

OWL 문서의 저장과 질의 형태에 관한 연구

고명석¹, 김연희¹, 김병곤², 임해철¹
¹홍익대학교 컴퓨터공학과, ²부천대학 e-비즈니스과
 (msko, im, kyh)@cs.hongik.ac.kr, bgkim@bc.ac.kr

The Study of Storing and Query Pattern for OWL Documents

Myoung Suk Ko¹, Youn Hee Kim¹, Byung Gon Kim², HaeChull Lim¹
¹Dept. of Computer Engineering, Hongik University
²Dept. of e-Business, Bucheon College

요 약

차세대 웹 환경을 위한 시맨틱 웹에 대한 관심이 날로 증대되고 있다. 이와 더불어 시맨틱 웹에서 자원의 의미적인 관계를 표현하기 위한 언어들에 대한 연구가 활발히 진행중이다. 특히 이러한 연구의 시작점으로 RDF, DAML+OIL 같은 여러 온톨로지 언어들도 등장하였지만 자원의 특성에 대한 기술이나 그들간의 관계에 대한 정의를 명확하게 표현하기에는 미흡한 점이 많다. 그러나 최근에 제안된 OWL은 RDF나 RDF 스키마를 확장하여 보다 의미있는 자원의 관계를 정의할 수 있다. 본 논문에서는 OWL로 표현된 문서를 저장하기 위한 구조를 관계형 데이터베이스를 기반으로 제안한다. 본 논문에서 제안한 저장 구조는 기존의 RDF와 RDF 스키마를 위한 저장 구조를 확장하여, OWL을 통해 제공되는 클래스나 속성간의 동일성, 또는 이질성, 여러 클래스의 집합 관계 등의 추가적 기능을 지원하는데 목적을 두고 있다. 또한 제안한 저장구조를 이용한 OWL 문서의 질의 형태를 분석하고 이를 기존 RQL을 이용하여 표현하기 위해 OWL의 특성을 반영하도록 RQL을 확장했다.

1. 서 론

최근 차세대 웹으로 시맨틱 웹(Semantic Web)이 각광을 받고 있다. 기존의 웹 환경과는 달리 시맨틱 웹은 웹 상에 정보의 리소스들이 서로 의미적 연결을 가지고 있고 인간과 기계 모두가 쉽게 문서를 이해할 수 있도록 지원한다. 이런 의미적인 연결을 기술하기 위해 RDF(Resource Description Framework), RDF 스키마, OWL(Web Ontology Language)등이 온톨로지 언어로 제안되었다.

온톨로지는 용어와 용어들간의 관계를 정의해 놓은 것으로서 보다 의미있는 자원간의 관계를 표현할 수 있다[1]. 이러한 온톨로지를 표현하기 위해 W3C에서 제안된 RDF는 웹상에 존재하는 자원들의 메타데이터를 기술하기 위한 언어이다. RDF는 XML 문법을 따르고 있으며, 기본적으로 자원-속성-값의 기본 구조를 이용하여 자원의 메타데이터를 기술한다[3]. RDF는 XML에서 문체점으로 제시되었던 태그 사용에 있어서의 모호성이 여전히 존재한다. 예를들면, 동음이의의 태그 관계나 이음동의의 태그 관계를 정확히 표현하지 못한다. 그래서 RDF에서는 RDF 스키마를 사용하여 이러한 문제를 해결하고 있다. RDF 스키마는 기술 대상이 되는 자원의 클래스 타입과 그 자원을 표현하기 위해 필요한 속성들을 정의하고 클래스와 클래스간의 관계, 속성과 속성간의 의미적 관계를 명확히 정의한다[2].

그러나 RDF 스키마는 클래스간 또는 속성간의 동일 관계, 이질 관계, 역관계의 정의 그리고 union, intersection등과 같이 클래스간의 관계 정의나 속성에 대한 제약을 표현하는데는 한계가 있다. 그래서 W3C에서는 RDF와 RDF 스키마의 부족한 모델링 요소를 보완하여 OWL을 제안하였다[1]. OWL의 풍부한 표현력때문에 많은 사람들이 OWL에 관심을 가지게 되었고 이와 관련한 연구가 증가하는 추세이다. 하지만 지금까지는 RDF와 RDF 스키마를 위한 저장 및 질의

시스템에 대한 연구가 대부분이었고, OWL 문서를 대상으로 하는 연구가 미진한 상태이므로 OWL의 특성을 지원하는 저장 구조가 필요하다.

따라서 본 논문에서는 첫째, RDF 스키마에 비해 확장된 OWL의 기능을 분류하였다. 둘째, OWL의 확장된 특성을 지원할 수 있는 관계형 데이터베이스 기반의 저장구조를 제안한다. 마지막으로 RDF와 RDF 스키마에서 확장된 OWL의 특성에 관한 질의가 가능하도록 기존 RQL을 확장하였다.

2. 관련 연구

RDF 스키마는 자원의 의미에 대한 동일성, 이질성, 집합에 대한 표현 등과 같은 자원의 관계 정의 부분이 미흡하다. W3C에서는 이러한 RDF의 미흡한 점을 보완하여 OWL이라는 온톨로지 언어를 제안하였다. 그러나 RDF와 RDF 스키마를 위한 대표적인 저장 시스템인 RDFSuite와 Sesame 시스템 등은 이러한 OWL의 추가적인 기능을 완벽하게 저장하기 어렵고, OWL에 대한 새로운 질의 형태를 제공하지 않는 단점이 있다.

2.1 OWL

OWL은 크게 OWL Lite, OWL DL, OWL Full로 구성되어 있다. 후자로 길수록 클래스나 속성들에 대한 특성 정의나 관계 정의에 대한 표현력이 더 풍부하지만 사용되지 않는 기능이 많기 때문에 본 논문에서는 OWL Lite에 초점을 맞추고, OWL DL나 OWL Full에서 필수 기능이라고 생각되어 지는 unionOf, intersectionOf등 클래스들의 집합 관계에 대한 표현 부분만을 포함하였다[1].

2.2 RDFSuite

ICS-FORTH의 RDFSuite 시스템은 RDF 문서의 파싱을 위한 파서인 VRP, RDF의 저장 구조인 RSSDB 그리고 객체-관계 데이터베

이스에 기반한 RDF의 질의 처리기로 구성되어 있다. RDFSuite는 클래스와 속성 정보를 저장한 테이블들과 그들의 포함관계를 포함한 subclassOf와 subPropertyOf 테이블을 유지하고 각 속성마다 테이블을 만들어 관계있는 자원들을 연결함으로써 객체-관계 데이터베이스를 기반으로한 저장 구조에 저장하고 질의 가능하도록 개발하였다[6].

2.3 Sesame

On-to-Knowledge 프로젝트의 일부인 Sesame 시스템은 객체-관계 데이터베이스 뿐만 아니라 관계형 데이터베이스에서도 저장 가능한 구조를 제공한다. Sesame의 저장 구조는 RDFSuite와 거의 유사하다. 다만, 속성의 도메인과 레인지 테이블을 각각 따로 유지한다는 점이 차이점이다[5].

RDFSuite, Sesame 시스템 모두 데이터를 저장하고 RQL 질의가 들어오면 이를 SQL3로 변환하여 결과값 반환한다. RQL은 RDF와 RDF 스키마에 대해 질의하기 위해 제안된 질의 언어로, 기존의 SQL과 비슷한 형태로 사용자가 보다 쉽게 접근 가능하도록 지원한다[4].

OWL을 두 시스템에 저장하면 OWL이 지원하는 기능들은 일반 RDF 속성으로 저장되므로 원하는 질의에 대한 정확한 결과물을 얻지는 못한다. 따라서 OWL에 맞는 새로운 저장구조의 개발이 필요하다.

3. OWL의 추가 기능 분류

본 장에서는 OWL이 클래스나 속성간의 관계를 정의하는데 있어 RDF, RDF 스키마에 비해 확장된 개념들에 대하여 분류한다. 특히 클래스간 또는 속성간의 관계 표현에 필요한 의미의 동일성과 이질성에 대한 분류와 클래스 집합에 대한 표현은 RDF나 RDF 스키마에서는 지원하지 않는다.

<그림 1>은 OWL에서 확장 요소를 분류하고 그들의 기능을 간단히 설명한다[1][7].

구분	Property	설명
동위성	equivalentClass	- 두 클래스가 동치임을 표현
	equivalentProperty	- 두 속성이 동치임을 표현
이질성	disjointFrom	- 두 개체가 서로 다르다는 사실용 정의
	AllDifferent	- 여러 개의 개체가 서로 다르다는 사실용 한번에 기술
클래스 집합	unionOf	- 기존에 있던 클래스들을 하나의 클래스 집합으로 묶어 새로운 클래스 정의
	intersectionOf	- 평가될 모든 조건들을 동시에 만족 시켜야 하는 제약용 두고 있다
	complementOf	- 특정 클래스에 속하지 않는 클래스용 정의
	one of	- 한 클래스를 구성하는 구성요소용 정의
의관계	disjointWith	- 한 클래스의 구성원이 다른 클래스의 구성원이 될 수 없다는 사실용 규정
	inverseOf	- 두 속성간의 역 속성의 명시 가능

<그림 1> OWL의 확장 요소 분류

4. OWL 문서의 저장구조

본 장에서는 OWL이 제공하는 기본적인 기능뿐만 아니라 클래스나 속성의 동치 관계, 이질 관계, 역관계, 클래스들의 집합 관계의 추가적 기능을 위해 관계형 데이터베이스에 기반한 저장구조를 제안한다. 본 논문에서 제안한 저장구조는 Class, Property, Namespace, subclassOf, subPropertyOf, Tag, EqualClass, EqualProperty, ComplexClass 테이블로 구성되어 있다.

Class 테이블에는 OWL 문서에서 정의된 클래스의 이름과 각 클래스를 식별할 수 있는 네임스페이스 정보를 저장한다. Property 테이블에는 OWL 문서에서 정의된 속성의 이름과 각 속성을 식별할 수 있는 네임스페이스를 정보, 각 속성의 도메인과 레인지 정보를 저장

한다. Class나 Property들의 고유한 식별자 역할을 하는 네임스페이스 정보는 Namespace 테이블에 저장한다.

```
<?xml rdf="http://www.w3.org/2001/XMLSchema#" xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#" xmlns:owl="http://www.w3.org/2002/07/owl#" xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#" >
<owl:Class rdf:ID="Wine">
<rdfs:subClassOf>
<owl:Restriction>
<owl:Property rdf:resource="#hasColor">
<owl:cardinality>1</owl:cardinality>
<owl:Restriction>
<rdfs:subClassOf>
<owl:Restriction>
<owl:Property rdf:resource="#hasMaker">
<owl:allValuesFrom rdf:resource="#Winery">
<owl:Restriction>
</rdfs:subClassOf>
...
<owl:ObjectProperty rdf:ID="hasColor">
<rdfs:subPropertyOf rdf:resource="#hasWineDescriptor">
<rdfs:domain rdf:resource="#Wine">
<rdfs:range rdf:resource="#WineColor">
</owl:ObjectProperty>
<owl:ObjectProperty rdf:ID="hasMaker">
<rdfs:subPropertyOf rdf:resource="#hasWineDescriptor">
<rdfs:domain rdf:resource="#Wine">
<rdfs:range rdf:resource="#Winery">
</owl:ObjectProperty>
<owl:ObjectProperty rdf:ID="hasSugar">
<rdfs:subPropertyOf rdf:resource="#hasWineDescriptor">
<rdfs:domain rdf:resource="#Wine">
<rdfs:range rdf:resource="#WineSugar">
</owl:ObjectProperty>
<owl:Class rdf:ID="WineColor">
<rdfs:subClassOf rdf:resource="#WineDescriptor">
<owl:Ontology rdf:base="http://www.w3.org/2002/07/owl#">
<owl:Class rdf:ID="Winery">
<owl:Class rdf:ID="WineSugar">
<rdfs:subClassOf rdf:resource="#WineTaste">
<owl:unionOf rdf:resource="#WhiteTaste">
<owl:unionOf rdf:resource="#Sweet">
<owl:Class rdf:about="#WineDry">
<owl:Class rdf:about="#WineDry">
</owl:unionOf>
</owl:Class>
<Winery rdf:ID="Selasko">
<Winery rdf:ID="White hallLane">
<Winery rdf:ID="WhitehallLane">
<Region rdf:ID="NewZealandRegion">
<Region rdf:ID="Napa">
<IceWine rdf:ID="SelaskoWine">
<IceWine rdf:ID="NewZealandRegion">
<hasMaker rdf:resource="#Selasko">
<hasColor rdf:resource="#WineWhite">
</IceWine>
<CabernetFrane rdf:ID="WhitehallLaneCabernetFrane">
<Shiraz rdf:resource="#WhitehallLane">
<Shiraz rdf:resource="#WhitehallLane">
<hasSugar rdf:resource="#WineDry">
</CabernetFrane>
<DesertWine rdf:ID="WhitehallLanePrimitivo">
<Shiraz rdf:resource="#WhitehallLane">
<hasMaker rdf:resource="#WhitehallLane">
<hasSugar rdf:resource="#Sweet">
</DesertWine>
</Region rdf:ID="Napa">
<owl:complementOf rdf:resource="#NapaRegion">
</Region>
<Winery rdf:ID="White hallLane">
<owl:disjointFrom rdf:resource="#WhitehallLane">
</Winery>
</?xml>
```

<그림 2> OWL을 이용한 와인 온톨로지 문서

각 클래스간의 포함관계는 subclassOf 테이블에, 각 Property의 포함관계는 subPropertyOf 테이블에 저장한다. 또한 자원의 동일성, 이질성 클래스 집합에 대한 표현을 위해 Tag 테이블, EqualClass 테이블, EqualProperty 테이블, 그리고 ComplexClass 테이블을 유기적으로 연관시켜 상관 관계를 표현하였다. EqualClass 테이블은 클래스의 이름은 다르지만 의미는 같은 클래스들의 관계를 저장하고, EqualProperty 테이블은 동일한 의미를 가지는 속성들에 대한 정보를 저장한다. EqualClass과 EqualProperty 테이블을 이용하여 동음이의 또는 이음동의어들에 대해 보다 정확한 결과를 얻을 수 있다. Tag 테이블에서는 각 기능별 속성을 분류하기 위한 코드 정보를 저장한다. 마지막으로 ComplexClass 테이블은 클래스 집합에 대한 표현 방법인 unionOf, intersectionOf, ComplementOf, oneOf 그리고 disjointWith중 사용된 속성을 구분하고, 집합 내 구성 요소인 클래스나 속성들을 저장하여 그들간의 명확한 관계 표현이 가능하도록 한다.

실제로 리소스와 프로퍼티 값으로 표현되는 인스턴스를 위한 테이블은 기존 RDF 관련 연구과 비슷하기 때문에 생략하였다.

<그림 2>는 OWL을 이용하여 와인에 대한 온톨로지를 구성한 문서의 일부분이다. <그림 3>은 <그림 2>의 OWL 문서를 본 논문에서 제안한 저장구조로 저장한 결과를 보여준다. 동치 관계에 대한 부분은 Property테이블이나 Class 테이블에서 데이터에 밀출을 그려 표현했다.

5. 질의 패턴

OWL은 RDF나 RDF 스키마보다 클래스간 또는 속성간의 관계를 보다 명확하게 표현할 수 있다. 특히 클래스나 속성의 동일 관계, 이질 관계, 역관계의 정의 그리고 클래스들의 집합 관계에 대한 표현들은 각 자원들의 의미있는 관계를 보다 정확하게 표현할 수 있도록 한다. 기존의 RDF와 RDF 스키마의 질의어인 RQL에서는 스키마에

대한 질의, 인스턴스에 대한 질의, 스키마와 인스턴스를 함께 고려한 기본적인 질의 형태를 제공하고 있다[4]. 그렇지만 기존의 RQL은 OWL에서 제공하는 확장된 기능들에 대한 질의 형태는 정의하지 않았다. 따라서, 본 논문에서는 기존의 RQL에서 제공하는 기본적인 질의 형태에 OWL이 추가적으로 제공하는 기능에 대한 질의 형태를 보강하고, 기존 RQL을 이용해 표현할 수 있도록 RQL을 확장한다.

<Class>				<Property>				<EqualityProperty>		
cid	ns	name	pid	ns	name	domain	range	pid	tag	target
1	2	Wine	1	2	hasColor	1	2	4	5	6
2	2	WineColor	2	2	hasMaker	1	6			
3	2	WineDescriptor	3	2	hasSugar	1	4			
4	2	WineSugar	4	2	locatedIn	1	7	<EqualityClass>		
5	2	WineTaste	5	2	hasWineDescriptor	1	3	cid	tag	target
6	1	Winery	6	2	placeln	1	7	31	3	32
7	1	Region	7	3	rdftype	6	17	18	4	19
8	1	White	8	3	rdftype	6	18			
9	1	Red	9	3	rdftype	6	19	<SubClassOf>		
10	1	Rod	10	3	rdftype	7	20	cid	superClass	
11	1	Sweet	11	3	rdftype	7	21	2	3	
12	1	ODDry	12	3	rdftype	7	22	5	3	
13	1	Dry	13	3	rdftype	14	23			
14	1	IceWine	14	3	rdftype	15	24	<ComplexClass>		
15	1	CabernetFranc	15	3	rdftype	26	25	cid	tag	T_cid
16	1	DeerWine	16	3	rdftype	23	26	2	1	9
17	1	Sektis	13	1	locatedIn	23	20	2	1	8
18	1	WhiteHallLane	14	1	hasMaker	23	17	2	1	10
19	1	WhiteHallLane	15	1	hasColor	23	8			
20	1	NewZealandRegion	16	3	locatedIn	24	21			
21	1	Napa	17	1	hasMaker	24	18	1	2	5
22	1	NapaRegion	18	1	WineTaste	24	15	3	2	2
23	1	SektisIceWine	19	1	locatedIn	25	22	4	2	11
24	1	WhiteHallLaneCabernetFranc	20	1	hasMaker	25	19	4	2	12
25	1	WhiteHallLanePinotVera	21	1	hasSugar	25	11	4	2	13

<Namespace>		<Tag>		<SubPropertyOf>	
cid	namespace	id	tag	pid	superProperty
1	http://www.example.org/wine#	1	one of	1	5
2	http://www.w3.org/2002/07/owl#	2	unionOf	2	5
3	http://www.w3.org/1999/02/22-rdf-syntax-ns#	3	equivalentClass	3	5
4	http://www.w3.org/2000/01/rdf-schema#	4	differentFrom	4	5
		5	equivalentProperty	4	5

<그림3>OWL문서의 테이블 저장구조

<그림 4>는 기본적인 OWL의 질의 형태와 기존 RDF 저장 구조에서는 가능하지 않았던 OWL만의 특성에 관한 새로운 질의 형태를 분류하고 이를 기존의 RQL을 이용하여 표현한 예이다[4].

질의	질의문 대역과 RQL표현	질의 표현
Q1	hasColor 속성의 Range를 찾자 Select {SC} From {SC}hasColor({SC})	- 속성의 Range 검색
Q2	Winery 클래스의 인스턴스 모든 속성을 찾자 Select {cp, range}({cp}) From {SC}@p Where {SC} = "Winery"	- 클래스의 인스턴스 속성의 검색
Q3	Wine 클래스의 모든 subclass 검색하라 subClassOf(Wine)	- 하위클래스 검색
Q4	Winery 클래스의 하위클래스의 숫자는? Select count(X) From {SC}subClassOf(X) Where {SC} like "Winery" and exist equality Y include range(differentFrom)	- 하위클래스의 수 계산 - 이질성/타당 검색
Q5	Wine의 색이 흰색인 Wine를 찾자 Select X, Y From {X}hasColor(Y) Where Y like "White"	- 조건에 일치한 클래스 검색
Q6	Napa2는 지역에서 생산되는 Wine를 찾자 Select X From {X}hasColor({SC}) Where {SC} like "Napa" and exist equality {SC} include range(equivalentClass) Wine의 알도가 Dry한 WhiteHallLane 지역에서 생산되는 Wine를 찾자 Select X, Y From {X}hasSugar(X)hasMaker(Y) Where X like "Dry" and Y like "WhiteHallLane" and exist equality Y include range(differentFrom)	- 조건에 만족하는 다양 검색 - 동일성/타당 검색
Q7	hasMaker 속성의 Range가 Sektis인 Class는? Select {C} From {hasColor}({C}) Where {C} like "sektis"	- 조건에 만족하는 다양 검색 - 이질성/타당 검색
Q8	흰색 Wine 가지고 있는 속성의 Domain 구하라 Select X From {X}hasColor({SC}) Where {SC} = "White"	- 조건에 만족하는 속성의 Domain 검색
Q9	Napa 지역에서 생산되는 Wine의 인스턴스 모든 속성을 찾자 Select {cp, range}({cp}) From locatedIn(X) {X}@p Where X like "Napa" and exist equality {X} include range(equivalentClass)	- 조건에 만족하는 클래스의 인스턴스 모든 속성의 검색 - 동일성/타당 검색

<그림 4> OWL문서에 대한 질의 패턴과 RQL 표현

Q1, Q2, Q3, Q5, Q8, Q9의 질의문은 기존에 RDF, RDF 스키마 기반에의 질의에서도 가능한 질의들이다. 그렇지만 Q4, Q6, Q7, Q10는 RQL에서 지원하지 않는 새로운 형태의 질의이다.

Q4의 질의문은 Winery Class의 하위 클래스를 검색하는 질의문이다. 직관적으로 보기에는 2개의 Winery를 포함하는 것처럼 보인

다. 하지만 <그림 2>의 OWL 문서에서 WhiteHallLane라는 양조장은 differentFrom을 사용하여, 이름은 비슷하지만 다른 양조장이라는 정보를 제공하고 있다. 따라서 기존의 저장 구조와 RQL 질의 형태로는 differentFrom 속성의 특징을 지원하지 못하기 때문에 이와 같은 새로운 질의에 대한 정확한 결과를 얻을 수 없다[4].

Q6는 Napa지방에서 생산되는 와인을 찾는 질의문이다. <그림 2> 문서에서 Napa와 NapaRegion 지역은 동일한 지역임을 설명하고 있다. 따라서 Napa 지역에 대한 질의 시 NapaRegion지역의 와인도 질의 결과에 포함되어야 한다. Q7 역시 Q4와 마찬가지로 클래스의 이질적인 부분을 고려해야 하므로 기존에 RQL 질의 형태로는 표현할 수 없다. Q10번의 경우도 Q6와 마찬가지로 클래스간의 동일성 여부를 체크해야 하므로 기존 RQL의 질의 형태로는 표현할 수 없다.

본 논문에서는 Q4, Q6, Q7, Q10과 같은 질의를 지원하기 위하여 RQL에 새로운 "exist equality"라는 기능을 추가한다. 만약 동일한 의미의 클래스나 속성이 존재하면 질의 처리 시 고려하여 보다 정확한 질의 결과를 얻을 수 있도록 한다. 또한 클래스나 속성의 이름은 비슷하지만 다른 의미를 가지고 있는 요소들은 제거하여 정확한 질의 결과를 유도할 수 있다. Q4, Q6, Q7, Q10에서 "exist equality"를 이용해 확장된 RQL의 예를 볼 수 있다.

6. 결론

OWL은 RDF나 RDF 스키마보다 클래스간 또는 속성간의 관계를 보다 명확하게 표현하기 위해 동치 관계, 이질 관계, 역관계, 클래스들간의 집합 관계 표현 등의 다양한 기능을 제공하고 있다. 따라서 본 논문에서는 OWL로 표현된 문서에 쉽게 접근하고 질의 처리를 하기 위하여 기존의 RDF나 RDF 스키마로부터 동일 관계, 이질 관계, 클래스들의 집합 관계에 대한 기능 등이 추가된 OWL 문서를 관계형 데이터베이스에 저장하기 위한 구조를 제안한다. 또한 사용자가 보다 정확하게 OWL 문서의 내용에 대해 질의할 수 있도록 기존의 RDF와 RDF 스키마에 대한 질의 형태를 보완하고 추가 질의 형태를 제안한다. 향후에는 본 논문에서 제안한 저장 구조와 질의 형태를 이용하여 OWL 문서의 저장 모듈과 사용자 인터페이스, 질의 처리 모듈로 구성된 OWL 저장 시스템을 설계하고 실제로 구축하는 연구가 필요하다.

[참고문헌]

- [1] W3C, OWL Web Ontology Language Overview, <http://www.W3.org/TR/owl-features/>
- [2] W3C, RDF vocabulary Description Language 1.0 : RDF Schema, <http://www.w3.org/TR/rdf-schema/>
- [3] W3C, RDF Primer, <http://www.W3.org/TR/2003/WD-rdf-primer-20030123>
- [4] G. Karvounarakis, A. Magganarakis, S. Alexaki, V. Christophides, D. Plexousakis, Michel Scholl, Karsten Tolle, "Querying the Semantic Web with RQL", Computer Network, Vol.42, No. 5, 2003.
- [5] Jeen Broekstra, Arjohn Kampman, Frank van Harmelen, "Sesame : A Generic Architecture for Storing and Querying RDF and RDF Schema", International Semantic Web Conference, P54 ~ P68, 2002
- [6] Sofia Alexaki, Vassilis Christophides, Gregory Karvounarakis, Dimitris Plexousakis, and Karsten Tolle, "The ICS-FORTH RDFSuite: Managing Voluminous RDF Description Bases", SemWeb 2001.
- [7] 오삼균, "Web Ontology Language와 그 활용에 관한 고찰", 데이터베이스 연구회 논문집, 18권 3호, P63 ~ P79, 2002.