

## 경로 정보를 이용한 RDF와 RDF 스키마의 저장 구조 설계

김연희\*, 김병곤\*\*, 이재호\*\*\*, 임해철\*

\*홍익대학교 컴퓨터공학과

\*\*부천대학 e-비즈니스과

\*\*\*경인교육대학교 컴퓨터교육과

{kyh, lim}@cs.hongik.ac.kr, bgkim@bc.ac.kr, jhlee@gin.ac.kr

### The Design of Storage Structure for Path Expressions in RDF and RDF Schema

Youn Hee Kim\*, Byung Gon Kim\*\*, Jaeho Lee\*\*\*, Hae Chull Lim\*

\*Dept. of Computer Engineering, Hong Ik University

\*\*Dept. of e-Business, Bucheon College

\*\*\*Dept. of Computer Education, Gyeongin National University of Education

#### 요 약

정보의 단순한 연결을 표현하는 현재 웹 환경의 한계를 보완하기 위해 차세대 웹으로서 평가되고 있는 시맨틱 웹에서는 정보 리소스의 의미와 개념적 관계를 정의하는 메타데이터와 온톨로지의 역할이 무엇보다 중요시되고 있다. 따라서 RDF와 RDF 스키마와 같은 표준 언어로써 표현되는 메타데이터와 온톨로지의 효과적인 저장과 검색의 방법이 필요하다. RDF와 RDF 스키마는 그래프 모델로서 표현이 가능하고 다양한 질의 형태가 그래프 형태에서 추출 가능한 경로 형식으로 표현될 수 있기 때문에 보다 효율적인 질의 처리를 위해서는 경로 정보의 저장에 대한 연구가 필요하다. 따라서 본 논문에서는 관계형 데이터베이스를 기반으로 RDF와 RDF 스키마의 기본적인 정보와 특정 클래스, 프로퍼티, 리소스로 시작하는 경로 정보를 함께 저장할 수 있는 저장 구조를 제안한다.

#### 1. 서 론

단순한 연결을 이용해 사용자의 검색 요구에 대해 관련된 정보를 쉽게 제공하는 현재 웹 환경은 정확하고 풍부한 정보의 검색과 정보의 의미 있는 해석에 한계를 가지고 있다. 정보의 의미 있는 관계를 개념적으로 정의하고 추론 등의 기능을 통해 새로운 지식의 생성이 가능한 차세대 웹 환경이 요구된다. 이러한 요구에 따라 W3C에서는 차세대 웹의 대안으로 시맨틱 웹을 제안하였다. 시맨틱 웹에서는 메타데이터와 온톨로지를 이용하여 정보 리소스의 개념을 정의하고 의미적 연관성을 표현함으로써 보다 지능적인 정보 검색은 물론 자동화된 다양한 웹 서비스를 제공할 수 있다. 온톨로지는 특정 도메인에서 사용되는 용어와 용어들 간의 관계를 정의한다. 메타데이터는 온톨로지서 정의된 용어를 이용하여 정보 리소스의 의미와 정보 리소스 간의 의미적 연관성을 표현한다. RDF(Resource Description Framework)와 RDF 스키마는 각각 메타데이터와 온톨로지를 기술하기 위한 표준 언어로서 W3C에서 제안하였다[1]. RDF는 웹 상의 정보 리소스에 대한 표현 및 교환의 수단으로 트리플(Subject/Predicate/Object)의 기본 구조로 구성되어 있으며 XML의 문법 형태를 그대로 이용한다. RDF 스키마는 RDF에 적용되는 클래스와 프로퍼티의 개념을 정의하고 각 개념들의 관계를 정의하는 온톨로지를 기술한다. 시맨틱 웹에서의 진보된 정보 검색 서비스를

제공하기 위해서는 RDF와 RDF 스키마로 기술된 메타데이터와 온톨로지를 함께 이용해야 한다. 그러므로 RDF와 RDF 스키마를 이용해 작성된 메타데이터와 온톨로지 정보를 효율적으로 저장하고 검색하기 위한 연구가 필요하다[2,3,4].

따라서 본 논문에서는 트리플의 기본 구조에 기반한 질의뿐만 아니라 복잡한 경로로 표현되는 질의 처리를 위한 RDF와 RDF 스키마의 저장 구조를 제안한다.

#### 2. 관련 연구

RDF와 RDF 스키마의 정보를 트리플 구조에 따라 기존 데이터베이스를 이용하여 저장하고 검색하는 시스템에 대한 몇몇 대표적인 연구 결과물들이 이미 제안되었다.

RDFSuite 시스템은 객체-관계 데이터베이스에 기반한 저장 구조와 질의 처리 모듈을 개발하였다[5]. RDFSuite 시스템은 RDF 문서의 파서인 VRP, RDF의 저장 구조인 RSSDB, RDF의 질의 처리기로 구성되어 있다. 그러나 기존 데이터베이스에서 제공되는 기능을 최대한 이용하면서도 이미 개발된 많은 애플리케이션과의 호환성을 고려할 때, 관계형 데이터베이스에 기반한 저장 구조의 개발이 필요하다. Sesame는 RDF와 RDF 스키마 정보를 모두 고려한 저장 시스템으로서, 어떠한 하부 저장 구조와도 호환이 가능하도록 RAL(Repository Abstraction Layer)와 RDF 처리를 위한 3가지 기능성 모듈을 개발하였다[6]. Sesame 시스템은 대부분의 저장 구조와의 호

본 연구는 한국과학기술원 기초과학연구소(과제번호 : R01-2004-000-10586-0(2004))의 지원을 받아서 작성되었음

환이 가능하지만 하부의 저장 구조에 맞게 변환하는 과정이 반드시 요구되므로 그에 따른 오버헤드가 크다.

RDF와 RDF 스키마에 대한 질의는 기본적인 트리플 구조에 기반을 둔 간단한 형태의 질의가 일반적이거나, RDF와 RDF 스키마 내에 존재하는 다양한 의미와 관계를 검색하기 위해서는 복잡한 경로로 표현되는 질의 형태에 대한 처리가 필요하다. 그러나 RDF의 저장과 검색을 위한 기존의 연구 내용은 주로 RDF의 트리플 구조에만 초점이 맞추어진 경우가 대부분이며 RDF 스키마를 고려하지 않는 경우도 있다. 또한 복잡한 경로에 대한 질의 처리를 위해서는 조인의 횟수가 급격히 증가하여 비효율적으로 처리되는 한계를 가지고 있다.

따라서 RDF와 RDF 스키마에 대한 다양한 질의 형태를 처리하기 위해서 경로 정보를 위한 저장 구조가 요구된다.

### 3. RDF와 RDF 스키마의 데이터 모델

RDF와 RDF 스키마는 노드와 간선에 모두 레이블이 표현된 그래프 형태로 표현될 수 있다. RDF를 표현하기 위한 그래프 구조에서 노드는 리소스나 리터럴 값을 의미하고 간선은 프로퍼티를 의미한다.

RDF 스키마는 RDF 내 리소스의 개념을 나타내는 클래스와 특정 클래스를 설명하는 프로퍼티, 프로퍼티 값의 데이터 타입, 클래스의 계층 구조, 프로퍼티의 계층 구조 등을 정의한다. RDF 스키마를 표현하기 위한 그래프 모델에서 노드는 클래스를 의미하고 간선은 특정 클래스에 포함된 프로퍼티를 의미한다. 따라서 간선의 레이블은 프로퍼티의 이름을 나타낸다. 그리고 클래스의 계층 구조 또는 프로퍼티의 계층 구조를 나타내는 부가적인 간선이 존재한다.

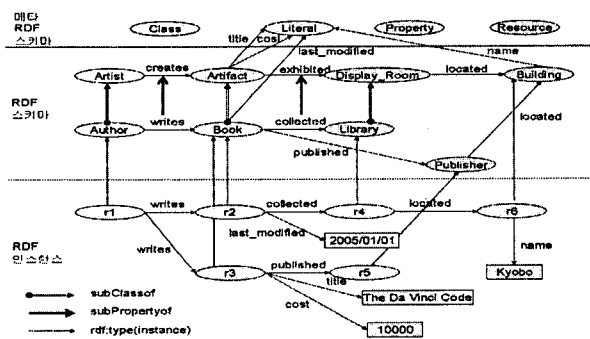


그림 1. RDF와 RDF 스키마의 그래프 표현 예

<그림 1>은 도서관에 소장되어 있는 책과 그 책의 저자에 대한 정보를 기술한 RDF 문서와 RDF 스키마 문서를 그래프 형태로 표현한 예를 보여준다. 사각형으로 표현된 노드는 리터럴 값을 나타낸다.

### 4. RDF와 RDF 스키마에 대한 질의 패턴 분석

RDF와 RDF 스키마에 대한 질의는 다음과 같이 크게 세 가지의 패턴으로 분류된다. RDF와 RDF 스키마에 대한 다양한 질의 형태는 트리플의 단순 경로 또는 여러

트리플 구조가 반복하는 복잡한 경로로서 표현이 가능하다. 따라서 RDF와 RDF 스키마로부터 추출된 경로를 위한 저장 구조가 요구된다.

(1) RDF 스키마에 관한 질의

① 계층 구조에 관한 질의

ㄱ. 클래스 계층 구조에 관한 질의

<subject, subClassof, object>

ㄴ. 프로퍼티 계층 구조에 관한 질의

<subject, subPropertyof, object>

② 프로퍼티의 도메인과 레인지 클래스에 관한 질의

<subject, domain, object> or <subject, range, object>

(2) RDF 인스턴스에 관한 질의

① 트리플 구조에 기반한 단순 경로를 이용한 질의

② 여러 단순 경로가 연결된 복잡 경로를 이용한 질의

(3) RDF 스키마와 인스턴스를 혼합한 질의

① rdf:type에 의해 표현된 클래스와 클래스의 인스턴스인 리소스의 관계를 이용한 질의 형태

### 5. RDF와 RDF 스키마를 위한 저장 구조

본 논문에서는 관계형 데이터베이스의 테이블을 이용한 RDF와 RDF 스키마 정보의 저장 구조를 제안한다. 특히 제안한 저장 구조는 RDF와 RDF 스키마가 정의하는 클래스, 프로퍼티, 리소스에 대한 정보는 물론 RDF와 RDF 스키마에서 표현될 수 있는 모든 경로 정보를 저장하는데 목적을 두고 있다. 제안한 저장 구조를 구성하기 위해서는 먼저 RDF 스키마의 그래프 표현에서 루트 노드에서부터 각 노드까지의 가장 긴 경로 표현을 추출한 후 PList(Path List)라는 구조에 저장한 후 리소스, 클래스, 프로퍼티, 인스턴스, 클래스 경로, 프로퍼티 경로, 리터럴의 총 7개 테이블 구조를 이용해 RDF와 RDF 스키마 정보를 저장한다.

PList는 각 경로 표현을 구별하는 PathID와 각 경로를 표현하는 리스트로 구성되어 있다. 각 리스트는 해당 경로를 구성하는 클래스나 프로퍼티의 노드들을 순서대로 표현한다.

<그림 2>는 <그림 1>의 RDF 스키마 부분에서 추출한 가장 긴 경로 정보들을 표현한 PList의 예를 보여준다.

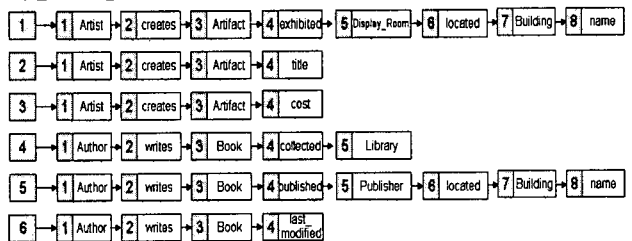


그림 2. PList에 저장된 경로 정보

리소스 테이블은 RDF 문서 내에 리소스에 대한 네임스페이스 정보를 저장한다. 리소스 테이블은 각 리소스에 대해 고유하게 정의된 RID(Resource ID)의 값과 각 리소스를 표현하는 네임스페이스 정보를 저장한다. 본

논문에서 RID는 r1, r2, r3와 같은 순서로 모든 리소스에 순서대로 지정된다.

클래스 테이블은 RDF 스키마에 정의된 클래스에 대한 정보로서 클래스의 이름과 클래스의 계층 정보를 저장한다. 특히 클래스간의 계층 정보를 표현하기 위하여 3-튜플 구조의 위치 정보를 이용한다. 3-튜플 구조의 위치 정보는 먼저 RDF 스키마에 정의된 클래스들을 이용해 트리 형태를 구성한 후 트리 구조에서의 각 클래스 노드의 위치를 (시작 위치, 끝 위치, 레벨) 정보를 이용하여 표현한 것이다[7]. 3-튜플 위치 정보는 트리를 깊이 우선 순회 방법을 이용하여 순회한 후 결정된다.

<그림 3>는 <그림 1>의 RDF 스키마 부분에서 정의한 클래스들을 이용해 트리를 구성한 예이다. 트리의 루트 노드가 존재하지 않을 경우는 임의의 루트 노드를 가정하고 특정 클래스의 서브 클래스로 정의된 클래스는 자식 노드로서 표현함으로써 클래스의 계층 구조를 표현한다. 예를 들어, 클래스 노드 C1의 위치 정보가 (S1, E1, L1)이고 또 다른 클래스 노드 C2의 위치 정보가 (S2, E2, L2)일때, S1<S2 이고 E2<E1, 그리고 L2=L1+1 이 만족한다면 C2는 C1의 서브 클래스라 할 수 있다.

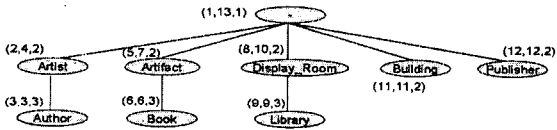


그림 3. 위치 정보를 표현한 클래스의 트리 구조 예

프로퍼티 테이블은 RDF 스키마에 정의된 프로퍼티에 대한 정보로서 이름과 계층 정보를 저장한다. 클래스와 마찬가지로 방법으로 프로퍼티간의 계층 구조를 표현한다.

인스턴스 테이블은 rdf:type 속성을 이용하여 특정 클래스의 인스턴스로 정의된 리소스에 대한 정보를 저장한다.

클래스 경로 테이블은 특정 클래스로부터 시작하는 경로에 대한 정보를 저장한다. 클래스의 CID와 PList에서의 특정 클래스가 나타나는 경로 리스트 번호와 리스트 내의 위치를 저장한다. 예를 들어, CID 2의 Artist 클래스로부터 시작되는 경로는 PList에서 PathID 1,2,3의 리스트이다. PathID 1,2,3의 경로 리스트에서 Artist 클래스는 처음에 나타나므로 클래스 경로 테이블에 <2, 1, 1>, <2, 2, 1>, <2, 3, 1>의 정보가 저장된다. 따라서 Artist 클래스로부터 시작되는 경로 표현의 질의를 처리할 수 있다.

프로퍼티 경로 테이블은 클래스 경로 테이블과 마찬가지로 특정 프로퍼티로부터 시작하는 경로에 대한 정보를 저장한다.

리터럴 테이블은 리터럴 정보를 포함하는 프로퍼티에 대한 정보를 저장한다.

RDF와 RDF 스키마에 대한 모든 질의 형태는 RDF와 RDF 스키마의 그래프 표현에서 추출한 경로 표현을 이용해 표현될 수 있다. 따라서 본 논문에서 제안한 PList 구조와 7개의 테이블 구조를 이용하여 테이블간의 조인 횟수를 최소화하면서 특정 클래스, 프로퍼티, 리소스에서 시작하는 질의를 처리할 수 있는 장점이 있다.

<그림 4>는 <그림 1>의 RDF와 RDF 스키마 정보를 7개의 테이블 구조로 저장한 예를 보여준다.

RID	URI	CID	name	start	end	level
r1	www.libraryinfo.com/author#A1	1	-	1	13	1
r2	www.libraryinfo.com/book#B1	2	Artist	2	4	2
r3	www.libraryinfo.com/book#B2	3	Author	3	3	3
r4	www.libraryinfo.com/location#L1	4	Artfact	5	7	2
r5	www.libraryinfo.com/location#L2	6	Book	6	6	3
r6	www.libraryinfo.com/building#B1	6	Display_Room	8	10	2
		7	Library	9	9	3
		8	Building	11	11	2
		9	Publisher	12	12	2

<리소스 테이블>

PID	name	start	end	level
1	-	1	14	1
2	creatas	2	4	2
3	writes	3	3	3
4	exhibited	5	7	2
5	collected	6	6	3
6	title	8	8	2
7	cost	8	9	2
8	located	10	10	2
9	name	11	11	2
10	last_modified	12	12	2
11	published	13	13	2

<프러퍼티 테이블>

Literal	PID
The Da Vinci Code	6
10000	7
Kyobo	9
2005/01/01	10

<리터럴 테이블>

CID	list_num	position	PID	list_num	position
2	1	1	2	1	2
2	2	1	2	2	2
2	3	1	2	3	2
3	4	1	3	4	2
3	5	1	3	5	2
3	6	1	3	6	2
4	1	3	4	1	4
4	2	3	5	4	4
4	3	3	6	2	4
...	...	...	...	...	...

<클래스 경로 테이블> <프로퍼티 경로 테이블>

CID	RID
3	r1
5	r2
5	r3
7	r4
8	r5
9	r5

<인스턴스 테이블>

그림 4. 경로 정보에 기반한 저장 구조 예

6. 결론

본 논문에서는 RDF와 RDF 스키마에 대한 질의 형태를 분석하여 모든 질의 형태가 RDF와 RDF 스키마의 그래프 표현에서 추출한 경로 정보로서 처리가 가능하다는 데 초점을 맞추고 RDF와 RDF 스키마를 위한 저장 구조를 제안한다. 제안한 저장 구조는 RDF와 RDF 스키마에서 추출 가능한 모든 경로 정보는 물론 클래스, 프로퍼티, 리소스에 대한 정의, 클래스의 계층 구조, 프로퍼티의 계층 구조에 대한 정보도 저장할 수 있어 다양한 형태의 질의 처리가 가능하다. 제안한 저장 구조는 관계형 데이터베이스의 테이블 구조를 이용하여 리소스, 클래스, 프로퍼티, 인스턴스, 클래스 경로, 프로퍼티 경로, 리터럴의 총 7개 테이블로 구성된다. 이러한 저장 구조를 이용하면 단순한 트리 구조의 질의뿐만 아니라 여러 트리가 연결된 복잡한 질의에 대해서도 테이블의 조인 횟수를 최소화하면서도 특정한 클래스나 프로퍼티, 리소스에서 시작되는 모든 경로에 대한 질의 처리가 가능하다.

<참고 문헌>

[1] W3C, "RDF Primer", 2004. 2.  
 [2] A. Matono, T. Amagasa, M. Yoshikawa, S. Uemura, "An Indexing Scheme for RDF and RDF Schema based on Suffix Arrays", In Proc. of the first International Workshop on SWDB, pages 151-168, 2003.  
 [3] A. Reggiori, D. van Gulik, Z. Bjeloglic, "Indexing and retrieving Semantic Web resources: the RDFStore model", In Proc. of SWAD-Europe Workshop on Semantic Web Storage and Retrieval, 2003.  
 [4] S. Barton, "Designing Indexing Structure for Discovering Relationships in RDF Graphs", In Proc. of the Dataso 2004, pages 7-17, 2004.  
 [5] S. Alexaki et al., "The ICS-FORTH RDFSuite: Managing Voluminous RDF Description Bases", SemWeb 2001.  
 [6] J. Broekstra et al., "Sesame: A Generic Architecture for Storing and Querying RDF and RDF Schema", International Semantic Web Conference 2002.  
 [7] V. Christophides, D. Plexousakis, M. Scholl, S. Tourtouris, "On labeling schemes for the semantic web", In Proc. of the twelfth international conference on World Wide Web, pages 544-555, 2003.