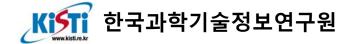
OntoReasoner[®]: 대용량 온톨로 지 저장 및 추론 시스템

이승우 (KISTI)

김재한 (KISTI)

성원경 (KISTI)

한선화 (KISTI)



머 리 말

목 차

1.	서론		1
	1.1.		
	1.2.		
	1.3.	OntoReasoner® 구조	4
2.	대용량	온톨로지 저장소	7
	2.1.	DB 테이블 스키마 구성	8
	2.2.	대용량 트리플의 효율적인 저장	13
3.	효율적인	인 추론	19
	3.1.	뷰를 이용한 효율적인 추론	20
	3.2.	하이브리드(Hybrid) Rete 추론	25
4.	질의 처	리	31
	4.1.	SPARQL 질의 처리 과정	31
	4.2.	효율적인 질의 처리	34
	4.3.	확장 클래스-속성 뷰 (ECPV)	36
	4.3.	1. 정의	37
	4.3.	2. 생성	39
	4.3.	3. 효과	42
5.	평가		45
6.	결론		48
[감·	수 의견]		
[참.	고 문헌]		

[부록]

1. 서론

1990년대 말에 Tim Berners-Lee가 시맨틱 웹 (Semantic Web)을 제안한 이후, 유행처럼 번지기 시작한 시맨틱 웹에 관한 연구는 온톨로지(Ontology)와 추론에 관한 연구로 점차 세분화되고 범위도 넓혀왔다. 온톨로지를 기술하기 위한 언어로, W3C (http://www.w3.org)를 중심으로 메타데이터를 기술하는 목적으로 사용되던 RDF (Resource Description Framework)와 웹을 통한 자유로운 문서의 교환을 목적을 지향하는 XML을 결합하고, 거기에 의미(semantics)를 추가한 RDFS (RDF Semantics)와 이를 다시 확장한 OWL (Web Ontology Language)로 발전해 왔다. OWL은 기술 논리 (Description Logic)에 기반하여 온톨로지의의미를 기술하고 있으며, 여기에 규칙 기술 능력을 부가하여 SWRL (Semantic Web Rule Language)로 발전해 가고 있다.

이러한 시맨틱 웹의 연구 방향은 크게 세 부류로 나눠 볼 수 있다. 그 첫 번째는 온톨로지에 대한 연구로, 인간의 지식을 온톨로지로 어떻게 설계하고 구축하고 활용할 것인가에 대한 연구이다. 최근에는 연구실 수준을 벗어나 실용적인 측면에서 활용될수 있는 온톨로지를 구축하기 위한 방법으로 자동 혹은 반자동방식의 온톨로지 구축에 관한 연구가 활발히 진행되고 있다. 두

번째는 추론에 관한 연구로, 모든 지식을 온톨로지로 일일이 기술하는 데에는 한계가 있으므로, 추론 기술을 이용하여 최소한으로 기술된 온톨로지 지식을 바탕으로 지능적으로 지식을 확장하는 메커니즘에 관한 연구이다. 세 번째는 시맨틱 웹 서비스에 관한 연구로, 온톨로지와 추론을 활용하여 사용자들에게 기존의 서비스와는 차별화된 서비스를 설계하고 보여주는 데 관한 것이다. 이 책에서는 두 번째에 해당하는 온톨로지의 추론에 관해 좀 더깊이 있게 다뤄 보고자 한다. 특히, 온톨로지를 추론하는 것은 필연적으로 온톨로지를 저장하는 과정을 수반하는데, 이 책에서는 온톨로지의 저장과 추론을 위한 시스템인 OntoReasoner 시스템의 기술적인 부분을 중심으로 기술하고자 한다.

시맨틱 웹의 발전 과정에서 수많은 다양한 시맨틱 웹 관련 시스템들이 제 각각의 특성을 갖고 개발되고 확장되어 왔다. 온톨로지 또한 초기에는 장난감 수준의 소규모 온톨로지 (예를 들면, Wine Ontology)에 지나지 않았으나, 최근에는 여러 기관들에서 대용량 온톨로지들이 개발 및 구축되고 실제 서비스에도 활용되는 사례가 속속 발견되고 있다. 이러한 추세에서 온톨로지를 저장하고 추론하는 시스템의 규모성(Scalability)과 실용성 (Practicality)에 관한 연구가 최근 전면으로 부상하고 있다. 이는 최근에 독일 칼스루에(Karlsruhe)에서 개최된 ISWC2008의 시맨

틱 웹 챌린지 (Semantic Web Challenge)에 BTC (Billion Triple Challenge) 트랙이 새롭게 추가된 사실로도 확인할 수 있다.

1.1. 규모성 (Scalability)

시맨틱 웹의 발전 초기에 온톨로지 규모는 작게는 수만에서 많아야 수백만 트리플에 지나지 않았다. 이 정도의 작은 규모의 온톨로지를 대상으로 하는 시맨틱 웹 서비스 시스템들은 모든 온톨로지를 메모리에 올려서 처리하는 것도 가능했기 때문에 규모성 문제를 신경쓸 필요가 없었다. 그러나 현재는 수억에서 수십억 트리플의 시대로 온톨로지의 규모가 방대해졌으며, 머지 않아기존 웹의 규모로까지 확대될 전망이다.

이와 같은 상황에서 시맨틱 웹 시스템, 특히 온톨로지 저장 및 추론을 위한 시스템의 규모성 문제는 더 이상 뒤로 미룰 수 없는 중요한 사안이 되었다. 이에 OntoReasoner는 대용량의 온 톨로지를 저장하고 추론할 수 있는 규모성 측면에 초점을 맞춰 개발되고 있다.

1.2. 실용성 (Practicality)

대용량의 온톨로지를 처리하는 데 있어서, 규모성과 함께 필연적으로 해결해야 할 문제가 실용성이다. 온톨로지 저장 및 추론 시스템이 아무리 대용량의 온톨로지를 처리할 수 있다 하더라도, 실용성 측면을 무시하고는 실제 응용에 적용되어 사용될 여지는 적다. 기존의 정보 검색 시스템이 수많은 사용자를 확보할수 있었던 것은 정보의 홍수인 웹의 엄청난 데이터를 빠른 시간에 점증적으로 색인하고 수많은 사용자의 질의를 짧은 시간 내에처리하여 응답할 수 있었던 규모성과 실용성 측면에서의 능력을 발판으로 하고 있었기 때문이다. 시맨틱 웹 응용 시스템들도 이와 같은 맥락에서 볼 때, 웹에 근접한 수준의 대용량 온톨로지를 빠른 시간에 저장하고 추론하여, 사용자의 질의에 짧은 시간 내에응답할 수 있는 실용성을 갖춰야만 사용자들의 사랑을 받는 시스템으로 발전할 수 있다고 믿는다.

1.3. OntoReasoner® 구조

이 책에서 설명할 *OntoReasonel*® 시스템의 전체 구조는 그림1과 같다. 입력으로 RDF/XML이나 N-Triples, Turtle과 같은 다

양한 형식의 온톨로지를 받아서 파싱(parsing)을 통해 주어 (subject), 술어(predicate), 목적어(object)로 구성된 RDF 트리플 (triple)로 해석하여 대용량 처리가 가능하도록 DBMS 기반의 트리플 저장소 (triple store)에 미리 정의된 DB 테이블 스키마에 맞게 저장된다.

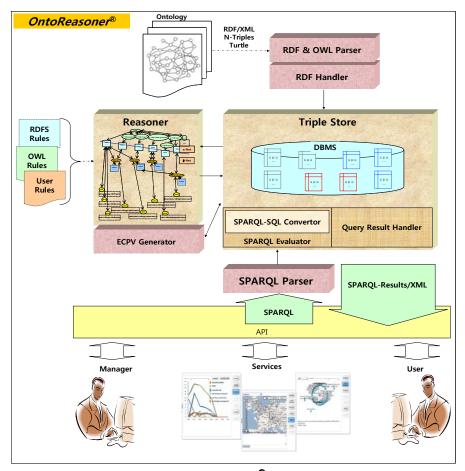


그림 1: OntoReasoner® 시스템 구조

이렇게 저장된 RDF 트리플에, 메모리 및 DBMS를 기반으로 하는 하이브리드 Rete 추론을 통해, RDFS (RDF Semantics)와 OWL Semantics에서 정의하고 있는 추론 규칙과 사용자 정의 추론 규칙을 적용하여 새로운 확장된 RDF 트리플을 생성하고 다시트리플 저장소에 저장한다.

사용자 인터페이스로 시맨틱 웹 질의 언어인 SPARQL을 입력 받을 수 있고, DBMS에 저장된 트리플로부터 응답을 얻기 위해 SPARQL은 SQL로 변환되는 과정을 거쳐 실행된다. 질의 결과는 SPARQL Results/XML 형식으로 사용자에게 돌려진다.

2. 대용량 온톨로지 저장소

온톨로지 저장 및 추론 시스템인 OntoReasoner®의 개발에 있어서 가장 중점을 둔 것은 규모성(scalability)과 실용성 (practicality)이다. 최근의 시맨틱 웹의 발전방향에 맞춰 수십억 이상의 RDF 트리플을 효율적으로 저장하고 질의할 수 있는 능력 이 필요하기 때문에, RDF 트리플 저장을 위해 특화된 저장소를 별도로 개발하기 보다는 수십 년간 대용량의 데이터 처리를 위해 개발되어 발전되어 온 기존의 DBMS를 백엔드(back-end) 저장 소로 사용하기로 결정하였다. 시맨틱 웹의 발전 초기에 작은 용 량의 온톨로지만을 다룰 경우에는 트리플을 모두 메모리에 유지 하는 방식이 가능했지만, 수십억 이상으로 증가한 온톨로지 지식 을 더 이상 메모리에 유지하는 것을 가능하지 않다. 이에 물리적 인 디스크 공간에 저장할 수 밖에 없게 되었는데, 별도의 저장소 를 개발하여 만족스러운 성능을 얻기 위해서는 많은 노력과 비용 이 들어갈 것이 분명한데, 그러기 보다는 그러한 노력과 비용을 보다 효율적인 추론 알고리즘의 개발과 앞선 시맨틱 웹 응용 서 비스를 개발하는 데 투자하는 것이 이득이라고 판단했기 때문이 다. 또한, DBMS는 이미 많은 테스트와 사용 과정을 통해 대용량 의 데이터를 처리할 수 있는 능력을 검증받은 상태이며, 이를 이 용하는 것은 우리의 목표 중의 하나인 규모성의 기반을 이미 확 보한 것이기 때문이다.

2.1. DB 테이블 스키마 구성

온톨로지 지식의 기본 단위는 주어(subject), 술어(predicate), 목적어(object)로 구성되는 트리플(triple)이다. 이러한 트리플을 DB 테이블로 저장하기 위해서는 우선 효율적인 스키마(schema) 를 정의할 필요가 있다.

이에는 크게 세 가지 방법이 가능할 것이다. 한 가지는 가장 단순한 방식으로 모든 트리플을 주어, 술어, 목적어 컬럼을 갖는 하나의 테이블에 저장하는 구조로 온톨로지 스키마에 무관 (schema-oblivious)한 방식(그림2)이라 할 수 있다. 이 방식은, 수직 방식 (vertical schema)이라고 불리는데, 단순함이 장점이지만, 모든 트리플을 하나의 테이블에 저장하는 것은 효율성 측면에서 바람직하지 않다. 왜냐하면, 테이블에 하나의 트리플(레코드)를 저장하거나 읽을 때, 색인(index)까지 고려하면, 모든 트리플과의 비교(물론, B+Tree 인덱스와 같은 효율 좋은 색인 기법을 사용하기 때문에, 실제로 모든 트리플과 하나하나 비교하는 것을 아닐 테지만)를 거쳐야 하기 때문이다. 더욱이, 추론이나 질의 응

답을 위해서는 복잡한 그래프 패턴을 검색해야 하는데, 이는 엄청난 레코드를 갖는 테이블에 대한 재귀 조인 (self join) 연산을 반복적으로 처리해야 한다는 얘기가 된다. 다시 말해, 상당한 비효율성을 내포하고 있는 셈이다.

Triples

Subject	Predicate	Object

그림 2: 스키마-무관 테이블 구조

Prop	oerty ₁	 Class ₁
Subject Object		Subject
Prop	perty ₂	 Class ₂
Prop Subject	oerty ₂ Object	

그림 3: 스키마-인지 테이블 구조

Properties with range Resource

Subject	Predicate	Object

Class Instances

Subject	Object

Properties with range integer

Subject	Predicate	Object	

그림 4: 하이브리드(hybrid) 테이블 구조

트리플을 저장하기 위한 DB 테이블 구조를 설계하는 두 번째 방식으로, 스키마-무관 방식과 정반대로, RDF 트리플을 그 유형에 따라 세밀하게 분산시키는 방식이 있을 수 있다 (그림3). 이를 스키마-인지(schema-aware) 방식이라 부르는데, 다른 표현으로, 수평 방식(horizontal schema)라 부르기도 한다. RDF 트리플은 크게 두 가지 유형, 즉, 술어가 rdf:type인 클래스 인스턴스(class instance)를 표현하는 트리플과 그 외의 트리플로 나눠 볼수 있다. 클래스 인스턴스 트리플은 목적어인 클래스 이름 별로테이블을 구분하여 트리플을 나눠 저장할 수 있으며, 그 외의 트리플 또한, 술어인 속성(property) 별로 테이블을 구분하여 트리플을 나눠 저장할 수 있으며, 그 외의 트리플 나눠 저장할 수 있다. 이렇게 함으로써 대용량의 RDF 트리플을 여러 개의 테이블에 효과적으로 분산시켜 저장할 수 있는 것이다. 또한 이렇게 함으로써 그래프 패턴을 검색할 때, 필요한테이블만을 대상으로 조인 연산을 수행하면 되기 때문에 추론 및질의 처리를 위해서도 효과적이라 할 수 있다.

마지막 세 번째 방식은 스키마-무관 방식과 스키마-인지 방식을 혼용한 방식으로 트리플의 분산을 중간 수준에서 정한 것이라 할 수 있다. 그림4는 그 한 예를 보여준다.

주어나 술어, 목적어에 해당하는 URI를 테이블에 어떻게 표현하느냐에 따라서도 두 가지로 나눠 볼 수 있다. 먼저, URI를 그대

로 저장하는 방식이 있는데, URI는 긴 문자열로 구성되기 때문에, 같은 URI가 여러 트리플에 반복적으로 나타날 수 있기 때문에, 긴 URI 문자열이 테이블에 반복적으로 저장됨으로 인해, 저장공간을 낭비할 수 있을 뿐만 아니라, 저장과 검색의 효율성도 저하시킬 수 있다. 이 때문에, 사전 인코딩 (dictionary-encoding)을통해 긴 URI를 정수 형태의 ID로 사상시켜 ID로 구성된 트리플을저장하는 방식을 사용하기도 한다. 대신 URI와 ID 사이의 매핑을당은 사전(dictionary) 테이블이 추가적으로 요구되며, 질의 응답을 처리할 때 사용자에게 최종적으로 URI를 보여주기 위해서는 ID를 다시 URI로 매핑하기 위해 한 번의 조인을 더 수행해야 하는 단점을 갖는다. 트리플의 수가 증가할수록 URI의 수도 증가하기 때문에 수십억 트리플을 대상으로 하는 경우에는 URI-ID 매핑을 메모리에 유지하는 것은 불가능해진다.

이상에서 살펴 본 바를 바탕으로, *OntoReasonel*®에서는 트리플을 저장하기 위한 DB 테이블 스키마로 스키마-인지 방식을 채택하고 있다(그림5). 다만, 데이터타입 속성(datatype property)의경우, 목적어에 리터럴(literal)이 오는데, 이는 레이블(label)과 데이터타입(datatype), 언어(lagnauge)로 나눌 수 있기 때문에 이를고려한 DB 테이블 스키마를 설계하였다. URI를 표현하는데 있어서는 URI 방식이나 ID 방식이 아닌, URI의 네임스페이스

(namespace)를 접두사(prefix)로 대체한 prefixed URI 방식을 고 안하였다. 접두사와 네임스페이스 사이의 매핑을 위한 테이블 (namespaces)을 별도로 두고 있으며, 하나의 온톨로지에서 사용 하는 네임스페이스의 수는 그리 많지 않기 때문에 이 매핑은 쉽 게 메모리에 유지하면서 매핑을 빠르게 처리할 수 있다. 또한, prefixed URI는 원래의 URI에 비해 문자열 길이를 상당히 줄여서 저장공간의 낭비도 상당히 줄일 수 있다. 그림5에서 tablenames 는 트리플 테이블들의 메타 정보를 기록하기 위한 테이블이다.

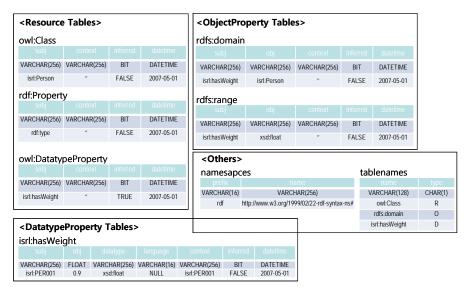


그림 5: OntoReasoner[®]의 DB 테이블 스키마 설계

효과적인 트리플 검색을 위해서는 각 테이블에 색인을 추가하

는 것이 중요하다. 클래스 인스턴스 테이블 (혹은 리소스 테이블)에서는 주어(subject) 컬럼에 대한 색인으로 충분하며, 객체 속성 (object property) 및 데이터타입 속성(datatype property) 테이블에서는 주어(subject)-목적어(object) 색인과 함께 목적어-주어색인을 두었다.

2.2. 대용량 트리플의 효율적인 저장

OntoReasoner®는 자바(Java)로 구현된 시스템으로 DB에 대한 작업을 위해 JDBC를 사용한다. RDF 트리플을 DB 테이블에 저장하는 것은 JDBC를 통해 SQL의 INSERT문을 사용함으로써 가능하다. 그러나, 몇 개의 RDF 트리플을 저장하는 것으로 끝나는 것이 아니라, 수십억 트리플을 빠르게 저장하기 위해서는 몇가지 고려해야 할 점들이 있다.

첫째, RDF 트리플의 삽입(insert) 과정들에 대한 트랜잭션 처리이다. JDBC를 통해 실행하게 되는 SQL의 문장들은 기본적으로 하나하나가 하나의 트랜잭션으로 처리된다 (이를 자동 커밋 (auto-commit) 모드라 한다). 다시 말해, 하나의 문장이 실행될때마다 커밋(commit) 동작이 발생한다. 그러나, 대용량의 트리플을 반복적으로 삽입하는 과정에서 매 트리플 삽입마다 커밋 동작

을 처리하는 것은 상당히 비효율적이며 커밋 동작에만 많은 시간을 낭비하게 된다. 따라서, 일련의 RDF 트리플의 삽입 과정을하나의 트랜잭션으로 처리할 수 있도록, 우선 자동 커밋 모드를해제하고, 일련의 트리플 삽입 문장을 처리한 후 커밋을 수동으로 실행하는 방식을 취해야 한다.

두 번째로, RDF 트리플에 대한 중복 검사를 처리하는 방식이다. 동일한 RDF 트리플을 DB 테이블에 중복으로 저장할 필요는 없으며, 사실 저장해서도 안 된다. 이를 위해서는 RDF 트리플을 저장할 때, 이 트리플이 이미 저장되어 있는지를 확인해야 한다. 이러한 중복 검사는 SQL의 SELECT문을 사용하여 처리할 수 있다. 그러나, 이렇게 처리하는 것은, 삽입할 모든 RDF 트리플에대해, 대상 테이블에서 색인을 두 번 스캔하는 결과를 낳는다. 결국, RDF 트리플의 저장 속도를 저하시키는 원인이 될 수 있다.이를 해결할 수 있는 간단한 방법은 자바의 예외 처리(exception handling)를 이용하는 것이다. 다시 말해, 각 테이블에 중복된 키가 저장될 수 없도록 유일 키(unique key)를 설정해 두고, 저장할 트리플을 JDBC를 통해 INSERT문으로 실행하면, 중복이 발생하는 트리플에 대해서는 SQLException을 발생시키게 된다. 이 때, 이 예외 상황에 대한 오류 코드를 조사하여 중복에 의한 예외 상황인 경우, 해당 트리플은 중복된 트리플이므로 무시하고 다음

트리플 처리로 계속 진행하면 되는 것이다. 이렇게 함으로써, 저장할 트리플에 대한 추가적인 테이블 색인 스캔이 소요되지 않기때문에 훨씬 효율적으로 트리플을 저장할 수 있다.

세 번째로, JDBC를 통해 INSERT 문장을 실행할 때, Statement 대신 PreparedStatement를 사용하는 것이 좋다. PreparedStatement는 저장할 트리플에 대한 내용을 INSERT 문장에서 인자화시켜(parameterized) SQL INSERT문장과 RDF 트리플을 분리함으로써 DBMS에서 SQL INSERT문장을 매번 파싱하고실행계획을 수립하는 과정을 한번으로 줄일 수 있다. 따라서, 동일한 유형의 RDF 트리플을 반복적으로 저장하는 경우, SQL INSERT문에 대한 파싱과 실행계획 수립에 소요되는 시간을 줄임으로써 전체 RDF 트리플의 저장 시간을 단축할 수 있게 된다.

네 번째로, 일괄 처리(batch execution)를 이용하는 것이 좋다.

JDBC에서는 Statement 혹은 PreparedStatement를 실행할 때,
여러 개의 SQL 문장(예를 들면 INSERT문)을 일괄적으로 처리할
수 있는 인터페이스를 제공하고 있다. 이를 이용하면, JDBC로부
터 DBMS로 실행할 여러 개의 SQL문장을 모아서 한꺼번에 전달
하여 실행할 수 있기 때문에 네트워크의 전송 효율과 함께
DBMS에서의 실행 효율도 높일 수 있다. 결과적으로, RDF 트리플의 저장 속도를 높일 수 있게 된다.

표 1: 효율적인 트리플 저장을 위한 방법들의 효용성 실험

일괄처리 개수 ¹	커밋 단위 ²	중복 검사	소요 시간
_	(auto commit)	SELECT	49:28
10000	100	Exception	07:25
20000	100	Exception	07:22
20000	200	Exception	06:53

이상의 네 가지 방법은 실험을 통해 그 효용성을 확인하였다. 먼저, 표1에서는 온톨로지로 LUBM(1000,0)과 DBMS로 MS SQL Server 2005를 사용하여 트리플 저장을 위해 제안한 네 가지 방 법들 중 일괄 처리와 커밋, 중복 검사에 대한 효용성을 확인하기 위한 실험을 수행하였다. 표1에서 알 수 있듯이, 예외 처리 (exception handling)에 의한 중복 검사가 SELECT문을 사용하는 경우에 비해 훨등한 성능 향상을 보여 주고 있으며, 커밋의 경우 도 자동 커밋으로 실행하는 경우에 비해 수동으로 여러 개를 묶 어서 커밋하는 것이 더 낫고 그 묶음의 크기를 크게 할수록 더 낫다는 것을 보여 준다. 또한, 일괄 처리의 경우에도 하지 않는

¹ 일괄적으로 처리하는 RDF 트리플의 수를 가리킨다.

² LUBM은 하나의 학과(Department)를 하나의 OWL 파일로 구분하고 있는데, 커밋의 단위는 바로 이 파일의 개수를 단위로 하였다.

서 수행하는 묶음의 단위를 크게 할수록 효과가 크다는 것을 알 수 있다.

표 2: PreparedStatement와 일괄 처리의 효용성 실험

실험유형 ³	1차(ms)	2차(ms)	3차(ms)	평균(ms)
S	13,204	12,985	13,031	13,073.33
S + B	11,000	10,172	9,328	10,166.67
Р	5,047	4,969	4,922	4,979.33
P + B	2,188	2,188	2,172	2,182.67

다음으로, 표2에서는 PreparedStatement와 일괄 처리(batch execution)의 효용성을 확인하는 실험을 하였다. 여기서는, 두 개의 테이블 (하나는 객체 속성 (object property) 테이블이고, 다른하나는 데이터타입 속성(datatype property) 테이블)에 각각 10,000개의 중복 없는 트리플을 번갈아 저장하는 실험 환경을 가정하였고, 또한 커밋은 모든 트리플이 저장된 후에 마지막에한번만 발생하게 하였으며, 각각 세 번에 걸쳐 반복 실험을 수행하여 평균을 구하였다. 표2에서 알 수 있듯이, Statement를 사용하는 것에 비해 PreparedStatement를 사용하는 것이 월등히 시

³ S는 Statement를, P는 PreparedStatement를 사용하는 경우이며, B는 일괄 처리(batch execution)을 추가로 처리하는 경우를 가리킨다.

간을 단축시킬 수 있으며, Statement를 사용하는 경우와 PreparedStatement를 사용하는 경우 모두에 대해서 일괄 처리 (batch execution)로 실행하는 것이 저장 시간을 훨씬 단축할 수 있다.

3. 효율적인 추론

OntoReasoner®는 온톨로지에 대한 저장과 함께 추론도 수행하는 시스템이다. 온톨로지에 대한 추론은 크게 규칙 기반 추론과 기술 논리(Description Logic) 기반의 추론으로 구분되며, 또한 추론이 일어나는 시점을 기준으로 볼 때, 전방 추론 (forward chaining)과 후방 추론 (backward chaining)으로 나눠 볼 수 있다. 전방 추론은 질의가 들어 오기 이전에 추론이 발생하는 방법으로 오프라인 추론 (off-line reasoning)이라 말할 수 있으며, 후방 추론은 질의가 들어오는 순간에 추론이 일어나므로 온라인 추론 (on-line reasoning)이라 말할 수 있다. 이런 관점에서 볼 때, OntoReasoner®는 규칙 기반으로 전방 추론을 수행하는 시스템이다.

OntoReasoner[®]를 위한 추론 규칙은 Jena의 그것과 유사하다. 이 규칙에서는 사용하는 네임스페이스(namespace)에 대한 접두 사(prefix)를 정의할 수 있고 RDFS 혹은 OWL Semantics에 기반 한 공리(axioms)를 기술할 수 있으며, if-then 규칙 형태로 추론 규칙을 기술할 수 있다. 자세한 추론 규칙 명세에 대한 설명은 [부록]에 기술하였다. 최근의 시맨틱 웹의 발전 방향으로 볼 때, 온톨로지 저장소의 경우와 마찬가지로, 온톨로지 추론에 대해서도 수십억 트리플 이상을 다뤄야 함에 따라 규모성(scalability) 문제와 실용성 (practicality) 문제가 중요하게 되었다. 이런 추세에서 불완전 추론 (incomplete reasoning)과 같이 질의 응답에서 사용될 여지가 거의 없는 트리플들은 추론을 통해 확장할 필요가 적다는 근거하에 필요성이 적은 추론을 생략하여 추론의 효율성을 높이려는 시도가 늘고 있다. RDFS 포함관계에 대한 추론이 그러한 예이다. 그러나 *OntoReasoner*에서는 사용자가 질의할 가능성이 적다하더라도 그러한 추론을 아예 생략하기 보다는 좀더 효율적으로수행할 수 있는 방안을 모색하였다.

3.1. 뷰를 이용한 효율적인 추론

규칙 기반 추론은 명시적으로 주어진 지식으로부터 규칙을 통해 묵시적인 지식을 유도하는 과정이다. 시맨틱 웹에서 지식은 RDF, RDFS, OWL 등으로 기술되는 온톨로지로 표현되며, 이 온톨로지는 RDF 트리플의 집합이므로, 규칙 기반 추론은 명시적으로 주어진 RDF 트리플의 집합에 주어진 규칙을 적용하여 새로운트리플을 유도하는 과정인 셈이다.

DBMS를 활용함으로써 대용량의 트리플을 저장하는 것은 쉽게 달성될 수 있지만, 문제는 추론 과정에서 발생하는 불필요한 중복으로 인한 저장소의 비효율성과 이로 인한 추론 속도의 저하이다. 특히, RDFS 포함관계 함의 규칙인 rdfs7 규칙과 rdfs9 규칙, 그리고 OWL 역관계 규칙의 경우, 규칙이 적용되는 횟수도작고 필요 이상으로 많은 수의 트리플을 생성하게 됨으로써 저장공간을 많이 차지할뿐더러 추론의 효율성도 떨어뜨린다.

rdfs7: <?x,?a,?y> <?a,rdfs:subPropertyOf,?b> \rightarrow <?x,?b,?y> rdfs9: <?x,rdfs:subClassOf,?y> <?a,rdf:type,?x> \rightarrow <?a,rdf:type,?y> inverseOf : < ?a,owl:inverseOf,?b> <?x,?a,?y> \rightarrow <?y,?b,?x>

예를 들어, 그림6은 대학교를 모델링한 온톨로지의 일부로, ex:name 속성은 rdfs:label 속성의 하위 속성이기 때문에 ex:name 속성을 갖는 모든 주어(subject), 목적어(object)는 rdfs:label 속성의 주어 및 목적어로 확장된다. ex:Professor 클래스와 ex:Person 클래스의 경우도, 속성이 아닌 클래스라는 점만다를 뿐 상하위 관계에 의한 확장은 동일하다. 마지막으로, ex:hasMember 속성과 ex:memberOf 속성은 owl:inverseOf 관계에 있기 때문에 ex:hasMember 속성의 모든 주어 및 목적어는

각각 ex:memberOf 속성의 목적어 및 주어로 확장된다. 이러한 확장은 추론 시스템의 질의 응답 측면에서 완전성을 위해 반드시 필요하지만, 트리플의 술어만 대체되거나 혹은 추가로 주어와 목적어만 뒤바뀔 뿐, 여러 트리플 사이의 복잡한 조합을 필요로 하지 않기 때문에 단순하게 트리플을 복제하여 저장하는 것은 트리플 저장 및 추론 시스템의 비효율성을 초래한다.

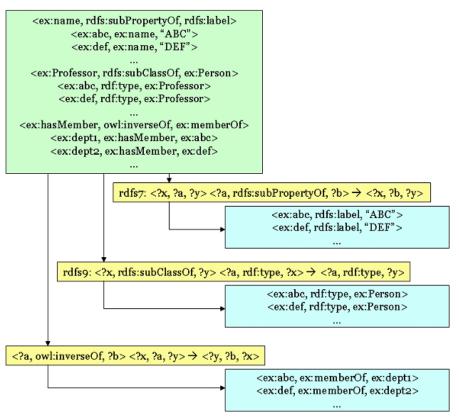


그림 6: 대학교를 모델링한 온톨로지의 일부로 RDFS 포함관계 함의 규칙과 OWL 역관계 규칙을 적용함으로써 추론 확장되는 트리플의 예

결국, 저장 공간과 추론의 효율성을 높이기 위해서는 지나치게 많은 저장 공간과 추론 규칙의 적용을 줄일 수 있는 장치가 필요하며, 물론, 이러한 장치가 RDF 트리플의 저장소에 대한 질의 응답의 효율성을 저해해서는 안 된다.

이에 *OntoReasonel®*에서는 DBMS 기반의 RDF 트리플 저장 및 추론 시스템으로서 RDFS 포함관계 함의 규칙과 OWL 역관계 규칙을 추론함에 있어서 뷰를 이용하여 저장 공간과 함께 추론 속도 측면에서 효율성을 높이는 방안을 고안하였다.

먼저, RDF 트리플이 입력되면, 이 트리플이 RDFS 포함관계함의 규칙이나 OWL 역관계 규칙에 해당하는지를 검사한다. 다시말해, 트리플의 술어(predicate)가 rdfs:subPropertyOf와 rdfs:subClassOf, owl:inverseOf 중의 하나에 해당하는지를 검사한다. 해당될 경우, 그에 따른 뷰를 정의한다. 그림7을 예로 들어설명하면, 화살표 왼쪽과 같이 트리플이 입력되어 저장된 상태에서, 트리플 <ex:name, rdfs:subPropertyOf, rdfs:label>이 입력되는 경우에 술어가 ex:name인 트리플의 주어와 목적어는 동시에 rdfs:label의 주어와 목적어가 될 수 있어야 하는데, 맨 위(a)에서보는 바와 같이 [ex:name] 테이블과 [rdfs:label] 테이블을 합집합(Union)한 것을 [rdfs:label_view]로 정의하고 질의 응답에서는 이 뷰를 사용하도록 함으로써 rdfs:subPropertyOf에 의한 포함관

계 함의 규칙 추론을 대신할 수 있다.

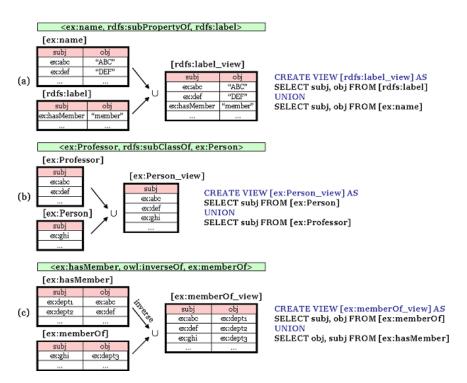


그림 7: RDFS 포함관계 함의 규칙 및 OWL 역관계 규칙의 추론을 대체할 수 있는 DBMS 뷰 정의 예

rdfs:subClassOf에 의한 포함관계 함의 규칙 추론의 경우도 그림7의 가운데(b)에서 보는 바와 같이 [ex:Person] 테이블과 [ex:Professor] 테이블을 합집합(Union)한 것을 [ex:Person_view]로 정의하고 질의 응답에서 이 뷰를 사용하도 록 함으로써 해결된다.

마지막으로, owl:inverseOf에 의한 역관계 규칙 추론의 경우

(c)도, 주어와 목적어를 뒤바꾸는 과정이 추가로 들어가는 것을 제외하면 동일한 방법으로 해결된다. 즉, [ex:memberOf] 테이블과 [ex:hasMember] 테이블을 합집합(Union)할 때, [ex:hasMember] 테이블의 주어 컬럼과 목적어 컬럼을 뒤바꿔서 합집합하는 것이다. 이와 같이 DBMS의 뷰는 물리적으로 저장하는 것이 아닌, 단지 정의만을 갖고 있기 때문에, 추론 확장에 의한 트리플을 저장하는 시간을 단축할 수 있을 뿐만 아니라 저장 공간의 효율성도 추구할 수 있다.

3.2. 하이브리드(Hybrid) Rete 추론

DBMS 뷰 정의로 대체할 수 있는 RDFS 포함관계 함의 규칙과 OWL 역관계 규칙 이외의 추론 규칙에 대해서는 Rete 추론 알고리즘을 적용할 수 있다. Rete 알고리즘은 원래 효율적인 패턴 매칭을 위해 고안된 알고리즘으로 매칭할 패턴을 요소 단위로나누고, 여러 개의 매칭할 패턴들 사이에 공통인 요소들에 대해서는 실제 매칭 동작이 한번만 일어나는 것으로 충분하도록 설계되었다. 한번 매칭된 내용은 메모리에 유지하여 재사용할 수 있게 한 것이다. 이렇게 함으로써 효율적인 패턴 매칭을 달성할 수 있는데, 이 방법의 문제점은 메모리를 많이 사용해야 한다는 점

이다.

패턴 매칭을 위해 고안된 Rete 알고리즘은 규칙 기반 추론에 도 그대로 적용될 수 있는데, 규칙 또한 <주어, 술어, 목적어>를 요소로 하는 그래프 패턴(graph pattern)이기 때문에 트리플에 규칙을 적용하여 확장하는 것은 그래프 패턴에 매칭되는 트리플 집합을 찾는 것과 같은 작업이기 때문이다.

그런데, 대상으로 하는 온톨로지가 소규모일 때에는 규칙에 매칭되는 모든 트리플을 메모리에 유지하는 것이 가능할지 모르지만, 수십억 이상의 대용량 트리플을 대상으로 하는 경우에는 메모리를 많이 사용하는 기존의 Rete 알고리즘을 그대로 사용하는 것은 불가능하다.

추론 규칙

[rdf1: (?x ?a ?y) → (?a rdf:type rdf:Property)]
[rdfs2: (?x ?a ?y) (?a rdfs:domain ?z) → (?x rdf:type ?z)]
[rdfs3: (?x ?a ?y) (?a rdfs:range ?z) → (?y rdf:type ?z)]
[uncle: (?x ex:father ?y) (?y ex:brother ?z) → (?x ex:uncle ?z)]

스키마 트리플

(b) <ex:father, rdfs:domain, ex:Person> <ex:father, rdfs:range, ex:Person> <ex:brother, rdfs:domain, ex:Person> <ex:brother, rdfs:range, ex:Person>

인스턴스 트리플

<ex:p11, ex:father, ex:p10> <ex:p12, ex:father, ex:p10> <ex:p10, ex:brother, ex:p20> <ex:p10, ex:brother, ex:p30>

그림 8: 추론 규칙과 스키마 트리플 및 인스턴스 트리플 예

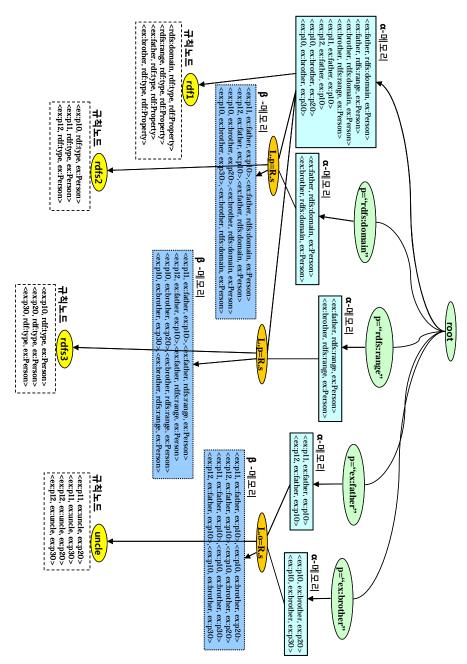


그림 9: 추론 규칙에 대응하는 Rete 네트워크 예

그림8은 추론 규칙과 이 규칙에 적용될 온톨로지(RDF 트리플)의 예를 보여 주며, 그림9는 기존의 Rete 알고리즘을 이용하여 그림8의 온톨로지를 추론 규칙에 적용한 예를 보여 주는데, 기존의 Rete 알고리즘은 α-메모리와 β-메모리의 내용을 모두 메모리에 유지하는 방식으로 입력되는 RDF 트리플이 수십억 이상일 경우, 매칭된 모든 트리플을 메모리에 유지하는 것은 불가능해진다. 결국, 수십억 이상의 RDF 트리플에 대한 추론을 처리하기 위해서는 메모리 사용량을 줄이면서 효율적인 추론을 수행할 장치가 필요해졌다.

이에 *OntoReasoner* 에서는 DBMS를 기반으로 기존의 Rete 알고리즘을 변형하여 효율적인 추론을 수행할 수 있는 하이브리 드 Rete 추론 기법을 고안하였다. 다시 말해, 기존의 Rete 알고 리즘이 갖는 단점인 메모리를 과다하게 사용하는 문제점을 개선 하기 위해 추론을 메모리와 함께 DBMS 상에서 병행하여 수행하 도록 개선하였다.

그림8에서 온톨로지(RDF 트리플)을 두 가지, 즉 스키마 트리플(schema triple)과 인스턴스 트리플(instance triple)로 구분하였다. 스키마 트리플은 RDFS 혹은 OWL에서 정의하고 있는 어휘들로 구성된 트리플로 온톨로지의 구조(schema)를 표현하는 반면, 인스턴스 트리플은 온톨로지 스키마에 따른 구체적인 사례를

표현하는 트리플이다. 마찬가지로 추론 규칙 또한 스키마 추론 규칙(그림8 (a)에서 rdf 및 rdfs 추론 규칙들)과 인스턴스 추론 규칙(그림8 (a)에서 uncle 추론 규칙)으로 나눠 볼 수 있는데, 여기서, 스키마 추론 규칙은 대부분 트리플 간의 복잡한 조합을 필요로 하지 않으며, 규칙이 적용되는 횟수는 많을 지라도 이러한 적용으로부터 새롭게 확장되는 트리플의 수는 많지 않기 때문에 기존의 Rete 알고리즘에서와 같이 규칙 패턴에 매칭된 결과를 메모리 상에 유지하는 것이 가능하다.

인스턴스 추론 규칙

(a) | [uncle: (2x ex:father 2y) (2y ex:brother 2z) \Rightarrow (2x ex:uncle 2z)]

α-메모리 대체 뷰 정의

CREATE VIEW alpha mem1 AS
SELECT T.subject AS s, 'ex:father' AS p, T.object AS o

(b) FROM [ex:father] AS T

CREATE VIEW alphamem2 AS SELECT T.subject AS s, 'ex:brother' AS p, T.object AS o FROM [ex:brother] AS T

β-메모리 대체 뷰 정의

(c) CREATE VIEW betamem1 AS SELECT L.s AS c1, L.p AS c2, L.o AS c3, R.s AS c4, R.p AS c5, R.o AS c6 FROM alpha mem1 AS L INNER JOIN alpha mem2 AS R ON L.o=R.s

규칙 노드 대체 뷰 정의

(d) CREATE VIEW rulenode1 AS SELECT c1 AS s, 'ex:uncle' AS p, c6 AS o FROM betamem1

그림 10: Rete 네트워크에서 DBMS 뷰 정의 사용 예

반면, 인스턴스 트리플은 온톨로지의 규모에 비례하여 함께 증가하므로 인스턴스 추론 규칙에 매칭된 결과는 기존의 Rete처럼 메모리에 유지하는 것이 어렵다. 이런 경우에는 기존 Rete의 α-메모리와 β-메모리를 DBMS의 뷰 정의로 대체한다. 그림10은 그림9에서 인스턴스 추론 규칙에 해당하는 경우에 대하여 α-메모리와 β-메모리를 DBMS의 뷰 정의로 대체하는 예를 보여 준다. Rete의 단말노드에 해당하는 규칙 노드의 경우도 마찬가지로 DBMS 뷰로 정의할 수 있다.

4. 질의 처리

OntoReasonel®의 질의언어 형식은 W3C 권고안인 SPARQL 언어를 사용한다. SPARQL 질의어를 사용하여 저장소의 대규모 트리플 정보를 지정된 조건에 맞게 검색을 할 수 있다 OntoReasonel®에서 사용하는 SPARQL 질의언어는 W3C SPARQL 권고안 중 SELECT, CONSTRUCT문을 지원하며 아울러 일반적인 DBMS에서 자주 사용되는 집합(aggregate)관련처리를 위한 문법을 SPARQL의 SELECT문에 추가적으로 정의하여 지원한다.

요청된 SPARQL 질의문은 SPARQL 구문분석기(Parser)와 SPARQL-TO-SQL 변환 과정을 거쳐 최종적으로 DBMS에 의해 실행된다. DBMS의 실행 성능을 높이기 위해 OntoReasoner®에서는 Prepared-statement를 활용하는 기법인 Parameterized-SQL을 사용한다.

4.1. SPARQL 질의 처리 과정

다음 그림11은 *OntoReasoner*® 시스템의 SPARQL-TO-SQL 변환기와 DBMS와의 연관 구성도이다.

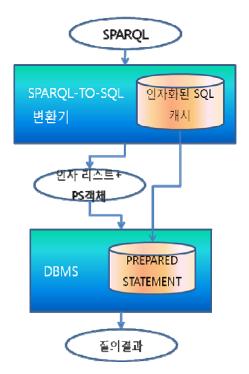


그림 11: SPARQL-SQL 변환기 구성도

시맨틱웹 서비스를 구성하는 과정에서 서비스하려는 요구사항에 맞춰 질의 서비스 설계 단계를 거친다. 이를 통해 SPARQL질의문 셋이 정의된다. 미리 정의된 SPARQL 질의문이 사용자가지정한 검색조건이나 서비스의 요구에 맞게 형태를 갖추어 변환기에 입력되면 먼저 SPARQL-TO-SQL 변환기 내부의 SPARQL구문보석기에 전달된다. 전달된 질의문을 파싱한 결과물은 파싱트리 형태의 내부 정보를 생성하게 되며, 그 이후 구문분석 결과를 이용하여 SQL구문으로 변환하는 과정이 진행된다.

SPARQL 구문분석기는 W3C 권고안에 추가된 독자적인 문법을 지원한다. SELECT 절에 나열되는 변수에 적용할 수 있는 추가적인 함수를 제공하는데 *COUNT(변수), COUNT(DISTINCT 변수), SUM(변수), AVG(변수), MAX(변수), MIN(변수)*의 6종류의 함수가 추가 되었다. 그리고 SELECT 절 이후에 GROUP BY 절을 두어 지정한 변수에 대해 그루핑이 가능한 SQL문을 생성하도록했다.

아래는 추가적인 문법을 적용하여 작성한 SPARQL 질의문 예이다.

4.2. 효율적인 질의 처리

OntoReasoner® 질의 시스템은 효율적인 DBMS 처리를 위해서 PREPARED STATEMENT 기능을 사용한다. PREPARED STATEMENT는 처리하고자 하는 SQL 질의문에서 변동되는 상수값을 '?'심볼로 대체한 질의문을 미리 DBMS로 하여금 분석하게하여 그 결과를 저장해 둔 이후, 실질적인 질의처리 때엔 상수값배열만 전달하여 DBMS의 처리효율을 높이는 기법이다.

일반적으로, 제공하는 서비스의 종류와 개수에 비례하여 질의 문의 수가 결정된다. 서비스의 개수는 정해져 있으므로 이러한 PREPARED STATEMENT 기능을 활용하여 미리 질의문을 분석 해놓고 처리한다면 질의처리 효율을 높일 수 있다.

OntoReasoner® 질의 시스템은 SPARQL 구문분석기를 이용하여 이러한 PREPARED STATEMENT 기능을 자동으로 적용하도록 구현되어 있어 사람에 의해 별도로 PREPARED STATEMENT기능을 구현하기 위한 질의문을 작성할 필요가 없는 특징을 가지고 있다.

SPARQL 구문분석기는 질의문 파싱 과정에서 상수(리터럴)로 사용된 문법적 요소들에 대해 자동으로 '?' 심볼로 대체하여 PREPARED STATEMENT 객체를 생성할 수 있도록 준비한다. 서 비스에 사용되는 다양한 종류의 질의문에 대해 '?' 처리를 하고 나면 그 고유한 질의문 패턴을 구분할 수 있게 되며 '?' 처리가 된 질의문과 이에 대해 DBMS에 의해 생성된 각각의 PREPARED STATEMENT 객체를 인자화된 SQL 캐시라는 저장 공간에 저장 하여 이후 질의문 요청에 대비하게 된다.

인자화된 SQL 캐시에 저장 되지 않은 질의문이 요청될 경우 DBMS에 의해 PREPARED STATEMENT 객체를 생성하여 '?'처리된 SQL 질의문을 고유 식별자로 변환하여 인자화된 SQL 캐시저장소에 저장하고, 캐시에 저장된 것으로 확인된 질의문은 바로 PREPARED STATEMENT 객체를 이용하여 DBMS 질의를 시작한다.

아래 그림12는 PREPARED STATEMENT 기능을 활용하는 방법에 대한 질의문 처리 과정의 예를 보여주고 있다.

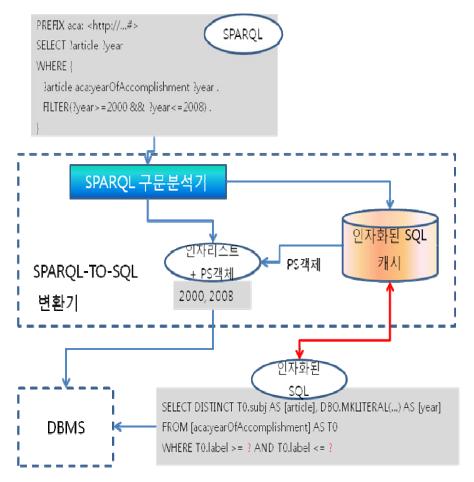


그림 12: Prepared Statement의 활용

4.3. 확장 클래스-속성 뷰 (ECPV)

전통적인 DBMS 시스템에서는 지식(데이터)을 해당 응용에 특화시켜 잘 짜여진 스키마 형태로 저장하여 지식 변경이나 확장에 있어서 유연성이 결여되는 반면 추론 시스템에서는 지식을 트리

플 형태의 작은 단위로 분산시켜 저장하여 응용에 따라 단편적인 지식을 조합하여 새로운 지식으로 확장하기 쉽도록 하는 유연함을 제공한다. 그러나, 이러한 추론 시스템의 유연함은 장점이기도 하지만, 질의를 처리하는 과정에서 매번 단편적인 지식들을 조합해야 함으로써 발생하는 응답속도 문제는 단점이기도 하다. 이를 극복하기 위한 한 가지 방법으로 *OntoReasonel®에서는* 확장 클래스-속성 뷰(Expanded Class-Property View; ECPV)를 고 안하였다.

4.3.1. 정의

확장 클래스-속성 뷰는, 기존의 트리플 형태의 지식은 주된 저장 구조로 유지하면서 질의 처리를 보다 효율적으로 수행하기 위한 보조적인 저장 구조로서, 특정 클래스로부터 직·간접적으로 관계를 갖는 속성 및 속성 값의 쌍들로 구성된 하나의 뷰이다.

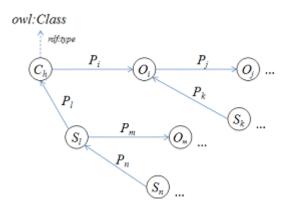


그림 13: 특정 클래스 (C_h) 에 대한 확장 클래스-속성 뷰를 구성하게 될 속성(P)과 속성 값이 속한 클래스(O와 S)

그림13은 특정 클래스 C_h 에 대한 확장 클래스-속성 뷰를 구 성하게 될 속성(P)과 속성 값이 속한 클래스(O와 S)를 보여준다. 실선 화살표의 시작은 속성(P)의 rdfs:domain을 가리키며 끝은 rdfs:range를 가리킨다. owl:Class에 속한 특정 클래스 C_h 를 rdfs:domain으로 갖는 속성(eg. P_i)뿐만 아니라 C_h 를 rdfs:range 로 갖는 속성(eg. Pi)도 확장 클래스-속성 뷰의 대상이 된다. 이 CCH $P_i =$ owl:FunctionalProperty로 제한하며 Pl은 owl:InverseFunctionalProperty로 제한한다. 이는 속성에 대한 대 응수(cardinality) 혹은 역대응수가 1인 경우만을 대상으로 하는 것이며, 1보다 큰 경우 확장 클래스-속성 뷰에 불필요하게 중복 된 데이터가 많이 들어가게 되기 때문이다. P_i 와 P_i 은 클래스 C_h 에 대한 직접적인 관계이며 P_j 와 P_k , P_m , P_n 등은 간접적인 관계

로 모두 확장 클래스-속성 뷰의 구성에 포함될 수 있다.

열이름	subject	P_{i}	P_{i}	INV P _i	INV P _i	P_i	INV P _i	•••		
열형식	URI_TYPE	URI_TYPE 또는 xsdType2SqlType()								

그림 14: 확장 클래스-속성 뷰의 스키마

4.3.2. 생성

확장 클래스-속성 뷰를 자동으로 생성하기 위해서는 우선 이 뷰의 스키마를 자동으로 생성해야 한다. 그림13에 대한 확장 클래스-속성 뷰의 스키마는 그림14와 같이 구성되며 이는 그림15에 기술된 알고리즘에 의해 자동으로 생성될 수 있다. 특정 클래스 C_h 에 속하는 인스턴스를 subject로 갖는다. C_h 로부터 나가는 방향의 속성들은 속성 자체를 열 이름으로 가지며 열 형식은, 속성이 owl:DatatypeProperty에 속하면 해당 XSD datatype에 대응되는 SQL type이 되고 그렇지 않으면 URI에 적합한 SQL type(여기서는 URI_TYPE으로 표현함)이 된다. C_h 로 들어오는 방향의 속성들은 구분을 위해 속성 앞에 'INV'를 붙여 열 이름으로 사용하며 열 형식은 URI_TYPE이 된다. 그림14에서 P_i 와 INV P_i 열은 각각 O_i 와 S_i 을 통해 연결될 수 있는 모든 속성들(P_i 와 P_k , P_m 과 P_n)

이 각각 owl:FunctionalProperty와 owl:InverseFunctionalProperty에 속한다면 생략된다. 그림15의 ECPV_schemaSub()에서 'flag'가 이를 위해 사용된다.

다음 단계로 온톨로지의 인스턴스들 사이의 트리플로부터 확장 클래스-속성 뷰에 데이터를 채운다. 이 과정 또한 그림15와 흡사한 알고리즘을 통해 자동으로 처리될 수 있다.

```
ECPV_schemaMain()
begin

get {<C<sub>i</sub>>} from the result of

"select ?C where {?C rdf:type owl: Class }"

for each <C<sub>i</sub>>
begin
       for each $5,000
begin
if isDatatypeProperty(P) then
column name[n] := R;
column type[n] := xsdType2SqlType(O<sub>i</sub>); n++;
else ECPV_schemaSub(P<sub>i</sub>, O<sub>i</sub>, false);
~ 356
        endif
end
        ena get (< S, P>) from the result of 
"select % P where {PP rdfs: domain % .

P rdfs: range C, .

P rdf type owl: InverseFuntionalProperty }"
        for each \langle S, P \rangle
        begin
ECPV_schemaSub(P<sub>i</sub>, S, true);
         end
ECPV_schemaCreate(table_name,column_name,column_type);
  end
 ECPV_schemaSub(P, O, isInverse)
begin flag := false;
get {<P, Q>} from the result of
"select ?P ?O where {?P rdfs: domain O ...
"P rdfs: range ?O }"
     begin
if !isFunctionalProperty(P) then
        if is runcional roperty(R) then
flag := twe;
else if is Datatype Property(R) then
column_name[n] := P;
column_type[n] := xsdType ZsqType(Q); n++;
else ECPV_schemaSub(R, Q, false);
endif
    end get {<$,R>} from the result of select ?$ ?P where {?P rift:domain ?$.
?P rdfs:range O }"
     for each <&R>
     begin
if !isInverseFunctionalProperty(R) then
        flag := true;
else ECPV_schemaSub(A, S, true);
        endif
    end

if flag=true then

if isinverse=true then column_name[n] := "INV" + P;
        else column_name[n] := P, endif
        column_type[n] := URITYPE; n++;
endif
end
```

그림 15: 확장 클래스-속성 뷰의 스키마 생성 알고리즘

4.3.3. 효과

추론 시스템의 응답 속도에 있어서 확장 클래스-속성 뷰의 효과를 살펴보기 위해 OntoFrame®의 추론 서비스를 대상으로 조사하였다. OntoFrame®은 연구자의 R&D 전주기 과정을 지원하기위한 플랫폼으로, 문헌으로부터 수집한 연구 토픽과 과학기술 온톨로지, 추론 시스템을 사용하여 연구 전문가, 연구자(공저자/인용) 네트워크, 연구성과 맵, 통계 정보, 연구자 정보, 기관 정보, 성과 정보 등의 추론 서비스를 제공한다.

표 3: ECPV의 효과

추론 서비스 유형	조인	비율		
-E ΛΠΞ πδ	w/o ECPV	w/ ECPV	ue	
연구 전문가	8	4	0.5	
공저자 네트워크	13	7	0.538	
인용 네트워크	14	12	0.857	
연구 성과 맵	4	4	1.0	
통계 정보	7	6	0.857	
연구자 정보	35	11	0.314	
기관 정보	31	11	0.355	
성과 정보	12	7	0.583	

이 추론 서비스들을 제공하기 위해 필요한 테이블 조인 횟수

는 표3의 'w/o ECPV' 열에 제시하였다. 예를 들어, 연구 전문가서비스를 위해서는 그림16에서 알 수 있듯이 WHERE절의 OPTIONAL 부분을 제외하고 8개의 트리플 조건이 있으므로 8회의 조인이 필요하다. 그러나 ECPV를 사용하면 그림17에서 알 수 있듯이 5가지의 ECPV 테이블 사이에 4번의 조인으로 처리될 수 있다.

그림 16: 연구 전문가 서비스를 위한 SPARQL

이와 같이 ECPV를 사용할 경우 각 추론 서비스별로 요구되는 조인 횟수는 표3의 'w/ECPV' 열에 표시하였다. 연구 성과 맵 서비스를 제외한 나머지 모든 서비스에서 ECPV를 사용함으로써 0.857 ~ 0.314배만큼 조인 횟수를 줄일 수 있었다.

열 이 때	테이블 이름
subject	ECT.X_Outcomes
hasPublicationYear	ECDV Outcome:
INVhasTopicDomain	ECEN_TobicDomain
hasTopicKeyword	FORT Having
INVhasCategoryDomain	ECEN_CalegoryDomain
hasCategoryKeyword	
INVhasCreationInformation	
hasCreator	ECPV_CreationInformation
contributionWeightOfCreator	
subject	
nameOfPerson	ECPV_Person
INV.sameAs	

그림 17: 연구 전문가 서비스에 필요한 ECPV 테이블 조인⁴

⁴ 두 가지 빗금 무늬는 조인의 기준이 되는 열을 가리킨다. 예를 들어, ECPV_Outcomes와 ECPV_TopicDomain은 각각 subject와 INVhasTopicDomain 열을 기준으로 조인된다.

5. 평가

OntoReasoner®를 평가하기 위한 데이터로는 LUBM을 사용하였다. LUBM은 미국 Lehigh 대학에서 만든 벤치마크 테스트 셋으로, 대학교를 모델링하여 OWL Lite로 온톨로지를 기술하고 있으며, 43개의 클래스와 32개의 속성 (이중 25개는 객체 속성 (object property)이고, 나머지 7개는 데이터타입 속성(datatype property)임)으로 구성되어 있다. 온톨로지의 인스턴스 데이터는 LUBM(n,s)로 표현하여 n과 s를 파마메타로 자동으로 생성되는데,여기서, n은 대학교의 수를 가리키며, s는 seed 번호로 각 대학의 학과나 교직원, 학생 수 등을 결정하는 역할을 한다.

LUBM을 사용한 OntoReasonel®의 평가는 두 가지 관점으로 진행하였다. 하나는 대용량의 온톨로지 트리플에 대한 OntoReasonel®의 규모성과 실용성 측면을 평가하기 위한 것으로 RDF 트리플의 적재 및 추론에 소요되는 시간을 측정하는 것이고, 다른 하나는 사용자 관점에서의 실용성을 평가하기 위해 질의 응답에 소요되는 시간을 측정한 것이다. 이번 평가는 대용량의 온톨로지를 대상으로 OntoReasonel®를 평가하는 것이므로, 대학의 수가 1000인 LUBM(1000,0)과 8000인 LUBM(8000,0)을 사용하였으며, 이는 각각 약 1.3억 트리플과 약 10억 트리플로

구성되어 있다. 질의 응답을 위해서는 LUBM의 14개 질의를 SPARQL로 기술하여 사용하였다. 참고로, 이 질의는 부록으로 첨부하였다.

표 4: 적재 및 추론 소요 시간

	적 재	추론	전체
LUBM(1000)	11h	18h	29h
LUBM(8000)	56h	104h	160h

먼저, LUBM에 대한 OntoReasone/®의 적재 및 추론 소요 시간에 대한 실험 결과는 표4에 나와 있다. 약 1.3억 트리플에 대해 적재 및 추론을 포함하여 30시간 이내에 완료할 수 있었으며,약 10억 트리플에 대해서는 160시간에 적재 및 추론을 완료할수 있었다. 이러한 실험 결과는 아직 만족스러운 결과는 아니나,OntoReasone/®가 규모성 측면에서 대용량 온톨로지를 대상으로처리하는데 있어서 문제가 없음을 보였으며,실용성 측면에서도가능성을 보였다. 또한,현재는 초기 구현 단계로,구현 상의 개선할 점들이 많이 있기 때문에 향후 실용성 측면에서 더 나은 결과를 얻을 수 있을 것으로 기대한다. 또한,다중 쓰레드(multi-thread) 기법을 도입하여 추가로 속도를 개선할 예정이다.

표 5: 질의응답 소요시간(QTPR)5

	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10	Q11	Q12	Q13	Q14	평균
LUBM (1000)	37.6	51.5	8.5	0.0	0.1	0.0	1.5	0.5	2.3	0.0	1.1	10.2	0.2	0.0	8
LUBM (8000)	51.9	71.1	11.8	0.1	0.2	0.1	2.1	0.6	3.2	0.0	1.5	14.1	0.3	0.1	11

다음으로, 질의 응답에 대한 소요 시간에 관해 OntoReasoner®의 성능을 실험한 결과는 표5에 나와 있다. 약 1.3억 트리플에 대해 질의 결과당 8ms, 약 10억 트리플에 대해 서는 질의 결과당 11ms가 소요되는 것으로, 이러한 성능은 지금까지 알려진 다른 시스템의 성능과 비교해도 뒤떨어지지 않는다.

⁵ QTPR은 질의 결과당 소요된 시간을 가리키며 Query Time Per Result의 약자이다. 여기서 측정된 시간은 밀리초(ms)단위이다.

6. 결론

최근의 시맨틱 웹 분야의 발전 방향은 수십억 이상의 온톨로지 트리플을 대상으로 실제 활용 가능한 시맨틱 웹 서비스를 도출하고 구현하는 데 집중되고 있다. 이러한 발전의 근간이 될 수 있는 것이 바로 규모성과 실용성을 갖춘 대용량 온톨로지를 위한적재 및 추론 시스템일 것이다. 이러한 관점에서 *OntoReasonel®*의 설계와 개발이 진행되어 왔으며, 이제 가시적인 결과를 얻는 단계가 되었다.

OntoReasone/®에서는 우선 규모성을 달성하기 위해 새로운 저장소를 자체적으로 개발하기 보다는 이미 안정성과 규모성이 확보된 DBMS 시스템을 백엔드(back-end) 저장소로 활용하는 방식을 사용하였다.

DBMS를 기반으로 대용량의 RDF 트리플을 효율적으로 적재하는 데에는 몇 가지 고려할 사항이 있었다. RDF 트리플에 대한 중복 검사 방식이나 트랜잭션의 관리, Prepared Statement와 일괄 처리의 활용 등이 그것으로, 실험을 통해 이러한 기법들이 대용량의 트리플 적재에 있어서 시간을 상당히 단축시킬 수 있음도확인하였다.

규모성을 위해 DBMS을 백엔드 저장소로 채택하였지만,

DBMS 자체가 추론 기능까지 갖고 있는 것은 아니다. 따라서, OntoReasonel®에서는 추론 기능을 구현하기 위하여, 규칙 기반의 추론 기법을 적용하였다. 우선 사용한 추론 규칙에 대한 명세를 정의하고, 이를 기초로 추론 규칙을 해석하는 모듈까지 개발하였다. 기술된 추론 규칙에 대해 적재된 RDF 트리플을 적재하여 새로운 확장된 트리플을 얻어내는 효율적인 방법을 고안하기위해 기존에 효율적인 패턴 매칭을 위해 고안되었던 Rete 알고리즘을 규칙 추론에 적용하였다. 그러나, 기존의 Rete 알고리즘은 메모리를 과다하게 사용하는 문제점이 있기 때문에 이를 해결하기 위해 추론 과정에 메모리 기반 추론과 DBMS 기반 추론을 병행하는 하이브리드(Hybrid) Rete 추론 기법을 고안하여 적용하였다.

여기에 추가적으로, RDFS 포함관계 함의 규칙과 OWL 역관계 규칙의 경우에 대해서는 추론 규칙의 적용을 DBMS의 뷰 정의로 대체함으로써 추론 규칙 적용에 소요되는 시간뿐만 아니라 확장된 트리플을 다시 적재하는 데에 소요되는 시간까지 줄임으로써 상당한 성능 개선 효과를 얻을 수 있었다.

OntoReasoner[®]의 질의 처리를 위해서는 SPARQL을 입력으로 받아 SQL로 변환하는 과정을 거쳐 SPARQL Results/XML 형태의 응답을 생성할 수 있도록 하였으며, 특히, 통계치를 요구하는 추 론 응용 서비스를 지원하기 위하여 Count, Sum, Min, Max, Avg 등과 같은 Aggregates 기능을 추가하여 SPARQL 스펙을 확장하였다.

질의 응답의 효율성을 위해서는 확장 클래스-속성 뷰(ECPV) 기법을 고안하여 질의를 처리하는 데 소요되는 트리플 테이블 간의 조인 횟수를 줄임으로써 질의에 대한 응답 속도를 개선할 수 있었다.

LUBM(1000)과 LUBM(8000)을 사용한 대용량 RDF 트리플에 대한 실험을 통해 *OntoReasonel®*의 규모성 측면에서는 만족할 만한 결과를 얻었으며, 실용성 측면에서는 만족스러운 정도는 아니지만, 아직 초기 구현된 단계이기 때문에 구현 상에 개선의 여지가 남아 있는 만큼, 향후 충분히 나은 성능을 보일 수 있는 가능성은 보인 셈이다.

[감수 의견]

[참고 문헌]

이승우, 정한민, 성원경, "R-DBMS 기반 추론 서비스인 OntoThink-K에서의 SPARQL 질의 지원", 2006 한국정보과학회 추계 학술발표논문, Vol.33, No.2(B), 2006.

RDF, http://www.w3.org/RDF/

RDF Vocabulary Description Language 1.0: RDF Schema, http://www.w3.org/TR/rdf-schema/

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

OWL Web Ontology Language Semantics and Abstract Syntax, http://www.w3.org/TR/owl-semantics/

Jena2, http://jena.sourceforge.net/

A. Chebotko, S. Lu, H. M. Jamil, F. Fotouhi, "Semantics Preserving SPARQL-to-SQL Query Translation for Optional Graph Patterns", Technocal Report TR-DB-052006-CLJF, May 2006.

Y. Theoharis, V. Christophides, and G. Karvounarakis, "Benchmarking Database Representations of RDF/S Stores", In Proceedings of the 4th International Semantic Web Conference, 2005.

Chong, E.I., S. Das, G. Eadon and J. Srinivasan, "An Efficient SQL-based RDF Querying Scheme", In Proceedings of the 31st VLDB Conference, Trondheim, Norway, 2005.

Ding L., K. Wilkinson, C. Sayers and H. Kuno, "Application-Specific Schema Desing for Storing Large RDF Datasets", In Proceedings of the 1st International Workshop on Practical and Scalable Semantic Systems, Sanibel Island, Florida, 2003.

Y. Guo, Z. Pan and J. Heflin, "LUBM: A Benchmark for OWL

Knowledge Base Systems", Journal of Web Semantics 3(2), 2005.

- O. Erling, "Towards web scale RDF", In Proceedings of the 4th International Workshop on Scalable Semantic Web Knowledge Base Systems (SSWS2008), Germany, 2008.
- A. Hogan, A. Harth and A. Polleres, "Scalable Authoritative OWL Reasoning on a Billion Triples", ISWC2008 Billion Triples Challenge, Germany, 2008.
- C. Forgy, "Rete: A Fast Algorithm for the Many Pattern/Many Object Pattern Match Problem", Artificial Intelligence, 19, 1982.

[부록]

[OntoReasoner[®]를 위한 추론 규칙 기술 명세]

1. 개요

OntoReasoner에서 사용할 목적으로 추론 규칙을 기술하는 형식을 정의한다. 이 규칙을 기술하는 언어를 *OntoReasoner Rule Language* (*ORL*)이라 부르기로 한다. ORL은 많은 부분에 Jena 규칙을 참조하였기때문에 많은 부분에 있어서 유사하다.

2. MIME 및 인코딩

지원하는 MIME type은 text/plain이며 인코딩은 US-ASCII를 기본으로 한다.

3. 구성

ORL은 크게 세 부분으로 구성되며 prefix 선언, 공리(axiom), 규칙이 그것이다. 각각은 한 행에 하나씩만 있을 수 있다.

prefix 선언부는 namespace에 대한 prefix를 선언하는 부분으로 예를 들면, 다음과 같다.

@prefix rdf: <a href="mailto:r

다음으로, 공리(axiom)는 조건절이 없는 규칙과 유사하며, 항상 사실인 트리플을 표현하며, 예를 들면, 다음과 같다.

-> (rdf:type rdf:type rdf:Property)

마지막으로 규칙은 조건절(body)과 결론절(head)로 구성되며 맨 앞에는

규칙의 이름이 온다. 조건절에는 하나 이상의 트리플 패턴이, 결론절에는 하나의 트리플 패턴이 올 수 있다. 여기서, 트리플 패턴은 (주어, 술어, 목적어)로 구성된 트리플에서 주어 혹은 술어, 목적어가 변수 (variable)일 수 있는 경우를 말한다. 규칙의 예는 다음과 같다.

```
[rdf1: (?x ?a ?y) -> (?a rdf:type rdf:Property)]
[rdfs7: (?x ?a ?y) (?a rdfs:subPropertyOf ?b) -> (?x ?b ?y)]
```

주석(comment)은 '#'으로 시작한다.

4. E-BNF Grammar

ORL은 BNF로 아래와 같이 표현될 수 있다.

```
<rules> ::= <line>*
< ::= <ws>* (<comment> | <prefix_decl_or_axiom_or_if_then_rule>)?
<eoln>
<comment> ::= '#' (<character> - (<cr> | <lf>))*
<prefix_decl_or_axiom_or_if_then_rule> ::= <prefix_decl> | <axiom> |
<if_then_rule>
<prefix_decl> ::= '@prefix' <ws>+ <prefix> ':' <ws>+ '<' <absoluteURI> '>'
<axiom> ::= '->' <ws>* <triple> <ws>*
<triple> ::= '(' <ws>* <subj> <ws>+ <pred> <ws>+ <obj> <ws>* ')'
<subj> ::= <uriref>
<obj> ::= <uriref> | literal>
<uriref> ::= '<' <absoluteURI> '>' | <prefixedURI>
coalName
::= <langString> | <datatypeString>
<langString> ::= "" <string> "" ('@' <language>)?
<datatypeString> ::= "" <string> "" '^^' <uriref>
```

```
<language> ::= [a-z]+ ('-' [a-z0-9]+)*
<if_then_rule> ::= '[' <ruleName> <ruleBody> '->' <ruleHead> ']'
<ruleName> ::= <ws>* <ruleid> ':' <ws>+
<ruleBody> ::= <triple_pattern_list> <ws>*
<ruleHead> ::= <ws>* <triple_pattern> <ws>*
<ruleid> ::= [a-zA-Z]+ ('-' [a-zA-Z0-9]+)*
<triple pattern list> ::= <triple pattern> <ws>* <triple pattern list>
<triple_pattern> ::= '(' <ws>* <subjVar> <ws>+ <predVar> <ws>+ <objVar>
<ws>* ')'
<subjVar> ::= <subj> | <variable>
<predVar> ::= <pred> | <variable>
<objVar> ::= <obj> | <variable>
<variable> ::= '?' [a-zA-Z][a-zA-Z0-9]*
<localName> := <NCName>
<absoluteURI> ::= <URI-reference>
<ws> ::= ' ' | '\t' /* space or tab */
<eoIn> ::= '\r' | '\f' | '\r\f'
<character> ::= [#x20-#x7E]
<NCName>ଖ
                     대해서는
                                    http://www.w3.org/TR/REC-xml-
names/#NT-NCName 참조하기 바라며, <URI-reference>에 대해서는
http://www.w3.org/TR/2004/REC-rdf-concepts-20040210/#dfn-
```

5. 예시

다음은 ORL로 기술된 규칙의 예시이다.

URI-reference를 참조하기 바란다.

@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#>

- @prefix xsd: <http://www.w3.org/2001/XMLSchema#>
- @prefix owl: "> http://www.w3.org/2002/07/owl#>">
- # RDF Axioms (RDF Semantics, section 3.1)
- -> (rdf:type rdf:type rdf:Property)
- -> (rdf:subject rdf:type rdf:Property)
- -> (rdf:predicate rdf:type rdf:Property)
- -> (rdf:object rdf:type rdf:Property)
- -> (rdf:first rdf:type rdf:Property)
- -> (rdf:rest rdf:type rdf:Property)
- -> (rdf:value rdf:type rdf:Property)
- -> (rdf:nil rdf:type rdf:List)
- # RDFS Axioms (RDF Semantics, section 4.1)
- -> (rdf:type rdfs:domain rdfs:Resource)
- -> (rdfs:domain rdfs:domain rdf:Property)
- -> (rdfs:range rdfs:domain rdf:Property)
- -> (rdfs:subPropertyOf rdfs:domain rdf:Property)
- -> (rdfs:subClassOf rdfs:domain rdfs:Class)
- -> (rdf:subject rdfs:domain rdf:Statement)
- -> (rdf:predicate rdfs:domain rdf:Statement)
- -> (rdf:object rdfs:domain rdf:Statement)
- -> (rdfs:member rdfs:domain rdfs:Resource)
- -> (rdf:first rdfs:domain rdf:List)
- -> (rdf:rest rdfs:domain rdf:List)
- -> (rdfs:seeAlso rdfs:domain rdfs:Resource)
- -> (rdfs:isDefinedBy rdfs:domain rdfs:Resource)
- -> (rdfs:comment rdfs:domain rdfs:Resource)

- -> (rdfs:label rdfs:domain rdfs:Resource)
- -> (rdf:value rdfs:domain rdfs:Resource)
- -> (rdf:type rdfs:range rdfs:Class)
- -> (rdfs:domain rdfs:range rdfs:Class)
- -> (rdfs:range rdfs:range rdfs:Class)
- -> (rdfs:subPropertyOf rdfs:range rdf:Property)
- -> (rdfs:subClassOf rdfs:range rdfs:Class)
- -> (rdf:subject rdfs:range rdfs:Resource)
- -> (rdf:predicate rdfs:range rdfs:Resource)
- -> (rdf:object rdfs:range rdfs:Resource)
- -> (rdfs:member rdfs:range rdfs:Resource)
- -> (rdf:first rdfs:range rdfs:Resource)
- -> (rdf:rest rdfs:range rdf:List)
- -> (rdfs:seeAlso rdfs:range rdfs:Resource)
- -> (rdfs:isDefinedBy rdfs:range rdfs:Resource)
- -> (rdfs:comment rdfs:range rdfs:Literal)
- -> (rdfs:label rdfs:range rdfs:Literal)
- -> (rdf:value rdfs:range rdfs:Resource)
- -> (rdf:Alt rdfs:subClassOf rdfs:Container)
- -> (rdf:Bag rdfs:subClassOf rdfs:Container)
- -> (rdf:Seq rdfs:subClassOf rdfs:Container)
- -> (rdfs:ContainerMembershipProperty rdfs:subClassOf rdf:Property)
- -> (rdfs:isDefinedBy rdfs:subPropertyOf rdfs:seeAlso)
- -> (rdf:XMLLiteral rdf:type rdfs:Datatype)
- -> (rdf:XMLLiteral rdfs:subClassOf rdfs:Literal)
- -> (rdfs:Datatype rdfs:subClassOf rdfs:Class)

XSD Axioms (XML Schema datatypes; excluding some less-useful)

- -> (xsd:float rdf:type rdfs:Datatype)
- -> (xsd:double rdf:type rdfs:Datatype)
- -> (xsd:int rdf:type rdfs:Datatype)
- -> (xsd:long rdf:type rdfs:Datatype)
- -> (xsd:short rdf:type rdfs:Datatype)
- -> (xsd:byte rdf:type rdfs:Datatype)
- -> (xsd:unsignedByte rdf:type rdfs:Datatype)
- -> (xsd:unsignedShort rdf:type rdfs:Datatype)
- -> (xsd:unsignedInt rdf:type rdfs:Datatype)
- -> (xsd:unsignedLong rdf:type rdfs:Datatype)
- -> (xsd:decimal rdf:type rdfs:Datatype)
- -> (xsd:integer rdf:type rdfs:Datatype)
- -> (xsd:nonPositiveInteger rdf:type rdfs:Datatype)
- -> (xsd:nonNegativeInteger rdf:type rdfs:Datatype)
- -> (xsd:positiveInteger rdf:type rdfs:Datatype)
- -> (xsd:negativeInteger rdf:type rdfs:Datatype)
- -> (xsd:boolean rdf:type rdfs:Datatype)
- -> (xsd:string rdf:type rdfs:Datatype)
- -> (xsd:anyURI rdf:type rdfs:Datatype)
- -> (xsd:hexBinary rdf:type rdfs:Datatype)
- -> (xsd:base64Binary rdf:type rdfs:Datatype)
- -> (xsd:date rdf:type rdfs:Datatype)
- -> (xsd:time rdf:type rdfs:Datatype)
- -> (xsd:dateTime rdf:type rdfs:Datatype)
- -> (xsd:duration rdf:type rdfs:Datatype)
- -> (xsd:gDay rdf:type rdfs:Datatype)
- -> (xsd:gMonth rdf:type rdfs:Datatype)
- -> (xsd:gYear rdf:type rdfs:Datatype)

- -> (xsd:gYearMonth rdf:type rdfs:Datatype)
- -> (xsd:gMonthDay rdf:type rdfs:Datatype)

```
# RDF/S entailment rules (RDF Semantics W3C Recommendation)
```

[rdf1: (?x ?a ?y) -> (?a rdf:type rdf:Property)]

[rdfs2: (?x ?a ?y) (?a rdfs:domain ?z) -> (?x rdf:type ?z)]

[rdfs3: (?x ?a ?y) (?a rdfs:range ?z) -> (?y rdf:type ?z)]

[rdfs4a: (?x ?a ?y) -> (?x rdf:type rdfs:Resource)]

[rdfs4b: (?x ?a ?y) -> (?y rdf:type rdfs:Resource)]

[rdfs5: (?a rdfs:subPropertyOf ?b) (?b rdfs:subPropertyOf ?c) -> (?a rdfs:subPropertyOf ?c)]

[rdfs6: (?a rdf:type rdf:Property) -> (?a rdfs:subPropertyOf ?a)]

[rdfs7: (?x ?a ?y) (?a rdfs:subPropertyOf ?b) -> (?x ?b ?y)]

[rdfs8: (?x rdf:type rdfs:Class) -> (?x rdfs:subClassOf rdfs:Resource)]

[rdfs9: (?x rdfs:subClassOf ?y) (?a rdf:type ?x) -> (?a rdf:type ?y)]

[rdfs10: (?x rdf:type rdfs:Class) -> (?x rdfs:subClassOf ?x)]

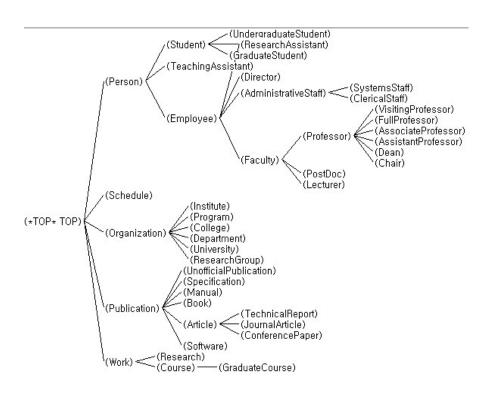
[rdfs11: (?x rdfs:subClassOf ?y) (?y rdfs:subClassOf ?z) -> (?x rdfs:subClassOf ?z)]

[rdfs12: (?x rdf:type rdfs:ContainerMembershipProperty) -> (?x rdfs:subPropertyOf rdfs:member)]

[rdfs13: (?x rdf:type rdfs:Datatype) -> (?x rdfs:subClassOf rdfs:Literal)]

[LUBM 벤치마크 테스트 셋]

1. LUBM 온톨로지 클래스 계층 구조



2. SPARQL로 기술한 LUBM 테스트 질의

Q1:

 $PREFIX\ rdf: < http://www.w3.org/1999/02/22-rdf-syntax-ns\#>$

PREFIX ub: ">http://www.lehigh.edu/~zhp2/2004/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/univ-bench.owl#>">http://www.lehigh.ed

SELECT ?x

WHERE {

?x rdf:type ub:GraduateStudent .

 $?x\ ub: takes Course < http://www.Department0.University0.edu/Graduate Course0>.$

```
}
 Q2:
PREFIX rdf: <a href="http://www.w3.org/1999/02/22-rdf-syntax-ns#">http://www.w3.org/1999/02/22-rdf-syntax-ns#</a>
 PREFIX ub: <a href="http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/univ-bench.owl#>">http://www.lehigh.ed
 SELECT ?x ?y ?z
 WHERE {
 ?x rdf:type ub:GraduateStudent .
  ?y rdf:type ub:University.
 ?z rdf:type ub:Department .
 ?x ub:memberOf?z.
 ?z ub:subOrganizationOf?y.
 ?x ub:undergraduateDegreeFrom ?y .
 }
 Q3:
 PREFIX rdf: <a href="http://www.w3.org/1999/02/22-rdf-syntax-ns#">http://www.w3.org/1999/02/22-rdf-syntax-ns#</a>
 PREFIX ub: <a href="http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/univ-bench.owl#>">http://www.lehigh.ed
 SELECT ?x
 WHERE {
 ?x rdf:type ub:Publication .
 ?x ub:publicationAuthor
  <a href="http://www.Department0.University0.edu/AssistantProfessor0">http://www.Department0.University0.edu/AssistantProfessor0</a>.
 }
 Q4:
 PREFIX rdf: <a href="http://www.w3.org/1999/02/22-rdf-syntax-ns#">http://www.w3.org/1999/02/22-rdf-syntax-ns#</a>
 PREFIX ub: <a href="http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>
 SELECT ?x ?y1 ?y2 ?y3
 WHERE {
```

```
?x rdf:type ub:Professor.
   ?x ub:worksFor <a href="http://www.Department0.University0.edu">http://www.Department0.University0.edu</a>.
 ?x ub:name ?y1.
   ?x ub:emailAddress ?y2.
 ?x ub:telephone ?y3.
 }
 Q5:
PREFIX rdf: <a href="http://www.w3.org/1999/02/22-rdf-syntax-ns#">http://www.w3.org/1999/02/22-rdf-syntax-ns#</a>
 PREFIX ub: <a href="http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/univ-bench.owl#>">http://www.lehigh.ed
 SELECT ?x
 WHERE {
 ?x rdf:type ub:Person.
 ?x\ ub:memberOf < http://www.Department0.University0.edu>.
 }
 Q6:
 PREFIX rdf: <a href="http://www.w3.org/1999/02/22-rdf-syntax-ns#">http://www.w3.org/1999/02/22-rdf-syntax-ns#</a>
 PREFIX ub: <a href="http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/univ-bench.owl#>">http://www.lehigh.ed
 SELECT ?x
 WHERE {
 ?x rdf:type ub:Student.
 }
 Q7:
 PREFIX rdf: <a href="http://www.w3.org/1999/02/22-rdf-syntax-ns#">http://www.w3.org/1999/02/22-rdf-syntax-ns#</a>
 PREFIX ub: <a href="http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/univ-bench.owl#>">http://www.lehigh.ed
 SELECT ?x ?y
 WHERE {
 ?x rdf:type ub:Student.
```

```
?y rdf:type ub:Course.
  ?x ub:takesCourse ?y .
 <a href="http://www.Department0.University0.edu/AssociateProfessor0">http://www.Department0.University0.edu/AssociateProfessor0</a>
 ub:teacherOf?y.
 }
 Q8:
 PREFIX rdf: <a href="http://www.w3.org/1999/02/22-rdf-syntax-ns#">http://www.w3.org/1999/02/22-rdf-syntax-ns#</a>
 PREFIX ub: <a href="http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/univ-bench.owl#>">http://www.lehigh.ed
SELECT ?x ?y ?z
 WHERE {
 ?x rdf:type ub:Student.
 ?y rdf:type ub:Department .
 ?x ub:memberOf ?y .
 ?y ub:subOrganizationOf <a href="http://www.University0.edu">http://www.University0.edu</a> .
 ?x ub:emailAddress ?z.
 }
 Q9:
 PREFIX rdf: <a href="http://www.w3.org/1999/02/22-rdf-syntax-ns#">http://www.w3.org/1999/02/22-rdf-syntax-ns#</a>
 PREFIX ub: <a href="http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/univ-bench.owl#>">http://www.lehigh.ed
 SELECT DISTINCT ?x ?y ?z
 WHERE {
 ?x rdf:type ub:Student.
 ?y rdf:type ub:Faculty .
 ?z rdf:type ub:Course .
 ?x ub:advisor ?y.
 ?y ub:teacherOf?z.
 ?x ub:takesCourse ?z .
 }
```

```
Q10:
PREFIX rdf: <a href="http://www.w3.org/1999/02/22-rdf-syntax-ns#">http://www.w3.org/1999/02/22-rdf-syntax-ns#</a>
PREFIX ub: <a href="http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www
SELECT ?x
WHERE {
?x rdf:type ub:Student.
?x\ ub: takes Course < http://www.Department 0. University 0. edu/Graduate Course 0>.
}
Q11:
PREFIX rdf: <a href="http://www.w3.org/1999/02/22-rdf-syntax-ns#">http://www.w3.org/1999/02/22-rdf-syntax-ns#</a>
PREFIX ub: <a href="http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/univ-bench.owl#>">http://www.lehigh.ed
SELECT ?x
WHERE {
?x rdf:type ub:ResearchGroup.
?x\ ub: subOrganizationOf < http://www.University0.edu>.
}
Q12:
PREFIX rdf: <a href="http://www.w3.org/1999/02/22-rdf-syntax-ns#">http://www.w3.org/1999/02/22-rdf-syntax-ns#</a>
PREFIX ub: <a href="http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/univ-bench.owl#>">http://www.lehigh.edu/~zhp2/2004/univ-bench.owl#>">http://www.lehigh.ed
SELECT ?x ?y
WHERE {
?x rdf:type ub:Chair .
?y rdf:type ub:Department .
 ?x ub:worksFor ?y.
 ?y ub:subOrganizationOf <a href="http://www.University0.edu">http://www.University0.edu</a> .
}
```

```
Q13:
PREFIX rdf: <a href="http://www.w3.org/1999/02/22-rdf-syntax-ns#">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#>
SELECT ?x
WHERE {
?x rdf:type ub:Person .
<a href="http://www.University0.edu">http://www.University0.edu</a> ub:hasAlumnus ?x .
}

Q14:
PREFIX rdf: <a href="http://www.w3.org/1999/02/22-rdf-syntax-ns#">http://www.w3.org/1999/02/22-rdf-syntax-ns#</a>
PREFIX ub: <a href="http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#">http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#</a>
SELECT ?x
WHERE {
?x rdf:type ub:UndergraduateStudent .
}
```