**경로 정보를 이용한 RDF와 RDF 스키마의 저장 구조 설계**

**1. 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 로퍼티를 의미한다. RDF 스키마는 RDF 내 리소스의 개념을 나타내는 클래스와 특정 클래스를 설명하는 프로퍼티, 프로퍼티 값의 데이터 타입, 클래스의 계층 구조, 프로퍼티의 계층구조 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]

**2. 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

**3. RDF와 RDF 스키마를 위한 저장 구조**

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. 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. 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+1이 만족한다면 C2 는 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 장한다. 클래스와 마찬가지 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 리스트내의 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 Artist클래스로부터 시작되는 경로 표현의 질의를 처리할 수 있다.

프로퍼티 경로 테이블은 클래스 경로 테이블과 마찬가지로 특정 프로퍼티로부터 시작하는 경로에 대한 정보를저장한다.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 조인횟수를 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.



[그림 4. 경로 정보에 기반한 저장 구조 예]