

시맨틱 웹에서의 메타데이터 검색을 위한 RDF 저장 구조 및 시스템의 개발에 관한 연구

A study on Development of RDF Triple Storage System for Retrieval of Metadata in the Semantic Web

박지형(Park Ji Hyoung)*, 이명진(Lee Myung Jin)**, 홍준석(Hong June Seok)***

초 록

RDF는 다양한 응용 분야에서 정보의 표현 및 저장 수단으로 활용되고 있으며, 차세대 웹으로 촉망받고 있는 시맨틱 웹에서는 중요한 기반구조의 역할을 갖고 있다. RDF 메타데이터의 저장과 검색 기능을 제공하기 위하여 개발된 기존 시스템들의 단점을 극복하기 위하여 본 연구에서는 새로운 RDF 메타데이터 저장 구조를 설계하고 이를 구현한 시스템 SMARTTriple을 개발하였다. 이 시스템은 기존 시스템보다 우수한 성능을 갖는 RDF 메타데이터에 대한 효율적인 저장 및 검색 기능을 제공함으로써 향후 RDF 표준을 이용하는 웹 기반 응용 시스템의 연구와 개발에 좋은 도구를 제공하게 될 것이다.

ABSTRACT

The Resource Description Framework(RDF) utilized for the representation and storage of information in various application, plays an important role in the Semantic Web, the next generation web. A number of RDF systems have been implemented for retrieval of RDF metadata. However, they do not provide the sufficient performance for the querying function of practical application system. In this paper we describe the design and implementation of RDF system, SMARTTriple, which has a RDF storage structures using index map and provides the better performance than the previous systems. We present a performance comparison results with the well-known RDF systems. SMARTTriple will be a great contribution to the research and development of Semantic Web application.

키워드 : RDF, 메타데이터, 시맨틱 웹
RDF, metadata, Semantic Web

* 연세대학교 정보산업공학과 석사과정
** 연세대학교 정보산업공학과 박사과정
*** 경기대학교 경상대학 경영정보학전공 부교수, 교신저자

1. 서 론

정보의 표현 및 교환을 위한 표준으로 W3C에서 만들어진 RDF(Resource Description Framework)는 뉴스, 도서, 음악 등의 다양한 콘텐츠의 수집 및 통합에서부터 기업과 정부, 생물정보학, 통신 등에 이르기까지 다양한 분야의 응용 시스템 개발에 사용되어져 왔다. 특히 웹 상에서의 지식의 교환을 목적으로 하는 시맨틱 웹이 확산됨에 따라 시맨틱 웹의 기반 기술이 되고 있는 RDF 표준은 그 활용도가 더욱 넓어지고 있다. 시맨틱 웹 시스템을 포함하여 많은 응용 시스템들이 RDF 메타데이터 저장 및 검색 기능을 기반 시스템으로 사용하므로 RDF 메타데이터 검색 기능의 성능은 시맨틱 웹을 포함한 관련 분야의 응용 시스템의 성능에 매우 중요한 영향을 미친다.

이러한 RDF 메타데이터의 저장과 검색 기능을 제공하는 시스템 도구로 Jena[8], KAON2[11], Sesame[2] 등이 개발되었으나, 실용적인 시스템으로 적용하기에 충분한 성능을 보이고 있지 못하다. 효율적인 메타데이터 검색을 위해서는 RDF의 기본 표현 구조인 트리플 <subject predicate object>의 기본 데이터를 저장하는 데이터 구조와 더불어 빠른 검색 성능을 보장하기 위해 트리플의 인덱스에 대한 데이터 구조가 필요하다. 본 연구에서는 기존의 RDF 메타데이터 검색 시스템에 비해 우수한 성능을 제공하기 위하여 인덱스 맵 구조를 이용한 새로운 RDF 메타데이터 저장 구조를 설계하고 이러한 저장 구조를 이용한 RDF 저장 및 검색 시스템인 SMARTTriple(Semantic-web Reasoning Tools for RDF Triple)을 개발하였다. 대규모 시맨틱 웹 환경하에서 본 연구의 우수성을 입증하기 위하여 SMARTTriple과 현재 널리 사용되고 있는 RDF 저장 및 검색 시스템들과의 성능 비교를 수행하였다.

2. 관련 문헌 연구

시맨틱 웹을 포함하여 RDF 표준을 이용하는 응용 시스템의 개발 도구의 하나로 사용될 RDF 메타데이터 검색 시스템은 개발 목적에 따라 크게 2가지 유형으로 구분된다[13]. 첫 번째는 시맨틱 웹 응용 시스템에 사용할 목적으로 RDF 메타데이터 저장 및 검색 기능과 더불어 시맨틱 웹 개발 API, 시맨틱 웹 추론 기능 등을 동시에 제공하는 시스템들이다. 가장 널리 알려져 현재 가장 많이 활용되고 있는 HP의 Jena[8]는 소스가 공개된(open-source) 시스템으로 RDF 메타데이터의 생성, 추출, 변경, 삭제 기능을 갖는 API를 제공한다. 뿐만 아니라 시맨틱 웹 데이터에 대한 온톨로지 추론과 규칙 추론 기능을 제공하여 시맨틱 웹 응용 시스템을 개발하는데 필요한 거의 모든 기능을 제공하고 있다. 유럽을 중심으로 개발된 KAON2[11]도 시맨틱 웹 응용 시스템 개발을 위해 필요한 RDF 메타데이터 사용 기능과 Jena 수준의 추론 기능을 함께 제공하는 시스템이다. Jena와 KAON2는 대표적인 시맨틱 웹 구축용 도구 시스템으로 본 연구에서 대상으로 하고 있는 RDF 메타데이터의 저장과 검색 기능 외에도 시맨틱 웹 시스템 구축을 위한 다양한 기능을 제공하고 있다. 그러나 RDF 메타데이터의 검색 성능을 향상시키기 위하여 기본 저장 구조 이외에 추가적인 인덱스 구조는 사용하지 않고 있다. 그 밖에도 시맨틱 웹 데이터를 대상으로 연관성 검색 기능을 주 목적으로 하는 BRAHMS[4], Java 뿐만 아니라 C#, Python, Perl, PHP 등 다중 언어 API를 제공하여 다양한 개발환경을 지원하는 Redland[1] 등과 같은 시스템이 있다.

두 번째 유형은 대용량의 RDF 메타데이터를 저장하기 위하여 기존의 데이터베이스 관리시스템을 기반으로 저장 및 검색 기능을 제공하는 시스

탐들이다. 소스 개방형으로 제공되는 Sesame[2]는 인덱스 구조를 이용하여 저장된 RDF 데이터베이스를 대상으로 메타데이터 검색 기능에 추가적으로 RDF 스키마 추론 기능까지 제공하고 있다. 그 밖에도 별도의 쿼리 언어를 이용한 Java API를 제공하는 YARS[3], SPARQL 인터페이스와 RDF 스키마 추론 기능을 포함한 AllegroGraph[7] 등이 있다.

3. RDF의 데이터 구조

3.1 RDF의 개요

현재의 월드 와이드 웹(World Wide Web, 이하 웹)은 HTML(HyperText Markup Language)을 사용해서 정보를 표현하는데 초점이 맞춰져 있다. 이는 단순히 사람에게 보여주기 위한 용도로써, 기계가 읽을 수는 있지만 이해할 수는 없다는 것을 의미한다. 시맨틱 웹(Semantic Web)은 이러한 문제를 해결하기 위한 노력으로써, 웹상에 존재하는 데이터를 기계가 이해하고 자동으로 처리하도록 하는데 그 목적이 있다.

RDF(Resource Description Framework)는 시맨틱 웹을 이루는 기반 기술로써 기계가 이해할 수 있는 형태의 메타데이터(Metadata)를 기술하기 위한 방법이다. 도서관에서 직접 책을 찾아 본 사람이라면 도서관의 색인, 즉 메타 데이터가 얼마나 중요한 것인지를 알고 있을 것이다. 마찬가지로 웹에 존재하는 자원을 표준화된 메타 데이터 기술 방법인 RDF를 이용해서 기술함으로써 상이한 애플리케이션 간에 정보를 교환하고 이용할 수 있게 한다. 예를 들어 웹 페이지의 표제나 저자, 혹은 갱신 일자나 저작권 정보 공유 자원과 같은 웹 자원을 RDF를 이용해서 기술할 수 있다. RDF를 이용함으로써 웹 자원에 대한 메타 데이터를 기술할

수 있을 뿐만 아니라 자원 간의 관계를 기술할 수 있다. 또한 이를 통해 지식의 공유와 교환을 촉진할 수 있고 더 나아가 지능화된 에이전트를 이용해서 새로운 지식을 창출할 수 있다.

RDF는 웹에 존재하는 모든 자원(Resource)을 URI(Uniform Resource Identifiers)를 이용해서 명시하며, 그 자원이 가진 속성과 값으로 자원을 기술한다. 즉 자원과 자원 사이의 관계, 혹은 자원과 자원이 가진 특정 값을 이용해서 자원을 표현한다. 이러한 관계는 ER-다이어그램과 유사한 모양을 가진 RDF 그래프(RDF Graph)로 표현할 수 있다. 예를 들어, 다음의 선언문이 있다.

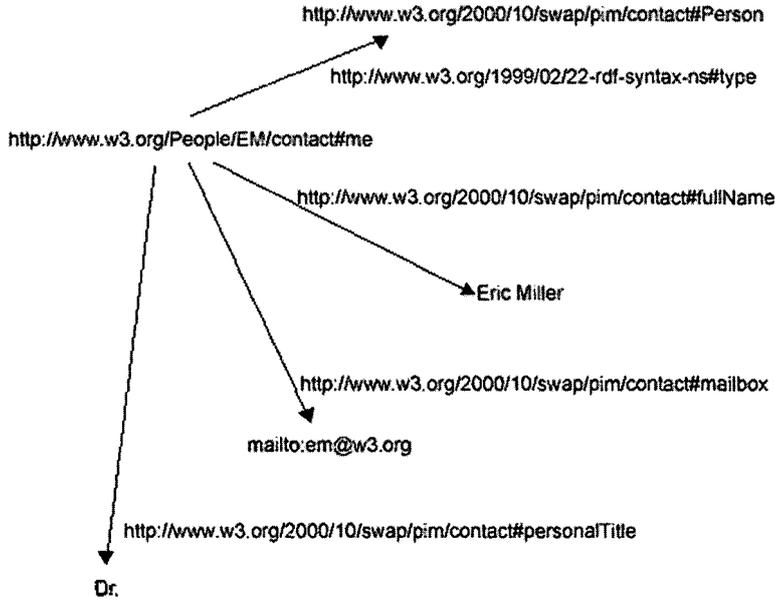
`http://www.w3.org/People/EM/contact#me`는 직함이 Dr.이고 이름이 Eric Miller인 사람으로써, 메일 주소는 `em@w3.org`이다.

이 선언문을 RDF 그래프로 표현하면 <그림 1>과 같다.

위의 RDF 그래프에서 자원을 식별하기 위해 다음과 같은 URI가 사용되었다.

- Eric Miller라는 이름을 가진 개체는 `http://www.w3.org/People/EM/contact#me` URI로 식별된다.
- 해당 개체의 종류는 `http://www.w3.org/2000/10/swap/pim/contact#Person`로 식별되는 사람이다.
- 메일 주소 값과의 관계를 나타내는 속성은 `http://www.w3.org/2000/10/swap/pim/contact#mailbox` URI로 식별된다.
- 속성의 값은 `mailto:em@w3.org`이다.

RDF는 메타 데이터를 표현하기 위한 모델이며, 모델의 표현 방법으로써 애플리케이션 사이의 데이터 교환 시 상호 운용성을 지원하기 위해 XML(eXtensible Markup Language) 문법을 따르는 RDF/XML 구문을 사용한다. RDF의 주요 목표



〈그림 1〉 예제 선언문의 RDF 그래프

중 하나는 바로 애플리케이션 사이에 표준화된 정보의 교환이며, 이는 XML과 그 제반 기술을 사용함으로써 충족될 수 있다. 위에 기술된 RDF 그래프를 RDF/XML 구문으로 표현하면 다음과 같다.

```

<?xml version="1.0"?>
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:contact="http://www.w3.org/2000/10/swap/pim/contact#">
  <contact:Person rdf:about="http://www.w3.org/People/EM/contact#me">
    <contact:fullName>Eric Miller</contact:fullName>
    <contact:mailbox rdf:resource="mailto:em@w3.org"/>
    <contact:personalTitle>Dr.</contact:personalTitle>
  </contact:Person>
</rdf:RDF>
    
```

RDF/XML 구문은 기계에 의해서 처리될 수 있으며, 또한 URI를 사용함으로써 표준화된 방법으로 자원을 유일하게 식별할 수 있다. 이와 같이 RDF는 웹상에 존재하는 자원뿐만 아니라 사람이나 자동차 같은 사물을 기술할 수도 있으며, URI를 사용함으로써 자원과 자원간의 관계를 정확하

게 표현하고 식별할 수 있다.

3.2 RDF 메타데이터의 Triple 구조

RDF는 웹 자원에 대한 Triple 구조의 선언문을 작성하는 방법을 제공한다. 가령 예를 들어, "http://www.example.org/index.html 웹 페이지를 John Smith가 작성했다"라는 구문은 영어를 이용하여 다음과 같은 문장으로 표현할 수 있다.

```

http://www.example.org/index.html
has a creator whose value is John
Smith
    
```

위의 구문을 살펴보면 기술하려는 대상이 존재하며, 대상에 대한 속성, 그리고 속성 값으로 이루어져 있다. 이와 같은 선언문에서 기술하려는 대상, 즉 http://www.example.org/index.html을 주어(subject)라고 하며, 대상이 가지고 있는 속성 creator

를 술어(predicate), 속성의 값 John Smith를 목적어(object)라고 한다. 이를 정리하면 다음과 같다.

- 주어(subject)는 URL `http://www.example.org/index.html`이고,
- 술어(predicate)는 “creator”라는 단어이며
- 목적어(object)는 “John Smith”라는 구이다.

위의 예제에서는 술어와 목적어로서 creator와 John Smith라는 단어를 사용했지만 RDF에서는 각각을 웹 자원으로 구분하기 위해 URL로 대체해서 사용한다. 이는 RDF 그래프를 이용해서 다음과 같이 노드(node)와 아크(arc)로 표현할 수 있다.

`http://www.example.org/index.html` has a creator whose value is John Smith
`http://www.example.org/index.html` has a creation-date whose value is August 16, 1999
`http://www.example.org/index.html` has a language whose value is English

RDF 그래프에서 URI를 나타내는 노드는 타원으로 표현되며, 리터럴(literal) 값을 갖는 노드는 4각형으로 표시된다. 또한 아크는 술어를 나타내며, 방향성을 갖는다. RDF는 이와 같이 주어, 술어, 목적어로 구성된 선언문 형태를 가지고 있으며, 이는

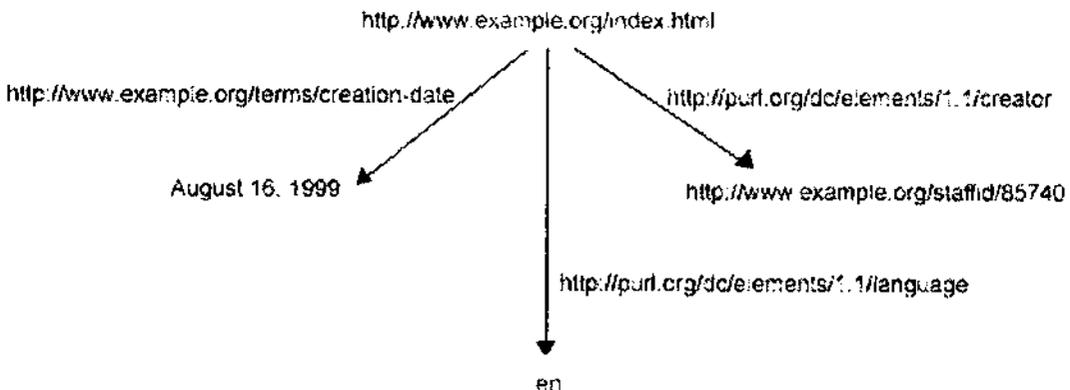
다음과 같이 Triple 형태로 표현할 수 있다.

```
<http://www.example.org/index.html>
<http://purl.org/dc/elements/1.1/creator>
  <http://www.example.org/staffid/85740>
<http://www.example.org/index.html><http://www
.example.org/terms/creation-date>
  "August 16, 1999" .
<http://www.example.org/index.html><http://www
.example.org/terms/language> "en" .
```

Triple을 작성하기 위해서 참조되는 전체 URI를 포함하도록 작성해야 하지만 편의상 네임스페이스를 정의해서 다음과 같은 단축형을 사용할 수도 있다.

```
ex:index.html dc:creator ex:staff:85740 .
ex:index.html ex:terms:creation-date "August 16, 1999" .
ex:index.html dc:language "en" .
```

RDF의 모든 선언문은 이와 같이 주어, 술어, 목적어로 구성된 Triple 형태로 이루어져 있다. RDF 문서를 작성하기 위해 XML Schema의 데이터 타입과 공백 노드(Blank Node), RDF 스키마(RDF Schema)와 같이 다양한 개념이 사용되며, 이에 대해서는 따로 언급하지 않도록 하겠다.



〈그림 2〉 트리플 문장의 RDF 그래프

4. 효율적인 검색을 위한 RDF 저장 구조

4.1 Triple 저장 구조

RDF는 정형화된 어휘들(vocabularies)을 기술하기 위한 선언 형식의 표현법이다. RDF를 그래프 형태로 표현한 RDF 그래프는 주어, 술어, 목적어로 이루어진 Triple의 집합으로 표현된다.

Triple을 이루고 있는 주어, 술어, 목적어 노드는 각각 URI나 리터럴 중 한 가지 형태를 갖는다. 또한 리터럴은 데이터 타입이 정의되지 않은 문자열인 Plain 리터럴(Plain literal)과 데이터 타입이 부여된 Typed 리터럴(Typed literal)로 구분되며, Typed 리터럴의 경우 XML Schema의 일부 데이터 타입을 따른다.

이를 기반으로 설계하고 구현된 SMARTTriple 시스템에서 사용하는 개략적인 데이터 구조는 다음과 계층적 관계를 갖는다.

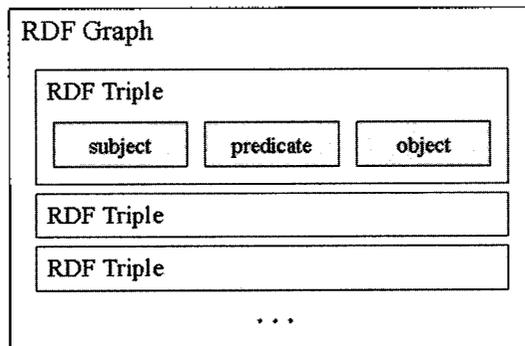
RDF 문서로부터 추출된 Triple들의 집합은 SMARTModel로 관리되며 즉, 이것은 RDF Triple이 모여서 만들어진 하나의 RDF 문서를 나타낸다. 각각의 Triple들은 SMARTIndexMap에 의해 인

텍싱된 구조로 메모리에 저장되며, Triple은 SMARTNode 타입의 주어, 술어, 목적어로 구성된다. SMARTNode는 그 종류에 따라 URI나 Literal 혹은 Blank로 생성되며, Literal의 경우 XML Schema의 데이터 타입 분류에 따라 크게 네 가지 타입으로 구분된다.

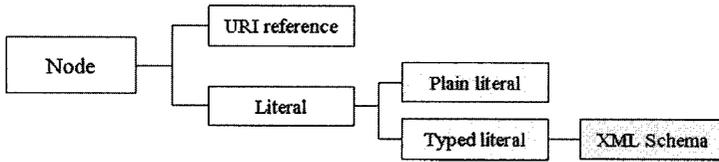
이와 같이 본 SMARTTriple 시스템은 RDF를 기반으로 설계 및 구현함으로써 RDF 시맨틱을 포함한 RDF의 확장 및 OWL과 같은 상위 온톨로지(Ontology) 언어를 반영할 수 있도록 하였으며, 이를 기반으로 시맨틱 웹의 핵심 역할 중 하나인 온톨로지 추론과 규칙 추론 기능이 반영 가능하도록 설계 및 구현하였다.

4.2 검색을 위한 인덱스 구조

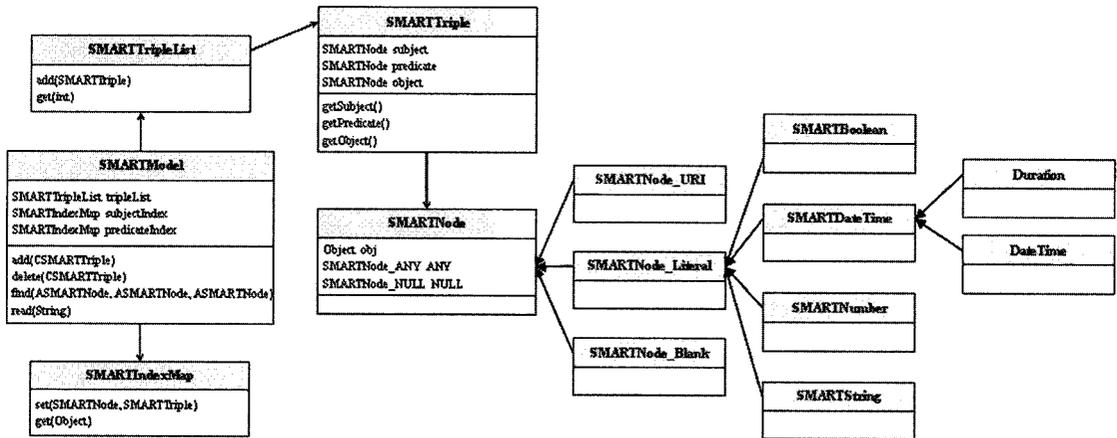
본 절에서는 Triple을 보다 신속하게 검색하기 위한 인덱스 맵 구조에 대해 설명한다. 인덱스 구조란 특정 값(value)의 검색 효율을 높이기 위해 해당 값의 키(key)와 그 값의 위치를 쌍으로 저장하는 데이터 저장 구조를 말하며, 키와 값의 연결(Mapping)을 표현한 것이 인덱스 맵이다. 일반적인 인덱스 구조의 검색은 키의 입력을 통해 이루어진다. 키가 입력되면 인덱스 맵 내에 존재하는



〈그림 3〉 RDF Triple의 저장 구조



〈그림 4〉 RDF Node의 데이터 타입



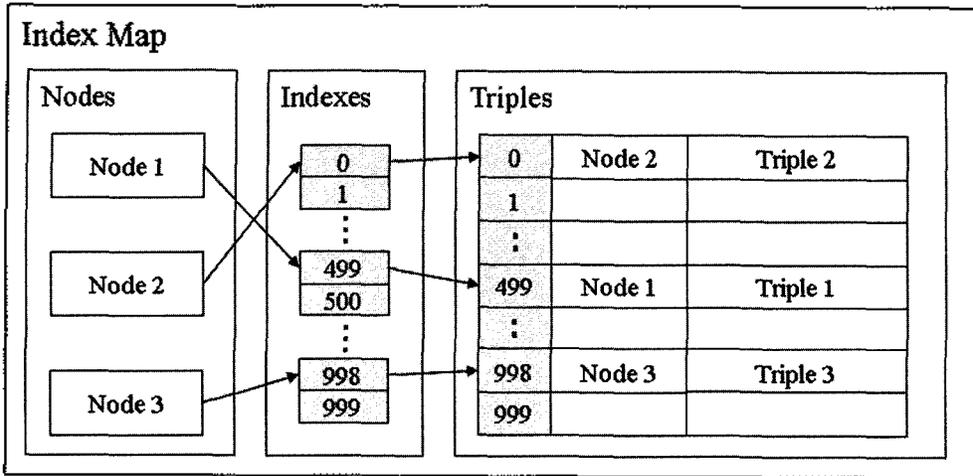
〈그림 5〉 SMARTTriple의 RDF Triple 데이터 구조

키와 인덱스의 쌍을 검색한 후 해당 인덱스와 연결(mapping)된 값에 직접 접근하여 원하는 값을 찾게 된다. 본 논문에서 Triple의 신속한 검색을 위해 구성한 인덱스 맵의 구조는 다음과 같다.

특정 Triple에 대해 그 Triple을 구성하는 노드를 키로 설정하고 해당 Triple을 값으로 설정하여, 키와 Triple의 위치를 나타내는 인덱스를 저장하는 인덱스 맵을 구성한다. 또한 필요에 따라 여러 개의 인덱스 맵을 구성하여 보다 효율적인 검색이 가능하다. 이러한 인덱스 구조가 없을 경우에는 찾고자 하는 Triple을 저장되어 있는 수많은 Triple과의 비교를 통하여 그 위치 및 내용을 검색하게 되는데, 특정 값을 갖는 노드를 포함한 Triple을 검색하려는 RDF 메타데이터 검색 시스템에서는 검색 해당 노드를 포함한 Triple들의 위치를 인덱스 맵

구조를 이용하여 바로 알아낼 수 있기 때문에 Triple의 노드 값을 저장하는 테이블의 모든 값들을 일일이 찾을 필요 없이 찾고자 하는 값의 위치로 바로 접근이 가능하다. Triple의 수가 많아질수록 인덱스 맵으로 인한 검색 속도 및 효율성의 향상은 극대화된다.

이와 같은 인덱스 맵을 사용함으로써 Triple에 저장하는 노드 데이터를 문자열이 아닌 인덱스 맵의 인덱스 값으로 저장할 수 있게 된다. 이러한 특징은 Triple의 검색 뿐만 아니라 Triple의 비교를 위한 노드 비교 속도의 향상시켜 SMARTTriple 시스템의 전체적인 속도를 향상시키는 추가적인 장점이 된다.



〈그림 6〉 SMARTTriple의 인덱스 맵 구조

5. 메타데이터 검색 시스템 : SMARTTriple

5.1 시스템 구조

본 논문에서 제안하고 있는 RDF 저장 시스템인 SMARTTriple은 앞서 살펴본 것과 같이 RDF와 RDF 시맨틱의 구조에 따라 설계되었다. 4장에서 제안한 Triple 저장 구조에 따른 SMARTTriple 시스템의 구조는 다음과 같다.

SMARTTriple 시스템은 크게 두 개의 시스템으로 구성된다. RDF 문서를 읽고 RDF 구조를 만들기 위한 RDF Loader와 노드와 Triple을 생성해서 메모리에 저장하기 위한 Graph Builder. 그리고 사용자의 질의를 처리하기 위한 Query Processor로 구성된다.

RDF Loader부터 살펴보면, RDF 문서를 처리하기 위해 기본적으로 SAX 파서가 이용된다. SAX 파서는 이벤트를 기반으로 한 XML 파서로서 한 번 XML 문서를 처리할 경우 메모리의 제약 없이 아주 빠르게 처리할 수 있는 장점을 제공한다.

SAX 파서를 거쳐 만들어진 토큰(token)은 RDF Lexer를 통해 RDF의 구조에 따라 생성된다. 생성된 토큰은 Graph Builder를 통해 노드로 만들어지게 되는데 노드의 종류, 즉 URI 노드인지 리터럴 노드인지에 따라 각각의 타입에 맞게 생성된다. 이때 Duplicator Checker를 통해 노드의 중복성을 해결하고 중복성이 해결된 노드는 Triple로 생성된다. 생성된 Triple은 메모리에 삽입되기 전 빠른 검색을 위한 인덱싱 과정을 거치게 된다. 이와 같은 과정을 통해 비로소 하나의 완전한 RDF 저장 구조를 갖추게 된다.

시맨틱 웹 애플리케이션 개발자는 SMARTTriple 시스템에서 제공하는 질의 API를 이용해서 Triple에 대한 질의를 할 수 있으며, 이때 인덱스를 이용한 저장 구조를 검색함으로써 검색 성능의 향상을 가져올 수 있다.

5.2 메타데이터 검색

SMARTTriple 시스템에서는 앞서 소개한 인덱스 구조를 사용함으로써 Triple에 대한 검색 속도의

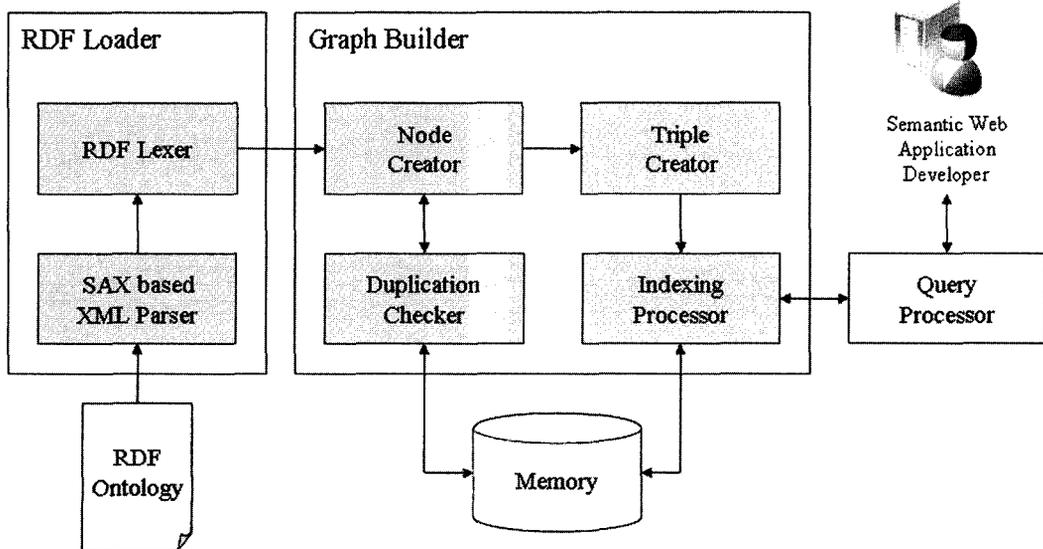
향상시킬 수 있다. SMARTTriple 시스템에서의 인덱스 구조는 Triple의 구성 요소를 이루는 각 노드를 키로 하고 Triple을 값을 갖는 인덱스 맵을 사용한다. 키로 사용되는 각 노드는 Triple로의 연결을 위한 인덱스 값을 갖고 있으므로, SMARTTriple 시스템에서는 URI 혹은 Literal을 이용해서 필요한 노드를 생성한 후에 Triple을 구성하여 검색을 할 수 있다. 이때 Triple을 각 자리에 변수를 사용함으로써, 여러 개의 결과 Triple을 포함하는 Triple Set을 검색·질의하는 것도 가능하다.

사용자로부터 SMARTTriple 시스템의 API를 통하여 Triple에 대한 질의가 올 경우 Indexing Processor는 변수를 제외한 노드를 이용해 인덱스 맵을 검색하게 된다. 이때 인덱스 맵을 이용하는 경우는 subject와 predicate이 검색어(상수)인지 변수인지에 따라 모두 네 가지로 나뉘게 된다. 둘 중에 하나라도 검색어가 있는 경우에는 해당 검색어를 이용하여 인덱스 맵을 검색하게 되고, 둘 다 변수

인 경우에는 인덱스 맵을 이용할 수 없게 된다. 인덱스 맵을 이용하는 경우에도 subject와 predicate이 모두 검색어(상수)인 경우에는 인덱스 맵을 2번 이용함으로써 찾고자 하는 Triple에 대한 검색 범위를 더욱 줄일 수 있게 된다. 인덱스 맵으로부터 Triple에 대한 인덱스 값을 얻은 후 이렇게 얻은 인덱스 값을 이용해서 사용자가 질의한 Triple을 찾아 Triple Set을 구성해 결과로 반환하게 된다. 이와 같이 인덱스 구조를 사용해서 검색을 함으로써 전체 Triple을 비교하지 않고도 인덱스 값을 이용해서 특정 Triple이 저장된 위치로 바로 접근 가능하기 때문에 타 시스템에 비해 보다 빠른 검색이 가능하다.

5.3 실험 및 성과비교

본 SMARTTriple 시스템의 성능 평가를 위하여 대표적인 RDF 저장 엔진인 Jena[8]와 KAON2[11]



〈그림 7〉 SMARTTriple 시스템 구조도

를 비교하였다. Jena는 시맨틱 웹 애플리케이션을 개발하기 위한 자바 프레임워크로써 RDF, RDFS와 OWL, 그리고 SPARQL을 지원하며 물 기반의 추론 엔진을 포함하고 있다. Jena는 현재 170,000회 이상의 다운로드 회수를 기록하고 2004년 2월까지 주요 컨퍼런스에서 발표되고 있는 시맨틱 웹 애플리케이션의 80%가 Jena를 기반으로 하고 있으며 [5]. 2006년에는 영국의 Bristol에서 Jena 사용자 컨퍼런스[9]가 개최될 정도로 널리 사용되고 있는 기반 시스템이다. KAON2는 RDF 기반의 온톨로지 엔진인 KAON(Karlsruhe ontology) 프로젝트[10]로부터 출발한 온톨로지 엔진으로써, KAON이 RDF 기반의 엔진인 반면에 KAON2는 OWL-DL 기반의 온톨로지 엔진이다. KAON2는 유럽의 IST에서 진행하고 있는 시맨틱 웹 관련 프로젝트의 기반 기술로 이용되고 있으며, 그 밖의 다양한 시맨틱 웹 관련 분야에서 활용 중에 있다.

SMARTTriple 시스템의 상대적 성능 평가를 위하여 사용된 시스템의 사양은 <표 1>과 같다.

SMARTTriple의 성능 평가는 RDF 저장 엔진의 핵심 기능인 온톨로지의 로딩(loading) 및 Triple의 추가와 질의를 대상으로 평가를 시행하였다. 온톨로지의 로딩 평가는 데이터베이스를 사용하지 않고 메모리를 기반으로 한 동일 환경 하에서 대용량 온톨로지를 어느 수준까지 메모리에 적재할 수 있는지 평가하였다. 평가 기준은 시맨틱 웹 리포지터리(Semantic Web repository)를 평가하기 위해 Lehigh University에서 만든 LUBM(the Lehigh

University Benchmark)[12]을 이용하였다. LUBM은 온톨로지 생성 옵션에 따라 다양한 크기의 온톨로지를 자동 생성할 수 있다. LUBM을 이용해서 생성한 온톨로지의 구성은 <표 2>와 같다.

위 온톨로지를 기반으로 평가한 결과는 <그림 8>과 같다.

결과를 살펴보면 세 개의 시스템 중 SMARTTriple이 다른 시스템에 비해 로딩 시간이 적게 걸린다는 것을 알 수 있다. 메모리를 사용하는 측면에서 바라본다면 SMARTTriple 시스템은 LUBM(4, 0)까지 메모리에 로딩할 수 있었으며, Jena는 LUBM(3, 0)까지, 그리고 KAON2는 LUBM(1, 0)까지 로딩할 수 있었다. 따라서 SMARTTriple이 다른 시스템에 비해 메모리를 보다 효과적으로 사용하고 있음을 알 수 있다.

다음으로 Triple의 추가와 질의에 대한 성능 평가는 정보통신부에서 발주된 시맨틱 웹 관련 프로젝트의 2005년 1차년도 산출물인 영화 온톨로지를 대상으로 하였다. 영화 온톨로지는 국내 상영 중인 영화에 대한 정보를 담고 있는 온톨로지로서, 자세한 구성은 <표 3>과 같다.

Triple 추가에 대한 시스템의 성능 평가는 위해 영화 온톨로지를 로딩한 상태에서 10,000 개의 Triple을 추가하는데 소요되는 시간을 비교하였다.(<표 4> 참조)

결과에 따르면 SMARTTriple 시스템에서 Triple을 추가하는데 소요된 시간이 Jena와 KAON2에 비해 월등히 적게 소요됐음을 알 수 있다.

<표 1> 성능 평가를 위한 시스템 사양

| 구분 | 시스템 사양 |
|------------------|--|
| Processor | AMD Athlon(tm) 64 Processor 3000+, MMX, 3DNow, ~1.8GHz |
| Memory | 1024MB RAM |
| Operating System | Microsoft Windows XP Professional |

〈표 2〉 LUBM의 온톨로지 구성

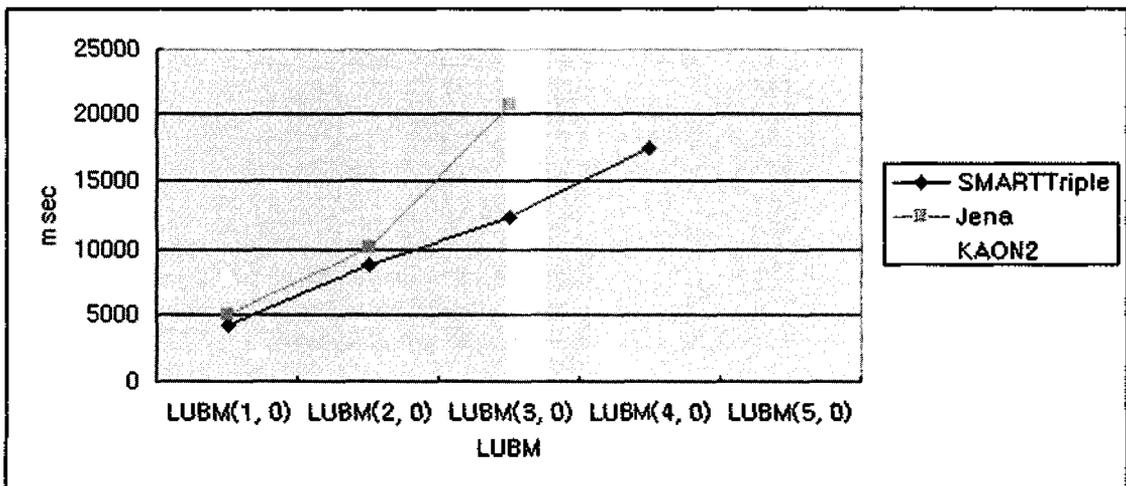
| LUBM | 크기 | Triple의 수 |
|------------|--------------|-----------|
| LUBM(1, 0) | 8,215 KByte | 103,076 개 |
| LUBM(2, 0) | 18,915 KByte | 237,144 개 |
| LUBM(3, 0) | 27,756 KByte | 348,008 개 |
| LUBM(4, 0) | 39,396 KByte | 493,704 개 |
| LUBM(5, 0) | 51,333 KByte | 645,651 개 |

마지막으로 로딩된 온톨로지로부터 질의를 하는데 필요한 시간을 비교해 보았다. 측정은 두 가지 방법으로 이루어졌으며, 첫 번째는 온톨로지로부터 하나의 Triple을 질의하는데 소요되는 시간, 두 번째는 전체 Triple을 질의하는데 소요되는 시간을 측정하였다. 각각 정확한 시간 측정을 위해 Triple 하나에 대한 질의는 10,000 번, 전체 Triple의 질의는 100 번을 수행하였으며 이에 따른 평가 결과는 〈표 5〉와 같다.

결과에서 보는 것과 같이 Triple을 질의하는데 있어서 SMARTTriple 시스템이 타 시스템에 비해 좋은 성능을 나타내고 있다는 것을 알 수 있다. 또

한 질의하려는 Triple의 개수에 따른 질의 소요 시간을 비교해보면 질의하려는 Triple의 수가 늘어남에 따라 Jena와 KAON2가 SMARTTriple에 비해 높은 비율로 증가한다는 것을 파악할 수 있다.

이상의 성능 평가 결과에 따르면 본 논문에서 제안하고 있는 SMARTTriple이 타 시스템인 Jena와 KAON2에 비해 탁월한 성능을 가지고 있음이 입증되었다. 본 논문에서는 두 개의 시스템을 대상으로 성능 평가를 시행하였지만 보다 정확하고 객관적인 평가를 위해 다양한 RDF 저장 시스템과의 비교 평가가 필요할 것으로 생각된다.



〈그림 8〉 온톨로지 로딩 평가 결과

〈표 3〉 영화 온톨로지의 구성

| 온톨로지 | 크기 | Triple의 수 |
|---------|-------------|-----------|
| 영화 온톨로지 | 3,320 KByte | 37,842 개 |

〈표 4〉 온톨로지에 Triple 추가 성능의 평가 결과

| RDF 저장 시스템 | 소요 시간 |
|-------------|----------|
| SMARTTriple | 32 msec |
| Jena | 110 msec |
| KAON2 | 458 msec |

〈표 5〉 온톨로지로부터 Triple 질의 성능의 평가 결과

| RDF 저장 시스템 | 하나의 Triple에 대한 수행 시간 | 전체 Triple에 대한 수행 시간 |
|-------------|----------------------|---------------------|
| SMARTTriple | 78 msec | 187 msec |
| Jena | 198 msec | 5,281 msec |
| KAON2 | 110 msec | 11,375 msec |

6. 결 론

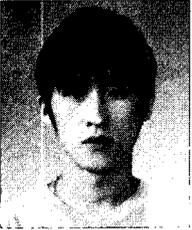
시맨틱 웹을 포함한 다양한 분야의 응용 시스템에서 기반으로 사용되고 있는 RDF 메타데이터의 검색 시스템의 성능을 기존의 시스템보다 향상시키기 위하여 인덱스 맵 구조를 이용한 새로운 RDF 메타데이터 저장 구조를 설계하고 이를 이용하여 RDF 메타데이터의 저장 및 검색 기능을 갖는 SMARTTriple 시스템을 개발하였다. 새로운 시스템의 성능을 평가하기 위하여 LUBM 온톨로지와 영화 온톨로지를 이용하여 로딩 성능, Triple 추가 성능, Triple 검색 성능을 비교해 본 결과, 기존의 대표적인 시스템에 비해 우수한 결과를 보여 주고 있다. 본 연구를 통해 개발된 SMARTTriple

시스템은 향후 RDF 메타데이터를 활용하는 시맨틱 웹 관련 응용 시스템의 기반 구조로 활용됨으로써 시맨틱 웹의 연구와 개발을 수행하는데 좋은 도구를 제공해 줄 것이다.

참 고 문 헌

- [1] David Beckett, "The Design and Implementation of the Redland RDF Application Framework." *Computer Networks*, Vol.39, No.5, pp.577-588, 2002
- [2] Jeen Broekstra, Arjohn Kampman and Frank van Harmelen, "Sesame: A Generic Architecture for Storing and Querying RDF and RDF Schema," *International Semantic Web Conference 2002, Sardinia Italy, 2002*
- [3] Andreas Harth and Stefan Decker, "Optimized Index Structures for Querying RDF from the Web," *the Third Latin American Web Congress, Argentina, October 2005*
- [4] Maciej Janik and Krys Kochut, "BRAHMS: A WorkBench RDF Store And High Performance Memory System for Semantic Association Discovery," *the Fourth International Semantic Web Conference ISWC 2005, Galway, Ireland, November 2005*
- [5] Martin Merry, HP releases new version of leading Semantic Web developers' toolkit http://www.hpl.hp.com/news/2004/jan-mar/jena21.html?jumpid=reg_R1002_USEN
- [6] Ora Lassila and Ralph R. Swick, "Resource Description Framework(RDF): Model and Syntax Specification." Recommendation, World Wide Web Consortium, Feb. 1999, <http://www.w3.org/TR/REC-rdf-syntax>
- [7] AllegroGraph. <http://www.franz.com/products/allegrograph/>
- [8] Jena, <http://jena.sourceforge.net/>
- [9] Jena User Conference 2006, <http://jena.hpl.hp.com/juc2006/proceedings.html>
- [10] KAON. <http://kaonsemanticweb.org/>
- [11] KAON2. <http://kaon2.semanticweb.org/>
- [12] LUBM. <http://swat.cse.lehigh.edu/projects/lubm/>
- [13] Semantic Web Development Tools, <http://csw.w3.org/topic/SemanticWebTools>, W3C Wiki

저 자 소 개

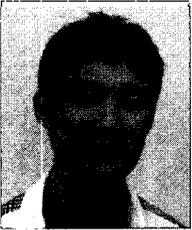


박지형

(E-mail : zensents@yonsei.ac.kr)

현재
관심분야

한성대학교 산업공학및 컴퓨터시스템공학 전공
연세대학교 정보산업공학과 지능정보시스템 연구실 석사과정
시맨틱 웹 서비스 시맨틱 웹 포털



이명진

(E-mail : xml@yonsei.ac.kr)

현재
관심분야

'자바 개발자를 위한 XML프로그래밍', '소셜같은 XML과 XML
Web Services' 집필
연세대학교 정보산업공학과 지능정보시스템 연구실 박사과정
XML과 XML Web Services, Semantic Web



홍준석

(E-mail : junehong@kyonggi.ac.kr)

현재
관심분야

서울대학교 경영학과 졸업
한국과학기술원(KAIST) 경영학과 (석사)
테크노경영대학원 (박사)
경기대학교 경영학부 부교수
지능형에이전트, 자동협상시스템, 전자상거래, 시맨틱 웹,
Description Logics