웹 기반 OWL 온톨로지 접근을 위한 SPARQL 쿼리 툴

조대웅, 최지웅, 김명호
숭실대학교 컴퓨터학과
e-mail : dwjo@ss.ssu.ac.kr, jwchoe@ss.ssu.ac.kr, kmh@ssu.ac.kr

SPARQL Query Tool for Web-based OWL Ontology Access

Daewoong Jo, Jeewoong Choe, Myungho Kim
Dept. of Computer Science, SoongSil University

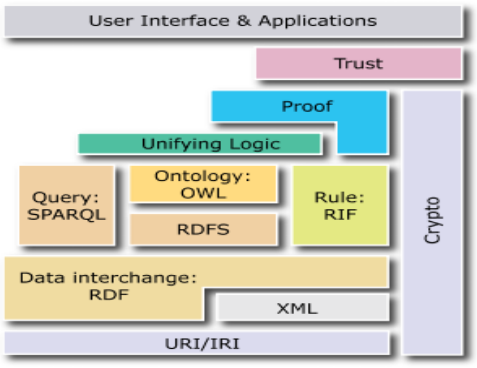
요 약

지금까지의 웹은 표현중심의 HTML에서 데이터의 구조적인 모습을 나타내는 XML로 그리고, 온톨로지 구축을 통해 의미 있는 데이터의 접근성에 초점을 둔 RDF, RDFS, OWL과 같은 형태로 변화 하고 있다. 이러한 흐름을 통해 현재는 OWL을 이용하여 온톨로지 구축을 위한 연구가 진행 중이다. 이 논문에서는 구축된 온톨로지를 활용하기 위해 웹에 있는 OWL 파일을 http 프로토콜을 이용하여 접근을 하고 기존의 툴 종속적인 접근이 아닌 웹 표준 프로토콜과 OWL 파일 형식으로 접근 하여 SPARQL 쿼리를 보낼 수 있는 웹 기반의 OWL 온톨로지 접근을 위한 SPARQL 쿼리 툴을 제안한다.

1. 서론

시맨틱 웹(Semantic Web)[1]은 현재의 인터넷과 같은 분산환경에서 리소스(웹 문서, 각종파일, 서비스 등)에 대한 정보와 리소스 사이의 관계를 통해 의미 정보(Semantics)를 기계(컴퓨터)가 처리할 수 있는 온톨로지 형태로 표현하고, 이를 자동화된 기계(컴퓨터)가 처리하도록 하는 프레임워크이자 기술이다[12].

(그림1)은 W3C의 Tim Berners Lee 에 의해 제안된 시맨틱 웹의 스택이다. 현재 시맨틱 웹 기술은 OWL(Web Ontology Language)과 같은 온톨로지 언어까지 발전된 상태이다.



(그림 1) 시맨틱 웹 스택

XML, RDF, OWL과 같이 시맨틱 웹을 위해 필요한 기술들의 표준화 작업이 가시화 되면서 이를 활용하기 위한 연구가 활발히 진행되고 있다. 하지만 현재 연구 되고 있는 분야는 OWL을 이용하여 온톨로지를 구축하는 구축 툴 중심이며, 구축된 온톨로지를 활용하는 데에는 연구가 더 필요하다.

본 논문에서 제안하는 툴은 온톨로지를 구축하는 것이 아닌 구축된 온톨로지를 활용하기 위한 것 이다. 웹 표준 온톨로지 기술 언어인 OWL, RDF 파일을 기존의 로컬 접근 방식에서 벗어나 웹으로 접근이 가능 하게끔 하였다. 그래서 온톨로지 파일의 URL만 안다면 툴에서 접근하여 온톨로지 정보를 확인할 수 있고, 온톨로지 쿼리 언어인 SPARQL(SPARQL Protocol and RDF Query Language)[6]을 이용하여 원하는 데이터만을 툴 안에서 검색이 가능하다. 검색된 데이터는 BI(Business Intelligence)를 위한 레포팅 툴[14]에서도 사용이 가능하며 온톨로지를 이용하는 지능형 웹 에이전트 에게도 활용성은 요구된다.

따라서 본 논문에서는 이와 같은 온톨로지 활용성에 기초한 웹 기반의 SPARQL 쿼리 툴을 제안한다.

2. 관련 연구

본 장에서는 논문에서 제안하는 시스템을 위한 웹 기반 표준 기술들인 OWL과 OWL 파일의 검색을 위한 SPARQL에 대해 알아본다. 그리고, 다른 온톨로지 툴들을 살 펴 본다

2.1 OWL

OWL[3]은 웹에서 온톨로지를 표현하기 위한 언어로 W3C 에서 2002년부터 발전하여 현재 OWL 1.1 까지 나온 상태며 OWL 2.0이 개발 중으로 있다.

OWL 은 기존의 RDF 와 RDF Schema를 바탕으로 온톨로지를 표현 하기 위한 더 많은 문법들이 추가 되었다. OWL 구문은 클래스(class), 속성(property)과 원소(individual)의 3가지 기본 요소를 가진 것으로 요약될 수 있다[3].

OWL은 3가지의 형태가 있으며 각각은 OWL-Lite, OWL-DL, OWL-Full로 나누어진다. OWL-Full은 OWL-DL을 포함하고, OWL-DL은 OWL-Lite를 포함하는 구조로 OWL-Lite에서 OWL-Full로 갈수록 온톨로지를 표현 할 수 있는 표현력이 늘어난다[3]. 현재는 OWL을 추론 할 수 있는 추론(Reasoning)엔진이 OWL-DL 수준까지여서 대부분의 OWL 파일은 OWL-DL의 문법에 따라 쓰여지고 있는 추세이다.

2.2 SPARQL

시맨틱 웹 환경에서 온톨로지 정보를 검색하기 위한 RQL[7], RDQL[8], SPARQL과 같은 다양한 쿼리 언어가 개발되었으며 이중에서 SPARQL은 W3C에 의해 권고안으로 채택된 언어로서 이를 지원하는 다수의 시스템이 개발되고 있다[9].

SPARQL을 이용한 간단한 예제는 다음과 같다. <표1>은 SPARQL을 이용해 검색 하게 될 데이터가 되겠고, <표2>는 쿼리 문을 보여주고 있다. 쿼리는 기본적으로 SELECT, FROM, WHERE의 SQL문과 비슷한 문법을 가지고 있다. <표3> 은 쿼리에 대한 결과를 나타내고 있다.

<표 1> 데이터

```
@prefix foaf : <http://xmlns.com/foaf/0.1/>
 _a:foaf:name "Johnny Lee Outlaw".
 _a:foaf:mbox <mailto:jilow@example.com>.
 _b:foaf:name "peter Goodguy".
 _b:foaf:mbox <mailto:peter@example.org>.
 _c:foaf:mbox <mailto:carol@example.org>.
```

<표 2> 쿼리

```
PREFIX foaf: http://xmlns.com/foaf/0.1/
SELECT ?name ?mbox
WHERE
        {?x foaf:name > name .
         ?x foaf:mbox ?mbox}
```

<표 3> 쿼리 결과

names	mbox
Johnny Lee Outlaw	<mailto:jlow@example.com>
Peter Goodguy	<mailto:peter@example.com>

2.3 온톨로지 툴

OWL 및 RDF 파일을 편집하고, 나타내는 데는 대표적으로 Stanford University에서 만든 Protégé[11]가 있다. 현재 Protégé는 4.0Beta까지 나온 상태이다. 마지막 버전이던 Protégé 3.x과 Protégé 4.0의 차이로는 Protégé 3.x는 OWL 1.0을 지원하였으나, Protégé 4.0이 되면서는 OWL 2.0까지 지원이 가능하다. 하지만 Protégé 4.0은 아직 OWL 2.0을 기반으로 한 SPARQL은 지원 못하고 있다.

기존 Protégé 3.x에서는 웹 상의 OWL 파일의 접근은 툴의 종속적인 형태로의 접근이 요구되었다. Protégé에서 쓰이는 .pprj 형태인 프로젝트 파일을 서버에 올려놓아야 접근이 될 뿐 OWL 파일의 확장자인 .owl에 대한 접근은 이루어 지지 못하였다. 이러한 점은 Protégé에서 만든 온톨로지 파일이 아니면 안되므로 웹 접근을 통한 온톨로지 활용에 있어서는 불편한 점이 있었다.

그 외, ontoprise의 OntoStudio, Eclipse IDE 기반의 OWL 에디터인 SWeDE(Semantic Web Development Environment), MIT 에서 만든 Swoop과 같은 온톨로지 편집 툴들이 있다. 이러한 툴들은 로컬 상태의 OWL 파일을 가져오거나 온톨로지를 만드는데 초점이 있는 반면 웹 상의 OWL 파일을 접근하거나 로딩해서 SPARQL을 이용하여 쿼리 작성을 위한 기능은 같이 제공하지 못하고 있다.

따라서 본 논문에서 제안하는 SPARQL 쿼리 툴은 이와 같은 점을 보완하여 이 전 툴에서 보던 온톨로지의 저작과 툴에 종속적인 웹의 접근이 아닌 웹 표준 .owl .rdf와 같은 파일에 직접 접근하여 온톨로지를 불러올 수 있으며 불러온 파일의 온톨로지 정보를 바탕으로 SPARQL 쿼리도 함께 작성 할 수 있다.

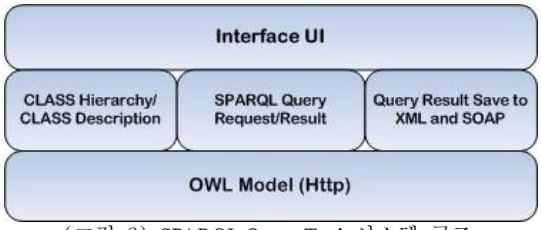
3. 시스템 요구사항

본 논문에서 제안하는 SPARQL 쿼리 툴을 구현하기 위한 시스템 요구사항은 다음과 같다.

1) http 프로토콜을 사용한 온톨로지 접근이 이루어 져야 한다.
2) OWL 파일의 클래스구조(ClassHierarchy) 및 제한사항을 포함한 클래스 요약 정보들을 볼 수 있도록 UI로 나타내주어야 한다.
3) SPARQL 문법을 모르는 일반 사용자를 고려하여 SPARQL 쿼리를 직접 쿼리 문 작성이 아닌 UI를 통해 작성을 할 수 있어야 한다.
4) SPARQL 쿼리 결과를 나타낼 수 있는 UI가 필요 하다.
5) SPARQL 쿼리 결과를 XML 형식으로 저장하고 다시 그 파일을 불러 왔을 때 결과가 로딩 되어서 계속적인 분석이 이루어 질 수 있도록 한다.
6) SPARQL 쿼리 결과를 다른 웹 에이전트와의 정보 교환을 위해 웹 서비스 프로토콜인 SOAP 형태로 변환이 요구된다.

4. Query Tool 시스템 구조

앞 장에서 살펴본 시스템 요구사항을 바탕으로 시스템에서 필요한 기능에 대한 구조는 (그림 2)와 같다.



(그림 2) SPARQL Query Tool 시스템 구조

본 논문에서 제안한 SPARQL Query Tool은 웹에 있는 OWL 파일을 접근한다. 따라서, http 프로토콜을 통해 OWL 파일에 접근하고, 접근된 파일을 읽어서 하나의 OWL을 클라이언트에서 내부 Model화 시킨다. Model화 된 OWL 파일은 클래스 구조(Class Hierarchy)와 전체적인 Individual List 정보도 함께 보여주며 개별 클래스에 대한 클래스 요약 정보도 제공한다. 따라서 사용자에게 제공된 온톨로지 정보를 바탕으로 SPARQL 쿼리를 작성하고, 결과를 볼 수 있다. 쿼리 결과는 XML 파일 형식 또는 웹 서비스 지원을 위한 SOAP 형태로 저장 될 수 있다.

5. 구현

본 장에서는 사용된 API와 실제로 구현된 툴의 인터페이스에 대해 살펴 본다.

5.1 API

다음은 본 논문에서 구현한 SPARQL Query Tool 의 사용된 API에 대한 설명이다. 기본 베이스는 Java 6을 기반으로 UI는 Swing을 이용하였다. 그리고, 웹에 있는 OWL 파일을 가져오기 위해 http 프로토콜을 이용하여 접근하였다.

OWL 파일의 모델화 및 SPARQL 쿼리 질의를 위한 API 로는 Jena2, Protégé, ARQ 가 사용되었으며 기본 설명은 다음과 같다.

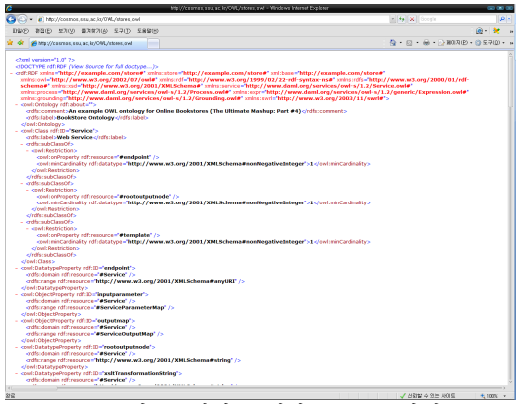
Jena2 – Jena 는 시맨틱 웹 응용프로그램을 만들기 위한 자바 프레임워크(framework)다. Jena 는 RDF, RDFS와 OWL, SPARQL을 프로그램 하기 위한 API를 제공한다[10].

Protégé-owl – Protégé-owl API는 owl과 RDF를 위한 Java 기반의 오픈 소스이며, 좀 더 풍부한 UI의 API를 위해 사용되었다[11].

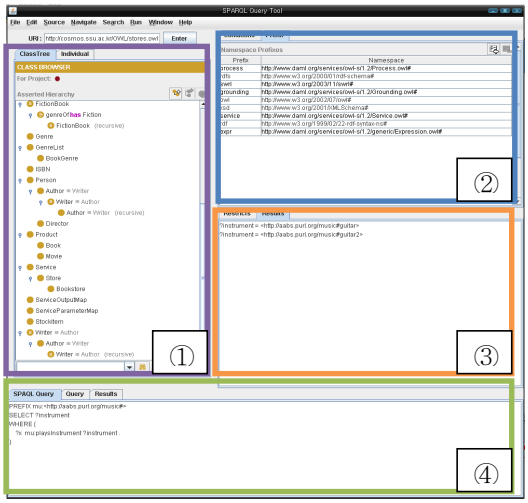
ARQ – ARQ 는 SPARQL 및 RDQL과 같은 RDF Query 언어를 지원하기 위한 Jena 기반의 쿼리엔진이다[13].

5.2 인터페이스

웹에 존재하는 OWL, RDF 파일의 URL를 통해 이 툴에서는 Tree 형태로 클래스들을 읽어 들인다. 읽어 들인 OWL 파일의 온톨로지 정보를 바탕으로 SPARQL 쿼리를 사용하여 원하는 정보를 얻을 수 있다.

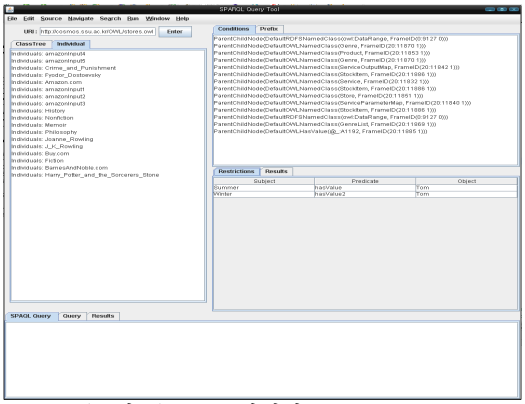


(그림 3) 웹에 존재하는 OWL 파일



(그림 4) SPARQL 쿼리 툴

(그림3) 에서는 실재 웹 상에 존재하는 OWL 파일을 보여 주고 있다. (그림 4)는 전체적인 툴의 모습을 보여 주고 있다. (그림 4)에서 ①에 해당하는 부분이 (그림3)의 OWL 파일을 접근하여 그 파일에 대한 온톨로지의 구조를 트리 형태로 보여주는 화면이다. ①의 탭을 통해 OWL 파일의 Individual List를 볼 수 있다. 사용자는 Individual List와 클래스 구조를 보고 SPARQL 쿼리를 작성할 때 참고 한다. ②는 이 OWL 파일의 Prefix 부분에 대한 정보를 나타내주고 있으며, 탭을 통한 Conditions에서 관련된 클래스의 Restriction정보가 나타난다. ③은 ④에서 작성한 SPARQL 쿼리에 대한 쿼리결과를 나타내주는 판넬이다. 지금까지 구현된 부분은 여기까지 이며 구현중인 부분은 쿼리 결과 값에 대한 XML Format 변환 및 SOAP 형태로의 변환을 구현 중에 있다.



(그림 5) OWL 파일의 Individual List

6. 결론 및 향후 과제

시맨틱 웹 연구에서 핵심 요소는 데이터들을 온톨로지 시켜 메타데이터를 풍부하게 하는 것이다. 그래서 현재 까지는 온톨로지를 구축하기 위한 구축 툴 중심으로 연구가 진행되었다.

하지만 본 논문에서 제안하는 SPARQL 쿼리 툴은 온톨로지의 구축이 아닌 구축된 온톨로지를 활용하기 위한 것이다. 그래서 이미 구축된 웹 상의 온톨로지 파일(OWL, RDF)을 http 프로토콜을 통해 파일의 URL로 접근이 가능하다. 이러한 웹으로의 접근은 기존 다른 툴에서도 있었다. 하지만 툴에 종속적인 모습으로 툴에서 만든 파일이 있을 때 만 가능 한 것이지 표준 온톨로지 파일(OWL, RDF)로 바로 접근은 시도 되지 못하였다. 그래서 기존의 이와 같은 방식을 보완하여 표준 온톨로지 파일 형식인 .OWL, .RDF의 파일로 툴 안에서 바로 접근이 가능하도록 하였다. 그리하여 OWL 파일 안의 Class, Property, Individual과 같은 온톨로지 정보들을 가져와서 확인하고 SPARQL 쿼리를 통해 원하는 정보를 검색이 가능하다.

지금까지의 구현은 하나의 툴 에서 이와 같은 일이 수행되는 것이며 현재 다음과 같은 작업을 구현 중으로 쿼리 결과를 SOAP 형태로 저장하여 웹의 에이전트끼리와의 정보 교환에 있어서 웹 서비스로의 접근이 가능 하도록 하고 있다.

향 후 발전 사항으로는 첫 째, 현재의 SPARQL 쿼리를 만드는 과정은 SPARQL 쿼리 문법을 안다는 가정하에 작성을 해야 원하는 결과를 얻을 수 있었다. 하지만, 앞으로 보완사항에서는 자동으로 SPARQL 쿼리 가 나 올 수 있도록 하고자 한다. 즉, 사용자는 원하는 Individual 또는 결과를 얻기 위해 쿼리 문을 입력할 필요 없이 조건 및 제어를 할 수 있는 콤보 박스를 마련해서 그곳에서 원하는 조건들을 클릭 하면 쿼리가 자동으로 매칭이 되는 형식으로 보완하고자 한다. 이렇게 했을 시 장점은 정확한 쿼리 문을 몰라도 OWL 파일에서 원하는 결과값을 이끌어 낼 수 있다는 장점이 있다. 둘 째, BI(Business Intelligence)와 관련하여 레포팅 툴에 SPARQL 쿼리 툴 을 결합하여 웹 상의 OWL 파일을 읽어 들여 원하는 Individual을 SPARQL을 이용하여 결과를 얻어내고, 그 결과를 가지고 레포팅 툴 을 이용하여 분석을 할 수 있도록 하는 것이 향 후 과제로서 남아 있다.

참고문헌

[1]T.Berners-Lee, J.Hendler, and O.Lassila, "TheSemantic Web," *Scientific American*, 2001.

[2]OWL Web Ontology Language Guide, http://www.w3.org/TR/owl-guide/, 2004.

[3]OWL Web Ontology Language Overview, http://www.w3.org/TR/owl-features/,2004

[4]A. Seaborne and E. Prud' hommeaux. SparQL Query Language for RDF, http://www.w3.org/TR/rdf-sparql-query/,2008

[5]이현실, 한성국, "OWL을 이용한 온톨로지 기반의 목록시스템 설계 연구", 2004.6

[6]SPARQL Query Language for RDF, http://www.w3.org/TR/rdf-sparql-query/,2008

[7]The RDF Query Language (RQL), http://139.91.183.30:9090/RDF/RQL

[8]RDQL-A Query Language for RDF, W3C, 2004

[9]손지성, 정동원, "SPARQL-to-SQL 변환 알고리즘의 저장소 독립적 활용을 위한 시스템 모델, 2008.7

[10]Jena 2, http://jena.sourceforge.net

[11]Protégé, http://protege.stanford.edu

[12]Wikipedia, http://en.wikipedia.org/wiki/Semantic_Web

[13]ARQ-ASPARQL Processor for Jena, http://jena.sourceforge.net/ARQ/

[14]최지웅, 김명호, "웹 기반의 Ad Hoc 리포팅을 위한 Fat Client를 갖는 리포팅 툴", 정보과학회논문지 제12권 제4호, 2006.8