제공자와 요청자 사이에 공개적인 정보 활용이 가능하기 때문에 업무의 효율성 뿐만 아니라 비용 측면에서도 만족도를 높일 수 있는 가능성을 제시하고 있다.

웹 서비스의 기능적 구현을 위해 많은 노력이 현재까지 진행되었고 이를 기반으로 하여 SOAP(Simple Object Access Protocol)[15], WSDL(Web Services Description Language)[5,9], UDDI(Universal Description, Discovery and Integration)[10]와 같은 웹 서비스 표준이 권고되었다. 그러나 제안된 표준들이 서비스의 탐색(discovery), 합성(composition) 또는 관리(management)와 같은 다양한 요구 조건을 처리하기에는 여전히 많은 한계가 있다. 특히 WSDL이나 UDDI는 구문적 수준에서 서비스의 기술과 탐색을 위한 기능만을 제공하고 있기 때문에 콘텐츠에

포함된 의미를 처리하는데 한계가 있다.

컨텐츠의 의미(semantics)를 처리하기 위한 시도는 학계와 W3C[7]를 중심으로 진행되고 있으며 특히 시맨틱 웹 환경에서 자동화된 탐색, 합성과 같은 기술을 제공할 수 있는데 초점이 맞추어 지고 있다. 시맨틱 웹 서비스(Semantic Web Services : SWS)는 시맨틱 웹과 웹 서비스가 갖고 있는 특징을 집합시킨 것이라 정의할 수 있다. 시맨틱 웹 기반의 웹 서비스에서의 서비스 명세(혹은 기술, description)는 시맨틱 마크업 언어를 사용하여 콘텐츠를 표현함으로써 서비스의 탐색, 요청, 선택, 합성과 같은 작업을 자동적으로 처리할 수 있게 한다.

시맨틱 웹 서비스와 관련된 연구는 미국의 DARPA[8] 중심으로 진행중인 DAML-S[1,2]와 EU를 중심으로 연구중인 WSMF[11]가 대표적이라 할 수 있다. 최근 연구의 초점은 웹 서비스의 프레임워크를 확장하여 의미적으로 웹 서비스를 기술하기 위한 온톨로지 마크업 언어의 개발과 에이전트 기술을 서비스 중개(matchmaking)에 활용하는 방안에 맞추어져 있다. 이러한 연구에 있어서 최근까지 가장 성공적인 기술이 DARPA를 중심으로 진행중인 DAML+ OIL의 서비스 온톨로지인 DAML-S(DAML+OIL Services)이다.

본 연구에서는 DAML-S를 중심으로 시맨틱 웹 기반의 서비스 기술의 확장에 필요한 언어적 구성 요소들을 분석하고, 이를 지능형 웹 서비스 구현에 적용시키기 위한 방법을 가상적 경매사이트(testAuction.com)의 웹 서비스에 대한 구현 사례를 통하여 제안 할 것이다. 또한 현재 온톨로지를 이용한 서비스 기

솔 프레임워크에 대한 이론적 고려사항들에 대하여도 살펴볼 것이다.

2. 서비스 명세 언어의 요구사항

서비스 지향 아키텍처(Service-Oriented Architecture)[22]에서 서비스 명세는 서비스의 등록(publish/Update), 탐색(Discover), 연결(bind/Invoke) 오퍼레이션과 연관되어 있다. 서비스 제공자는 서비스 명세를 사용하여 해당 웹 서비스를 서비스 레지스트리에 등록하고, 서비스 요청자는 서비스 레지스트리에 있는 서비스 명세를 탐색하여 사용 가능한 서비스의 형태를 얻는다. 이러한 기능을 수행하기 위해 서비스 명세서는 서비스에 대한 기본적인 특성들을 정의할 수 있어야 하며 공식적인 메커니즘을 이용하여 서비스를 기술해야 한다. W3C에서 제안한 웹 서비스 명세 스택[23]은 서비스 명세를 기능성 명세와 비기능성 명세로 구분하고 있다. 기능성 영역은 메시지의 구문 형태, 프로토콜, 인코딩 형태등을 기술하는 반면 비기능성 명세는 정보 교환을 위한 용어, 보안, 인증, 지적 재산권 등과 같은 비즈니스 프로세스와 관련된 내용을 기술한다. 지금까지 제안된 명세 표준에서 WSDL은 기능성 명세로, WSEL[29], XLANG[23], WSFL[28], BPEL4WS[27], WSCI[25] 등은 비기능적 명세로 구분된다.

현재 정의된 웹 서비스 프레임워크에서는 명세를 기술하기 위해 WSDL을 사용한

다. WSDL은 웹 서비스를 사용하기 위한 기술적 구문을 나타내기 위해 제안된 표준으로 크게 서비스가 수행하는 일, 접근 방법, 네트워크상의 위치 정보를 포함하고 있다. 그러나 WSDL에 정의된 명세 정보는 의미적 수준에서 서비스 정보를 기술하는데 한계가 있다. 예를 들어 서비스 요청자가 세금에 관련된 웹 서비스를 이용하는 경우를 생각해 보자. 서비스 요청자는 세금을 계산하고 처리하기 위해 요구되는 여러 가지 절차를 웹 서비스를 이용하여 해결하고자 한다. 이때 사용자는 세금을 계산하기 위해 급여와 이름을 입력값으로 제공한다.

■ 빈약한 의미 정의

사용자에 의해 제공된 정보를 기반으로 웹 서비스는 스트링 형식의 이름과 정수 형태의 세금액, 세율에 대한 결과값을 돌려준다. 그러나 금액에 대한 단위가 명확하지 않을 경우에는 문제가 발생한다. 예를 들어 통화 단위가 원화 또는 달러일 경우 서비스를 이용하여 얻는 결과는 상이할 수 밖에 없다. 현재의 웹 서비스 환경에서는 이와 같은 문제는 서비스를 사기 전에 사용자가 확인해야 한다.

■ 상이한 의미 정의

사용자가 두 개의 서비스에서 동일한 결과값을 얻었을 때, 서비스 명세에서 결과값을 각각 판매세와 소득세로 정의하고 있다면 사용자는 결과값에 대해 혼동할 수 밖에 없다. 즉 결과값은 동일하지만 정의하고 있는 의미가 상이할 경우, 이를 해결하는데 어려움이 있다.

▪ 상이한 구문 정의

두 개의 서비스가 세율을 기술하기 위해 A 서비스는 퍼센트(예: 25%)로 기술하고, B 서비스는 백분율(0.025)로 정의할 수 있다. 이 때 각각의 서비스에서 정의한 세율은 같은 의미를 갖고 있지만 서로 상호 운용되기 어렵다.

현재의 웹 서비스 명세 언어는 풍부한 의미를 표현하는데 제약이 있다. 서비스 명세 언어에서 의미적 수준의 표현력을 제공하지 못하면 에이전트나 프로그램에 의한 자동화는 낮아질 수 밖에 없고 이는 자연스럽게 서비스를 이용하는 사용자의 수동적 처리를 증가시키게 된다. 콘텐츠의 의미적 정보를 기술하기 위해 제안된 언어에는 RDF[17], DAML+OIL, OWL[18] 등이 있다. 이들 언어는 에이전트가 기술된 명세를 바탕으로 서비스를 자동적으로 발견하고 호출할 수 있는 기능을 제공하고 있다. 특히 DAML-S는 DAML+OIL 온톨로지 언어를 기반으로 하여 서비스 프로세스를 기술할 수 있는 언어로 개발되었고 WSDL의 표현력을 의미적 수준으로 확장할 수 있는 가장 성공적인 대안으로 평가 받고 있다. 다음 절에서는 DAML-S의 일반적 특징을 살펴 보고 이를 이용한 서비스 프로세스 온톨로지 표현의 언어적 구성요소들을 분석한다.

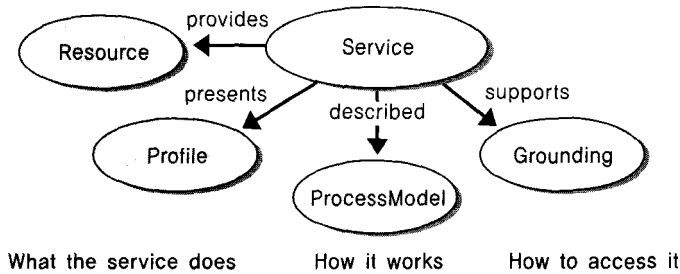
3. DAML-S를 이용한 의미적 서비스 명세

3.1 개요

DAML-S는 DAML+OIL의 데이터 모델을 상속 받아 서비스 명세 항목을 표현하고 항목들 간의 관계성을 기술함으로써 시맨틱 웹 기반의 웹 서비스를 구현할 수 있는 필수 요소로 언급되고 있다. DAML-S는 DAML 기반의 웹 서비스 온톨로지 컴퓨터가 처리 가능한 형태로 웹 서비스를 표현하고, 서비스 특성들을 기술하기 위한 마크업 언어 요소를 포함하여 서비스 제공자가 서비스를 명확하게 기술할 수 있도록 한다. DAML-S는 자동화된 웹 서비스의 탐색(discovery), 실행(execution), 합성(composition), 상호운용(interoperation)을 포함하는 서비스 작업의 자동화를 지원할 수 있다. DAML-S는 현재 0.9 베타 버전이 제안되었으며, 웹 온톨로지 언어인 OWL로의 전환을 위해 OWL 기반의 서비스 언어(OWL-S)도 동시에 제공하고 있다.

3.2 DAML-S의 상위 온톨로지 구조

DAML-S는 시맨틱 웹 서비스를 구현하기 위한 핵심적인 컴포넌트로 DARPA에 의해 개발된 DAML+OIL 기반의 서비스 기술을 위한 온톨로지 언어이다[4]. DAML-S는 SOAP(Simple Object Access Protocol), WSDL(Web Service Description Language), WSFL(Web Service Flow Language), BPEL4WS(Business Process Execution for Web



〈그림 1〉 DAML-S의 상위 온톨로지[2]

Services)[19]등과 같은 산업계 표준들의 상위 수준에 구축되어 웹 서비스의 탐색(discovery), 요청(invocation), 상호운용(interoperation), 합성(composition), 검증(verification), 통제(monitoring)등을 담당할 수 있다.

DAML-S의 온톨로지는 〈그림 1〉에서 보여 주듯이, presents, describedBy, supports의 세 가지 특성을 선언하고 있으며 이는 Service Profile, Process Model, Grounding의 세 가지 클래스로 구성된다.

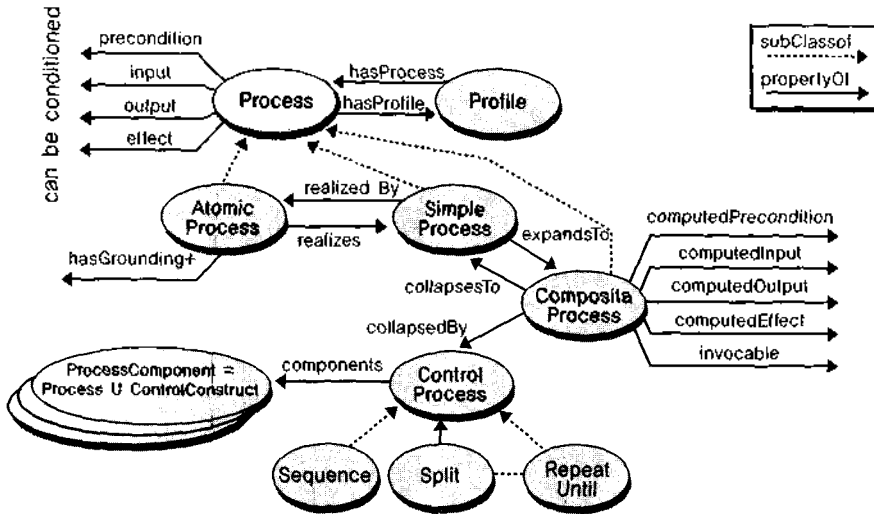
3.2.1 DAML-S 서비스 프로파일

서비스 프로파일은 서비스의 제공 목적에 대한 추상적 수준의 개요 형태의 정보를 포함하고 있다. DAML-S의 프로파일은 서비스의 제공자에 대한 정보(what organization provided the service), 서비스를 처리하기 위한 기능에 대한 정보(what function the service computes), 서비스의 특성을 구체화한 특성에 대한 정보 등과 같은 세가지 기능의 서비스를 기술하고 있다. 서비스 제공자의 정보는 서비스를 제공하는 조직의 정보를 제공하는 연락처와 관련된 정보를 포함한다. 기능적 기술(functional description)은 서비스에 의해 처리되는 변환

내용을 담고 있다. 또한 프로파일은 서비스의 특성을 기술하기 위해 사용되는 특성의 명세를 포함할 수 있다. 이러한 명세에는 제공된 서비스의 카테고리화 관련된 정보를 구체화하는 형식과 서비스의 등급을 구체화하는 형식으로 구분할 수 있다. 예를 들어 카테고리화 관련된 정보는 UNSPSC[20]와 NAICS[21]와 같은 분류 시스템(classification systems)을 포함할 수 있다. 프로파일에는 정보의 형식을 기술하기 위해 최대 응답 시간, 지리적 범위와 같은 범위를 매개변수(parameter)로 제공하고 있다.

3.2.2 DAML-S 프로세스 모델

프로세스 모델은 프로세스와 프로세스 제어 모델의 두 가지 컴포넌트로 구성되며 이를 각각 프로세스 온톨로지(Process Ontology)와 프로세스 제어 온톨로지(Process Control Ontology)라고 한다. 입력(inputs), 출력(outputs), 전제조건(preconditions), 결과(effects)에 기반으로 기술되는 프로세스는 에이전트가 서비스의 상호 운용, 합성을 가능하게 한다. 프로세스 제어 모델은 에이전트가 서비스 요청에 대한 실행을 모니터링하는 기



〈그림 2〉 프로세스 온톨로지[2]

능을 담당한다. 프로세스 온톨로지는 현재 DAML-S에 정의되어 있지만 프로세스 제어 온톨로지에 관련된 내용은 정의되어 있지 않다.

프로세스 모델은 〈그림 2〉과 같이 Atomic Process, Simple Process, Composite Process 로 구분된다. Atomic Process는 서비스 요청의 최소 단위로 서비스 요청을 단일 과정으로 구분하고 프로세스를 실행한다.

Simple 프로세스는 단일 작업 단위를 실행하는 면에서 Atomic 프로세스와 유사한 반면 서비스 요청자가 직접 서비스를 호출할 수 없으며 Grounding 클래스와 연결되지 않는다. Simple 프로세스는 Atomic 프로세스와 Composite 프로세스의 추상적 관점을 제공한다. Simple 프로세스는 realizedBy 속성을 이용하여 Atomic 프로세스 기술될 수 있고, expandsTo 속성으로 Composite 프로세스로 확장할 수 있다.

마지막으로 Composite 프로세스는 DAML-S의 제어 구조를 사용하여 다양한 Atomic/Simple/Composite 프로세스 내에 기술된 프로세스를 재구성할 수 있다. 프로세스의 재구성은 SEQUENCE, IF-THEN-ELSE 등과 같은 제어 구성자(control constructs)를 이용하여 구체화할 수 있다.

4. WSDL과 DAML-S의 연계 방안

4.1 WSDL과 DAML-S의 연계 필요성

WSDL은 특정 웹 서비스의 방법과 프로토콜, 데이터 형식들을 더욱 상세하게 정의하여 매우 효과적으로 인터페이스 규약에 맞추어 서비스를 호출할 수 있다. WSDL은 실질적 (de facto) 산업 표준으로 자리잡고 있다. 또

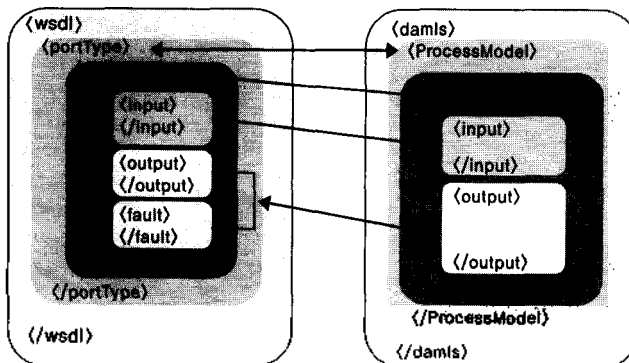
한 UDDI 레지스트리에서 호환되는 서비스들의 동적인 검색과 바인딩을 지원한다. 그러나 WSDL에 기술되는 자연어 기반의 명세는 단지 개발자만이 이해 가능한 형태이기 때문에 프로그램이 자동적으로 처리 가능한 형태의 명세와 의미적으로 기술된 형식 언어 기반의 명세 언어가 요구된다. 이러한 명세 언어는 웹 서비스에서 사용되는 프로세스의 통합을 용이하게 함으로써 궁극적으로 소요 비용을 절감할 수 있다. DAML-S는 컴퓨터가 처리 가능하고 명확하게 기능을 제공하기 위한 추상적 정보를 제공한다. 다음절에서는 DAML-S와 WSDL이 연계되는 구체적인 방법을 살펴 본다.

4.2 연계 방안

DAML-S는 서비스를 구성하고 있는 프로토콜, 메시지 형식과 같은 정보에 접근하기 위한 구체화된 정보를 기술하고 있다. 일반적으로 그라운드(grounding)는 통신 프로토콜, 서비스에 접근하기 위한 사용되는 포트와 관련된 정보를 구체화하기 위해 사용한다.

DAML-S에서 구체화는 그라운드 클래스를 이용하여 정의할 수 있다. 그라운드 클래스는 서비스 명세에 포함된 엘리먼트들을 정의하기 위한 과정으로 WSDL의 엘리먼트와 연결될 수 있다. DAML-S와 독립적으로 개발된 WSDL은 추상적 사양과 구체적인 오퍼레이션이 분리된 점을 활용하여 서비스 인터페이스 정의와 서비스 구현을 분리하여 반영하고 있다. 두 언어는 서비스의 입출력을 기술하기 위해 사용되는 추상 타입에 있어 다음과 같은 공통적인 특성을 갖고 있다.

- DAML-S의 Atomic process는 WSDL의 오퍼레이션(operation)과 일치한다.
- DAML-S의 Atomic process의 출력(outputs)의 집합과 입력(inputs)의 집합은 WSDL의 메시지(message)의 개념과 일치한다. <그림 3>는 DAML-S와 WSDL의 공통 특성을 비교하고 오퍼레이션의 변환을 위한 상호 관계를 도식화 하고 있다.
- DAML-S의 Atomic process의 입력과 출력의 타입은 WSDL의 확장 개념인 추상 타입(abstract type)과 일치한다.



<그림 3> DAML-S와 WSDL의 그라운드

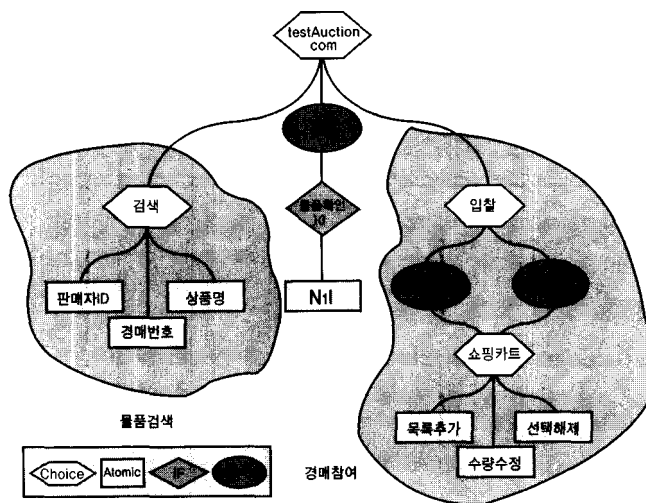
5. DAML-S를 이용한 서비스 기술 확장의 구현

본고에서는 DAML-S를 이용한 시맨틱 웹 기반 서비스 명세를 확장하는 방법을 검증하기 위하여 가상의 경매사이트인 테스트옥션닷컴(testAuction.com)을 구현하였다. 우선 일반적인 웹 서비스 명세를 WSDL로 표현한 다음 의미적 수준의 서비스 명세를 확장하기 위하여 DAML-S로 구체화하였다. 5.1에서는 테스트옥션닷컴의 프로세스 모델을 설명하고 WSDL과 DAML-S 사이의 매핑을 살펴본다. 5.2에서는 의미적 수준에서 검색을 하고 이를 기반으로 사용자가 입찰할 수 있는 시나리오를 만들어 이를 구현한다.

5.1 테스트옥션닷컴의 프로세스 모델

위에 제시된 DAML-S와 WSDL의 공통점을 기반으로 가상의 경매 사이트(testAuction.com)에 대한 서비스 명세를 확장하는 방법을 살펴본다. 테스트옥션닷컴은 고객이 웹 서비스를 이용해 구입을 원하는 물품을 경매에 참여하여 쇼핑카트에 채우고 물품 목록을 탐색할 수 있도록 서비스를 공개한다. <그림 4>와 같이 테스트 옥션의 웹 서비스는 다양한 검색 가능(판매자 ID, 물품 ID, 상품명)을 제공하고, 쇼핑카트의 관리는 목록의 추가, 수정 또는 삭제 기능을 포함하고 있다.

테스트 옥션 웹 서비스의 WSDL 명세는 20개의 오퍼레이션, 60개의 포트타입, 40개의 입/출력 메시지로 구성되어 있다. 시맨틱 웹 서비스를 위한 DAML-S의 생성을 위해 WSDL2DAMLS[3,14]를 사용한다. WSDL2DAMLS는 WSDL를 DAML-S로 자



<그림 4> 테스트 옥션닷컴의 프로세스 모델


```
<operation name="SellerIDSearchRequest">
  <input message="ta: SellerIDSearchRequest"/>
  <output message="ta: SellerIDSearchResponse"/>
</operation>
```

〈리스트 1〉 테스트옥션 닷컴의 WSDL

동적으로 변환시켜 주는 역할을 한다. 변환 과정은 크게 XSD 타입을 DAML의 개념과 매핑하기 위한 과정과 WSDL의 오퍼레이션을 DAML-S의 프로세스로 변환하는 과정으로 구분할 수 있다. 이 과정을 통해 4개의 DAML-S 파일(profile.daml, process.daml, grounding.daml, concept.daml 등)을 생성할 수 있다.

〈리스트 1〉은 테스트옥션 닷컴에서 판매자 ID를 기반으로 검색하기 위해 정의한 WSDL의 일부분이다.

WSDL에 정의된 오퍼레이션은 〈그림 5〉와 같이 DAML-S의 그라운드링 파일로 변환된다. WSDL을 DAML-S의 그라운드링 클래스와 매핑할 때, 단일 프로세스는 Atomic Process로 정의(생성)된다.

WSDL2DAMLS를 이용한 변환은 DAML-S의 그라운드링 클래스를 자동적으로 생성하여 WSDL과의 상호 운용을 제공할 수 있다. 그러나 WSDL이 프로세스를 재구성할 수 있는 정보를 제공하고 있지 않기 때문에 DAML-S의 프로세스 모델과 프로파일은 자동적으로

```
<grounding:hasAtomicProcessGrounding
rdf:resource="#testAuctionPort_SellerIDSearchRequest"/>
Atomic 프로세스 생성

<grounding:WsdAtomicProcessGrounding
rdf:ID="WSDLGrounding_testAuctionPort_SellerIDSearchRequest">
  <grounding:damlProcess rdf:resource="#&pm_file;#testAuctionPort_SellerIDSearchRequest"/>
  <grounding:wsdlOperation>
    <xsd:uriReference rdf:value="/usr1/jakarta/tomcat-4/webapps/v2ds/tmpfiles/wsd/SellerIDSearchRequest" />
  </grounding:wsdlOperation>
  operation

  <grounding:wsdlInputMessage>
    <xsd:uriReference rdf:value="http://www.testauction.com/#SellerIDSearchRequest" />
  </grounding:wsdlInputMessage>
  Inputs

  <grounding:wsdlOutputMessage>
    <xsd:uriReference rdf:value="http://www.testauction.com/#SellerIDSearchResponse" />
  </grounding:wsdlOutputMessage>
  Outputs

</grounding:WsdAtomicProcessGrounding>
```

〈그림 5〉 DAML-S의 그라운드링 클래스

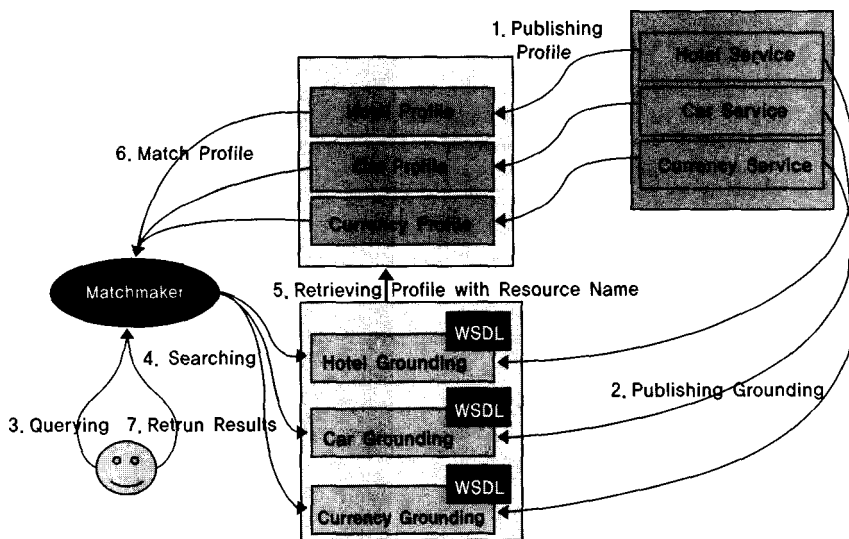
생성할 수 없다. 따라서 DAML-S의 프로세스 모델과 프로파일 클래스의 명세를 완성하기 위한 추가적인 작업이 요구된다.

5.2 테스트옥션 닷컴의 의미 기반 검색

DAML-S 기반의 웹 서비스는 에이전트가 서비스를 자동적으로 탐색하고 합성할 수 있게 한다. 예를 들어, 사용자가 테스트옥션 닷컴(testAuction.com)에서 여행 상품을 입찰하는 경우를 생각해 보자. 여행 상품의 경우 여행 목적지에 대한 정보 뿐만 아니라 교통 수단, 호텔 정보, 환율 정보와 같은 다양한 서비스들을 함께 고려해야 한다. 따라서 사용자가 특정 여행 상품을 찾고 입찰을 해도 위에 열거한 다른 정보와 관련된 정보를 찾아 입찰할 수 없다면 만족하는 결과를 얻을 수 없다. DAML-S를 이용하여 서비스를 기술하고 이

를 기반으로 하는 시맨틱 웹 서비스 환경에서는 이러한 문제를 해결하기 위해 에이전트 기술을 사용한다. <그림 6>은 사용자가 여행 상품을 검색하는 과정을 도식화한 것이다. 사용자는 자신이 원하는 조건을 입력하고 이에 따른 결과를 의미적 수준에서의 매칭 알고리즘(matching algorithm)을 이용하여 얻을 수 있다.

서비스 제공자는 여행 상품, 호텔, 자동차, 환율에 대한 정보를 DAML-S를 기반으로 제공한다. 호텔 정보에 대한 서비스의 경우, 서비스 제공자는 호텔의 예약하기 위해 요구되는 가격과 날짜에 대한 정보를 제공한다. [리스트 2]은 서비스를 위해 기술된 프로파일이며, 서비스에 의해 요구되는 입력값은 개념(concepts) 온톨로지에 정의된 가격(price)의 인스턴스로 제약된다. 반면 출력값은 숙박 시설을 정의한 온톨로지서 호텔의 인스턴스로 생성된다.



<그림 6> 시맨틱 웹 기반의 서비스 검색

```

<profile:Profile rdf:ID="HotelService">
  <profile:serviceName>HotelService</profile:serviceName>
  <profile:providedBy>.</profile:providedBy>
  <input>
    <profile:ParameterDescription rdf:ID="Input_Price">
      <profile:parameterName>Price</profile:parameterName>
      </profile:ParameterDescription>
    </input>
    <output>
      <profile:ParameterDescription rdf:ID="Output_Hotel">
        <profile:parameterName>Hotel</profile:parameterName>
        <profile:restrictedTo rdf:resource="accommodation.daml#hotel"/>
      </profile:ParameterDescription>
    </output>
  </profile:Profile>

```

<리스트 2> 호텔 서비스의 명세(프로파일)

서비스의 요청은 배포된 서비스 명세와 같은 형식으로 표현된다. 예를 들어 호텔을 찾을 때, 서비스는 가격의 인스턴스에 일치하는 입력값의 형태를 포함하며 호텔의 인스턴스로 출력값을 생성하게 된다. <리스트 3>는 호텔 서비스 요청의 예를 표현한 것이다. 배포된 서비스 명세와 요청된 서비스 사이의 일치

는 입력과 출력값 사이의 비교를 통해 이루어진다. 예를 들어 두 개의 서비스에서 같은 개념으로 입력값을 정의하고 있다면 일치된 결과로 인식할 수 있다. 또한 호텔에 대한 정보를 정의한 온톨로지에서 Hotel을 Accommodation의 서브클래스로 정의하고 있어도 추론이 가능하다.

```

<profile:Profile rdf:ID="RequestHotelService">
  <input>
    <profile:ParameterDescription rdf:ID="Input_Price">
      <profile:parameterName>Price</profile:parameterName>
      <profile:restrictedTo rdf:resource="content.daml#Price">
      </profile:ParameterDescription>
    </input>
    <output>
      <profile:ParameterDescription rdf:ID="Output_SHotel">
        <profile:parameterName>Hotel</profile:parameterName>
        <profile:restrictedTo rdf:resource="lodging.daml#accommodation#hotel"/>
      </profile:ParameterDescription>
    </output>
  </profile:Profile>

```

<리스트 3> 호텔 서비스의 요청 예제

〈그림 6〉에서 중개자는 사용자가 입력한 정보에 맞는 서비스를 WSDL과 DAML-S를 기반으로 검색한다. DAML-S 그라운드 클래스에서 검색된 서비스는 매칭 알고리즘을 이용하여 서비스 프로파일과 프로세스 모델에 정의된 온톨로지를 검색하여 결과를 돌려준다. 일반적으로 의미적 수준의 매칭을 할 때는 〈그림 7〉과 같은 서비스 기능성(service functionalities)과 기능적 속성(functional attributes)을 고려하게 된다. 서비스 기능성은 배포된 서비스가 요구된 서비스와 일치하는지 여부를 판단하는데 사용된다. 서비스 제공자에 따라 서비스는 같은 기능을 가졌어도 다른 방법으로 기술될 수 있다. 예를 들어 전제조건(precondition) 대신 입력(input)을, 효과(effect) 대신 결과(output) 속성을 이용하여 정의할 수 있다. 이런 이유 때문에 서비스 기능성은 입력 집합(input, precondition, accesscondition 등)과 출력 집합(output, conditionaloutput, effect 등)

으로 구분한다. 에이전트는 사용자의 입력 값을 이용하여 입력 집합에서 일치하는 서비스를 찾고, 해당 서비스의 예상되는 결과에 대한 값은 출력 집합을 이용하여 탐색한다. 기능적 속성은 요구된 서비스에 대한 적절한 타입이나 품질의 여부를 지원하는지 판단할 수 있다. 기능적 속성은 추론에 사용되는 상이한 온톨로지나 서비스 형식을 기술하고 있다.

테스트옥션 닷컴에서 중개자(matchmaker)의 역할은 요청된 서비스를 DAML-S의 그라운드 클래스에서 검색하고 이를 기반으로 하여 서비스 제공자의 기능적 속성을 정의한 프로파일에서 추론하는 역할을 한다. 그러나 실제 웹 서비스 프레임워크에서 의미적 수준에서 검색을 지원하기 위해서는 UDDI와 연결된 보다 향상된 중개자와 매칭 알고리즘이 필요하다.

<i>Description</i>	<i>Service Functionalities</i>	<i>Functional Attributes</i>
<i>service Name</i>	<i>parameter</i>	<i>geographicRadius</i>
<i>intendedPurpose</i>	<i>input</i>	<i>degreeOfQuality</i>
<i>textDescription</i>	<i>output</i>	<i>serviceType</i>
<i>role</i>	<i>conditionaloutput</i>	<i>erviceCategory</i>
<i>requestedBy</i>	<i>precondition</i>	<i>erviceParameter</i>
<i>providedBy</i>	<i>accesscondition</i>	<i>communicationThru</i>
	<i>effect</i>	<i>qualityRating</i>
	<i>domainResource</i>	<i>qualityGuarantees</i>

〈그림 7〉 서비스 프로파일의 특성[29]

6. DAML-S의 한계 및 관련 이슈

위에서 자세히 살펴보았듯이 DAML-S는 온톨로지 기반의 유연하고 표현력이 높은 서비스 명세를 위한 언어이다. 하지만 실제 시맨틱 웹 기반의 웹 서비스를 구현하기 위한 프레임워크로 자리잡기 위해서 고려되어야 할 사항들이 존재한다. 프로파일 클래스에서 서비스를 기술하기 위한 네 개의 속성(IOPEs : Inputs, Outputs, Preconditions, Effects 등)은 프로세스 클래스에서도 공통적으로 갖고 있다. 이들은 속성의 형태는 같지만 개념적으로 서로 다르게 처리된다. 또한 DAML-S를 이용한 서비스의 기술은 풍부한 표현력을 갖는 반면 서비스 제공자의 명세 범위에 대한 일반성(generality of the descriptions)에 관련된 문제를 포함하고 있다. 예를 들어 서비스 제공자가 검색 결과를 향상시키기 위해서 서비스 범위를 모호하게 기술하면 검색의 결과는 신뢰할 수 없게 된다. 이와 같은 문제는 중개자(matchmaker)를 통해 일부분 해결할 수 있지만 근본적인 해결을 위해 DAML-S의 개념 모델의 수정이 요구된다.

스를 기술할 수 있어야 한다. DAML-S는 DAML의 데이터 모델을 상속 받아 서비스 명세 항목을 표현하고 항목들 사이의 관계성을 기술함으로써 시맨틱 웹 서비스의 필수 요소로 언급되고 있다. 서비스 명세의 유연성과 표현력을 높이기 위해서는 의미적 차원의 명세 기능을 지원해야 한다. DAML-S의 그라운드 클래스는 WSDL과 연결되어 인터페이스와 바인딩을 정의할 수 있고 이를 통해 서비스 명세 항목의 의미적 기술이 가능하다. WSDL과 DAML-S의 연동은 지능형 웹 서비스 구현을 위한 기본적 단계라 할 수 있다. 그러나 지능형 웹 서비스를 구현하기 위해서 WSDL이 제공하지 못하는 프로세스 모델이나 규칙을 정의하기 위한 개념적 모델의 보완과 프레임워크의 개발이 요구된다.

7. 결 론

본 논문에서는 시맨틱 웹 기반의 서비스 기술 언어의 요구 사항을 구체적인 구현 사례를 통해 살펴 보았다. 지능형 웹 서비스의 구현에 적용하기 위한 서비스 기술 언어는 구문적 수준에서 뿐만 아니라 의미적 수준에서 서비

참 고 문 헌

- [1] Ankolekar, A., Burstein, M., Hobbs, J.R., Lassila, O., Martin, D.L., McIlrith, S.A., Narayanan, S., Paolucci, M., Payne, T.R., Sycara, K., and Zeng, H., "DAML-S: Semantic Markup for Web Services", 2003
- [2] DAML-S Coalition., "DAML-S: Web Service Description for the semantic web", In ISWC2002, 2002
- [3] Paolucci, M., Kawamura, T., Payne, T.R., Sycara, K., "Towards a Semantic Choreography of Web Services : from WSDL to DAML-S", 2003
- [4] Marta Sabou, Debbie Richards, and Sander van Splunter., "An experience report on using DAML-S", WWW 2003, May 20-24, 2003, Budapest, Hungary.
- [5] Web Services Description Language., <http://www.w3.org/TR/2002/WD-wsdl12-20020709/>
- [6] Andrew Layman, "Web Service ramework", <http://www.w3.org/2001/04/wsws-proceedings/ibm-ms-framework/framework.pdf>. Microsoft, 2001.
- [7] W3C, W3C 웹 사이트, <http://www.w3c.org/>
- [8] DARPA, DARPA 웹 사이트, <http://www.darpa.mil/>
- [9] E.Christensen, F. Curbera, G.Meredith, and S. Weerawarana., "Web Services Description Language(WSDL) 1.1", <http://www.w3.org/TR/wsdl>, 2001
- [10] UDDI, "The UDDI Technical White Paper". <http://www.uddi.org/>, 2001
- [11] D. Fensel., C. Bussler., "The Web Service Modeling Framework WSMF" , <http://www.swsi.org/resources/wsmf-paper.pdf>, 2002
- [12] Frank Leymann, "Web Services Flow Language (WSFL) Version 1.0", IBM Software Group, <http://www-3.ibm.com/software/solutions/webservices/pdf/WSFL.pdf>, 2001
- [13] PATRICK C.K.Hung, "Specifying Conflict of interest in Web Services Endpoint Language(WSEL)", ACM SIGecom Exchange Vol.3, No 3., August 2002, Pages 1-8
- [14] WSDL2DAMLS, WSDL2DAMLS Tool 웹 사이트, <http://www.daml.ricmu.edu/wsdl2damls/>
- [15] W3C, " Simple Object Access Protocol(SOAP) (version 1.1)", W3C Note 08, 2000
- [16] ebXML, ebXML 웹 사이트, <http://www.ebxml.org/>,
- [17] Dave Beckett, "RDF/XML Syntax Specification (Revised)", <http://www.w3.org/TR/2003/WD-rdf-syntax-grammar-20030905/>, 2003
- [18] M.Smith, C.Welty, D. McGuinness, "OWL Web Ontology Language Guide Version 1.1", <http://www.w3.org/TR/owl-guide/>, 2003
- [19] Tony Andrews, Francisco Curbera, Hitesh

- Dholakia, Yaron Goland, Johannes Klein, Frank Leymann, Kevin Liu, Dieter Roller, Doug Smith, Satish Thatte, Ivana Trickovic, Sanjiva Weerawarana, "Business Process Execution Language for Web Services. Version 1.1", <http://www.oasis-open.org/committees/download.php/2046/BPEL%20V1-1%20May%205%202003%20Final.pdf>, 2003
- [20] UNSPSC, UNSPSC 웹 사이트, <http://www.unspsc.org/>, 2003
- [21] NAICS, NAICS 웹 사이트, <http://www.naics.com/index.html>, 2003
- [22] Hao He., "What is Service-Oriented Architecture?", <http://webservices.xml.com/pub/a/ws/2003/09/30/soa.html>, www.xml.com, 2003
- [23] Ajar M R., "Web services : A Detail view", <http://isavix.net/whitepapers/1026253685878.pdf>, 2002
- [24] Satish Thatte ., "XLANG : Web Services for Business Process Design", http://www.gotdotnet.com/team/xml_wsp/ecs/xlang-c/default.htm, Microsoft Corporation, 2001
- [25] Arkin, A, Askary, S, Fordin, S, Jekeli, W., Kawaguchi, K, Orchard, D, Pogliani, S., Riemer, K, Struble, S, Takacs-Nagy, P., Trickovic, I., Zimek, S., "Web Service Choreography Interface (WSCI) 1.0", <http://www.w3.org/TR/wsci/>, 2002.
- [26] Web Services Choreography Working Group, <http://www.w3.org/2002/ws/chor/>
- [27] Tony Andrews, Francisco Curbera, Hitesh Dholakia, Yaron Goland, Johannes Klein, Frank Leymann, Kevin Liu, Dieter Roller, Doug Smith, Satish Thatte, Ivana Trickovic, Sanjiva Weerawarana, "Business Process Execution Language for Web Services. Version 1.1", <http://www.oasis-open.org/committees/download.php/2046/BPEL%20V1-1%20May%205%202003%20Final.pdf>, 2003
- [28] Frank Leymann. "WSFL : Web Service Flow Lanuguage 1.0", <http://www-3.ibm.com/software/solutions/webservices/pdf/WSFL.pdf>, 2001
- [29] PATRICK C.K. HUNG, "Specifying Conflict of Interest in Web Services Endpoint Language(WSEL)" [http://www.acm.org/sigs/sigecom/exchange/s/volume_3_\(02\)/3.3-Hung.pdf](http://www.acm.org/sigs/sigecom/exchange/s/volume_3_(02)/3.3-Hung.pdf), 2002

저 자 소 개



김홍기 (E-mail : hgkim@dku.edu)
고려대학교 심리학(학사), University of Georgia 인공지능(석, 박사)
1997. 1 ~ 1998. 2 미국 University of Georgia 인공지능센터 Fellow
2001. 1 ~ 2001. 12 충남 테크노파크 기술개발부 겸임교수
1998. 3 ~ 현재 단국대학교 경상학부 조교수
2001. 1 ~ 현재 충남 전자상거래지원센터 운영위원
2002. 5 ~ 현재 한국 전산원 웹 정보화 자문교수
관심 분야 : 시맨틱 웹, 온톨로지, 웹 서비스, 지식관리시스템, 인공지능



김학래 (E-mail : hkim@dku.edu)
2002. 단국대학교 경영학과 졸업(학사)
2002 ~ 단국대학교 일반대학교 경영정보(석사)
관심 분야 : 시맨틱 웹, 웹 서비스, 지능형 에이전트, 온톨로지 모델링



이강찬 (E-mail : chan@etri.re.kr)
충남대학교 컴퓨터공학과 졸업(학사, 석사, 박사)
현재 한국전자통신연구원 표준연구센터 인터넷응용기술연구팀 선임연구원
ASTAP NG-Web EG Rapporteur
관심 분야 : 정보시스템 구현역량



전종홍 (E-mail : hollobit@etri.re.kr)
1993. 한림대학교 전산학과 졸업(학사)
1996. 한림대학교 컴퓨터공학과 졸업(석사)
1996 ~ 1999. 한국정보시스템 기술개발연구소 주임연구원
1999 ~ 현재 한국전자통신연구원 표준연구센터 선임연구원
관심 분야 : 차세대 웹 기술, 시맨틱 웹, 웹 서비스, KMS/EIP, CSCW, Workflow