

# 시맨틱 웹 서비스 조합을 위한 계획 영역 지식 변환기

김현식<sup>0</sup> 이상연 김인철  
경기대학교 전자계산학과  
(advance7, sylee, kic)@kyonggi.ac.kr

## A Planning Domain Knowledge Converter for Semantic Web Service Composition

Hyun-Sik Kim<sup>0</sup>, Sang-Youn Lee, In-Cheol Kim  
Department of Computer Science, Kyonggi University

### 요 약

최근 들어 자동화된 시맨틱 웹 서비스 조합을 위해 인공지능 계획기법을 이용하려는 연구가 활발하다. 하지만 이러한 계획 기법이 효과적으로 적용되기 위해서는 먼저 시맨틱 웹 서비스의 명세와 온톨로지들을 계획기에서 이용 가능한 계획 영역 지식 형태로 변환하여야 한다. 본 논문에서는 OWL-S로 기술된 웹 서비스 명세와 OWL로 정의된 온톨로지를 표준 계획영역지식 표현언어인 PDDL로 변환하는 변환기의 설계와 구현에 대해 설명한다. 이 변환기의 특징은 기존의 OWLS2PDDL와는 달리 KIF 기반의 전제조건과 효과에 대한 변환 기능을 추가로 제공하며, 웹 서비스의 입출력 데이터에 대한 명세변환도 보다 실용적으로 간소화하였다. 또한, 이 변환기는 계획을 위한 영역모델과 이 영역모델에 기초한 다양한 문제모델들을 별도로 분리하여 생성하며, 이 두 모델 모두에 온톨로지가 적용될 수 있도록 허용한다.

### 1. 서론

웹 서비스와 관련하여 가장 활발한 연구가 이루어지고 있는 분야로 웹 서비스 조합(web service composition)을 들 수 있다. 자동화된 웹 서비스 조합을 위해 BPEL4WS와 Petri Net 등 다양한 방법들이 시도되고 있으나 최근 들어 많은 연구들이 시맨틱 웹 기술에 기초한 서비스 명세와 인공지능 계획 기법을 이용하고 있다. 그러나 이러한 인공지능 계획(AI planning) 기법이 효과적으로 적용되기 위해서는 먼저 시맨틱 웹 서비스의 명세와 온톨로지들을 계획기에서 이용 가능한 계획 영역 지식 형태로 변환하여야 한다. 본 논문에서는 OWL-S로 기술된 웹 서비스 명세와 OWL로 정의된 온톨로지를 표준 계획영역지식 표현언어인 PDDL로 변환하는 변환기의 설계와 구현에 대해 설명한다. 이 변환기의 특징은 기존의 OWLS2PDDL에서 처리하지 못했던 KIF(Knowledge Interchange Format)[1] 기반의 전제조건(Precondition)과 효과(Effect)에 대한 변환 기능을 추가로 제공하며, 웹 서비스의 입출력 데이터에 대한 명세변환도 보다 실용적으로 간소화하였다. 또한, 이 변환기는 계획을 위한 영역모델과 이 영역모델에 기초한 다양한 문제모델을 별도로 분리하여 생성하며, 이 두 모델 모두에 온톨로지가 적용될 수 있도록 허용한다. 본 논문에서는 변환기의 유용성을 점검하기 위해 몇 가지 시맨틱 웹 서비스 예를 대상으로 변환작업을 하여 웹 서비스 조합을 위한 자동 계획생성을 시도해본다.

### 2. 관련 연구

OWL-S[3]는 표준 웹 온톨로지 언어인 OWL을 개발한 미국의 DAML(DARPA Agent markup Language) 컨소시엄에 의해 W3C 표준으로 제안된 시맨틱 웹 서비스 온톨로지이다. OWL-S는 전통적인 인공지능 계획 연구분야와 지능형 에이전트 및 다중 에이전트 시스템 분야로부터 많은 영향을 받았으며, OWL 즉, Description Logic의 풍부한 표현력과 웹 서비스 표준들의 실용성을 결합한 산물로 볼 수 있다. OWL-S 온톨로지는 서비스 개요를 기술하는 서비스 프로파일(Profile)과 서비스 실행과 연관된 프로세스 정보를 제공하는 프로세스

모델(Process Model), 서비스 매핑을 담당하는 서비스 그라운드링(Grounding) 등 3가지 구성요소를 포함하고 있다.

PDDL(Planning Domain Definition Language)[4]은 전통적인 계획 생성을 위한 작업 업무룰 구성하기 위한 표준 언어로서 도메인(Domain)과 문제(Problem)로 이루어져 있다. PDDL 계획 작업 구성 요소는 월드 정보를 나타내는 객체(Objects), 객체 속성을 나타내는 술어(Predicates), 월드 정보의 처음 상태를 나타내는 초기 상태(Initial state), 목표하고자 하는 상태를 나타내는 목표 명세(Goal specification), 월드의 상태를 변경하는 행동/연산자(Actions/Operators)로 구성되어 있다.

OWLS2PDDL[5,6]은 복합적인 계획 생성기인 XPlan의 서비스 정보 변환기로서, 본 논문에서 다루고 있는 변환기와 같이 서비스 명세 정보인 OWL-S 정보와 도메인 온톨로지 정보인 OWL 정보를 입력으로 받아 PDDL로 변환한다. OWLS2PDDL은 OWL-S 서비스 명세의 전제 조건과 효과를 OWL-S 1.1에서 제안하고 있는 방법과는 별도의 방법으로서 XML 형식으로 기재하고 있다. 이러한 정보 표현 방법은 현재 OWL-S 명세의 전제조건과 효과의 정보 표현의 범위와 표준화와 관련되어 발생하고 있는 문제를 해결하기 위한 것으로서 차후 변경되거나 다양하게 표현되고 있는 전제조건과 효과를 처리하기 위한 것으로 보인다. 그러나 이러한 처리 방법은 실제 정의된 OWL-S 정보만을 바탕으로 계획 생성을 위한 서비스의 전제조건과 효과를 정상적으로 처리할 수 없다. 또한 OWLS2PDDL에서는 계획 생성을 돕기 위해 서비스 정보와 온톨로지 정보 이외에 에이전트가 알고 있는 지식 정보를 정의하고 있다. 이러한 지식 정보는 계획 생성을 하기 위한 계획 과정 중에 발생할 수 있거나 발생할 정보를 미리 정의해 놓은 것으로서 실제 서비스를 이용하는데 있어서는 많은 제약이 뒤따르게 된다.

### 3. 변환기 구성

변환기(Converter)는 이용 가능한 시맨틱 웹 서비스 명세인 OWL-S에서 필요로 하는 정보를 분석해 내는 Service Parser와

Service Parser로부터 서비스 명세의 전제조건과 효과 부분을 받아 분석하는 KIF Parser, OWL 온톨로지 정보를 분석하는 Ontology Parser, 분석된 각각의 정보를 이용하여 도메인 정보를 생성하는 Domain Converter와 온톨로지 인스턴스(instance) 정보와 사용자가 알고 있는 정보를 받아 계획 문제를 생성하는 Problem Converter, 그리고 이러한 정보를 바탕으로 PDDL 형태의 계획 도메인과 문제를 작성하는 하는 Domain File writer와 Problem File Writer로 구성되어 있다.

[그림 1]은 이러한 변환기의 각 구성요소에 대한 기능과 동작 형태를 보인다. 먼저 Service Parser와 Ontology Parser는 입력으로 OWL-S 명세 정보와 온톨로지 정보인 OWL 정보를 받게 된다. OWL-S 서비스 명세는 Domain Service Parser와 KIF Parser에서 분석되게 되는데, KIF Parser는 KIF로 작성된 서비스의 전제 조건과 효과에 대한 명세 부분만을 처리한다. Ontology Parser는 입력으로 받은 온톨로지 정보를 클래스(Class)와 속성(Property), 인스턴스 정보로 분류하여 클래스와 속성은 계획 도메인 정보를 구성하는 Domain Converter로 보내지게 되며, 인스턴스 정보는 사용자 정보 입력을 거쳐 계획 문제를 구성하는 Problem Converter로 보내지게 된다. Domain File Writer와 Problem File Writer는 계획기의 입력 가능한 형태의 PDDL 파일을 작성하게 된다.

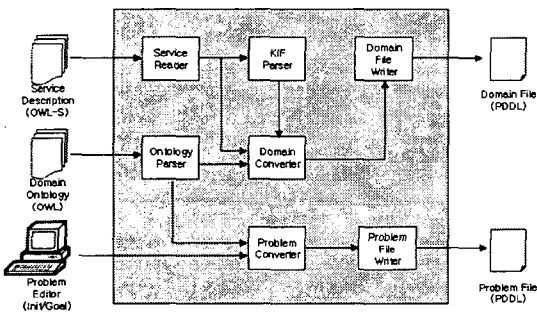


그림 1 변환기 구조

4. 변환 규칙

변환기는 입력 정보로서 서비스 정보인 OWL-S 정보와 OWL-S가 이용하는 온톨로지 정보인 OWL 정보를 입력으로 받게 된다. 이러한 입력된 정보는 SAPA나 FF와 같은 계획기에서 입력 정보로 받아들일 수 있는 형태인 PDDL 형태로 변환되게 된다. 변환기의 변환 규칙은 [그림 2]과 같다.

입력은 크게 온톨로지 정보인 OWL과 서비스 정보인 OWL-S로 나뉜다. OWL의 정보중에서 변환에 이용되는 것으로는 클래스 정보와 속성 정보, 그리고 외부로의 입력 혹은 기존 입력되어 있는 실제 값들인 인스턴스 정보를 이용을 하게 된다. OWL에서의 클래스는 동일한 속성을 지니고 있어 하나의 부류로 모아지는 개체들의 군을 정의하는데, 이 정보는 PDDL에서 도메인의 타입(Types) 정의 부분으로 변환되게 된다. 도메인의 타입은 해당 객체들의 타입 정보를 정의하게 된다. 이러한 정의를 바탕으로 할 때, PDDL의 객체 타입 정의는 OWL의 클래스 정보를 바탕으로 정의할 수 있다.

속성 정보는 클래스의 속성을 나타내며, 개체간의 관계 및 제제와 데이터 값 사이의 관계를 나타낸다. 이러한 속성 정보는 변수에 대한 특성 정보를 나타내는 도메인의 Predicates와 OWL-S 서비스 정보 부분의 전제조건과 효과 정보의 의미적 관계로 이용될 수 있다.

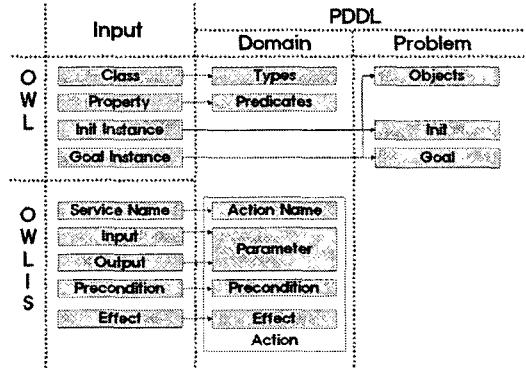


그림 2 정보 변환 규칙

외부로의 입력 혹은 기존 입력되어 있는 실제 데이터 값들인 인스턴스 정보는 변수의 실제 초기, 혹은 목표 상태 값을 나타내는 초기 상태(Initial State)와 목표(Goal)로 나타낼 수 있으며, 해당 타입의 객체를 정의하는 객체(objects) 정보를 나타낼 수 있다. 이러한 실제 값을 나타내는 인스턴스 정보는 상태를 정의하는 도메인 부분에서는 사용하지 않지만, 실제 문제를 정의하는 계획 문제 부분에서의 해당 객체의 타입을 정의하는 객체 부분과 초기 상태 정보 그리고 목표하고 있는 상태를 정의하는데 필수적이다. 위에서 설명한 바와 같이 인스턴스 정보는 온톨로지 정보인 OWL에 정의가 되며, 기존 정보로 정의가 되어 있거나 혹은 사용자가 해당 데이터 값을 정의해 주어야 한다. 이러한 인스턴스는 PDDL의 계획 문제 작성에 있어 모든 구성 요소에 영향을 주게 되는데, 계획 문제의 경우 초기 상태와 함께, 목표로 하고자 하는 최종 상태를 값으로 주어야 한다. 그러나 인스턴스가 OWL에 정의되어 있거나 혹은 사용자가 입력할 때, 초기 상태와 목표 상태를 해당 인스턴스를 입력한 사용자를 제외하고는 구분할 수 있는 기준이 없기 때문에 변환기에서는 이러한 인스턴스 처리를 위하여 초기 상태와 목표 상태로 구분하여 입력 받아 처리하였다. [그림 3]에서 보는 바와 같이 입력된 인스턴스는 초기 상태나 목표 상태나에 따라 처리 방법이 변하게 된다.

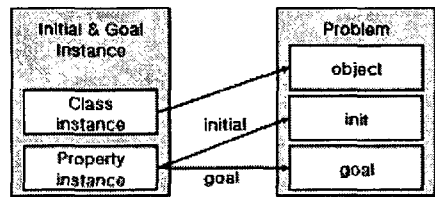


그림 3 인스턴스 변환 방법

인스턴스 정보는 크게 OWL의 클래스에 대한 정보와 속성에 대한 정보로 나뉘어 되는데, 클래스에 대한 정보는 초기 상태와 목표 상태 모두 PDDL의 객체 부분에 선언되게 되며, 속성에 대한 정보는 초기 상태인 경우 PDDL의 초기 상태로, 목표 상태의 경우 PDDL의 목표 부분으로 변환되게 된다.

서비스 정보를 나타내는 OWL-S 명세에서는 서비스명(Service Name), 입력(Input), 출력(Output), 전제조건(Precondition), 효과(Effect) 정보를 이용하여 PDDL의 도메인 부분의 행동(Action) 부분을 생성하게 된다. 도메인의 행동 부분은 행동명(Action Name)과 행동에서

