

관계형 데이터베이스 기반 구조적학술용어사전 (STNet)의 RDF 온톨로지 변환 방식 연구*

A Study on Conversion Methods for Generating RDF Ontology from Structural Terminology Net (STNet) based on RDB

고영만 (Young Man Ko)**

이승준 (Seung-Jun Lee)***

송민선 (Min-Sun Song)****

초 록

본 연구에서는 R2RML 방식과 Non-R2RML 방식을 각각 적용하여 RDB를 RDF 온톨로지로 변환한 결과를 비교하였다. RDB 기반의 구조적학술용어사전 데이터베이스인 STNet의 데이터를 대상으로, 변환이 완료된 데이터의 규모, 튜플당 변환에 걸리는 시간, 그리고 질의 응답 속도를 측정하였다. 변환 규모의 평가 결과 Non-R2RML 방식이 더 많은 수의 변환을 하였으며, 표현의 풍부성과 추론 가능성 정도를 높이는 변환을 수행한 것으로 나타났다. 튜플당 변환 시간의 경우 Non-R2RML 방식이 미세하지만 더 빠른 것으로 나타났으며, 질의 응답 속도는 두 방식 모두 300회 이상의 질의 횟수부터는 안정적인 성능을 보이면서 유사한 형태의 속도를 보였다. 측정에 대한 종합적 검토 결과 데이터의 변형이 빈번하고 새로운 데이터의 추가나 데이터들 간의 연결관계가 지속적으로 변화하는 STNet과 같은 동적인 RDB에는 Non-R2RML 방식이 적절한 것으로 평가되었다.

ABSTRACT

This study described the results of converting RDB to RDF ontology by each of R2RML method and Non-R2RML method. This study measured the size of the converted data, the conversion time per each tuple, and the response speed to queries. The STNet, a structured terminology dictionary based on RDB, was served as a test bed for converting to RDF ontology. As a result of the converted data size, Non-R2RML method appeared to be superior to R2RML method on the number of converted triples, including its expressive diversity. For the conversion time per each tuple, Non-R2RML was a little bit more faster than R2RML, but, for the response speed to queries, both methods showed similar response speed and stable performance since more than 300 numbers of queries. On comprehensive examination it is evaluated that Non-R2RML is the more appropriate to convert the dynamic RDB system, such as the STNet in which new data are steadily accumulated, data transformation very often occurred, and relationships between data continuously changed.

키워드: 구조적학술용어사전, STNet, 의미적 연결관계, 관계 유형, RDF 온톨로지, 온톨로지 언어, 관계형 데이터베이스, R2RML, D2R 서버
structural terminology net, STNet, semantic relation, relation type, RDF ontology, ontology language, relational database, R2RML, D2R server

* 이 논문은 2012년 정부(교육부)의 재원으로 한국연구재단의 지원을 받아 수행된 연구임 (NRF-2012S1A5B4A01035131).

** 성균관대학교 문과대학 문헌정보학과 교수(ymko@skku.edu) (제1저자)

*** 성균관대학교 정보관리연구소 연구원(dino@skku.edu) (교신저자)

**** 성균관대학교 정보관리연구소 연구원(songser@skku.edu) (공동저자)

■ 논문접수일자: 2015년 5월 27일 ■ 최초심사일자: 2015년 5월 27일 ■ 게재확정일자: 2015년 6월 19일
■ 정보관리학회지, 32(2), 131-152, 2015. [http://dx.doi.org/10.3743/KOSIM.2015.32.2.131]

1. 서론

1.1 연구의 목적

웹 상에서 온톨로지를 구축하고 배포할 수 있도록 하기 위한 ‘웹온톨로지 언어 표준’으로 RDF(Resource Description Format), RDFs(RDF Schema), OWL(Web Ontology Language) 등이 W3C에 의해 제안되었다. RDF는 정보자원과 속성, 속성 값의 트리플 구조를 사용하여 데이터를 정의하고 기술하며, RDFs는 RDF를 프레임 지식표현 패러다임으로 확장하여 속성의 도메인을 제한하거나 비슷한 자원을 한 데 묶어 클래스로 표현할 수 있게 한다. OWL은 RDFs에 비해 보다 다양하게 속성의 범위를 지정하고 또 특별한 속성을 부여함으로써 RDFs가 가지는 표현력과 추론 능력을 확장한 것이다.

이러한 도구를 사용하여 처음부터 온톨로지를 구축하는 작업은 도메인 전문가가 필요할 뿐만 아니라 많은 시간과 노력이 요구된다. 따라서 최근에는 이미 구축된 데이터베이스로부터 온톨로지를 생성하는 작업과 연구들이 활발하게 진행되었다. 특히 데이터의 구조가 복잡하고 방대한 양의 데이터가 저장된 경우 일반적으로 ‘관계형 데이터베이스(Relational Database, 이하 RDB)’의 데이터를 활용하여 RDF를 구성하는 것이 별도의 온톨로지 데이터를 OWL로 구축하여 시맨틱 웹을 작성해 가는 방식에 비해 효율적일 수 있기 때문이다.

RDB 데이터를 RDF 형태의 온톨로지로 변환하는 것은 두 모형의 요소 간 매핑을 정의한 매핑 규칙에 따라 이루어지며, 일반적으로 RDB의 테이블을 RDF의 클래스로 매핑하고 RDB

의 컬럼 및 속성은 RDF의 속성으로 매핑한다. 이를 표현하기 위한 방법론은 대체로 W3C에서 제안한 매핑언어 RDB2RDF Mapping Language(이하 R2RML)를 사용하는 방식과 독자적인 매핑언어를 사용하는 Non-R2RML 방식으로 구분된다. R2RML 방식은 매핑언어 중심 방법과 RDB2RDF 연결도구 활용 방법으로 다시 나눌 수 있다. R2RML을 기반으로 하여 개발된 연결도구로는 DB2Triples, Morph-RDB 등이 있으며, Non-R2RML 방식의 연결도구로는 Asio Semantic Bridge와 D2R 서버, DB2OWL, Virtuoso 등이 있다(Michel, Montagnat, & Faron-Zucker, 2014). 이와 관련하여 Hert, Reif, & Gall(2011)은 8개의 매핑언어(eD2R, R2O, Relational OWL, Virtuoso, DeRQ, Triplify, R2RML, R3M)를 15개 항목으로 측정하여 비교한 결과를 발표한 바 있다([부록] 참조).

R2RML 방식을 사용하는 경우 매핑언어 중심 방식이나 연결도구 활용방식 모두 RDB로부터 대상 데이터를 추출하여 별도의 RDF 저장영역에 저장된 데이터를 대상으로 시맨틱 웹을 구현하게 된다. Non-R2RML 방식의 경우에는 R2RML 방식과는 다르게 Oracle, PostgreSQL, HSQLDB 등의 다양한 RDB 시스템에 대한 자동 변환 매핑 파일을 제공하며, 이 매핑 언어를 통해 RDB에 접근하고 스키마 정보에 대한 매핑 정보를 자동으로 생성한다. 특히 Non-R2RML 방식의 연결도구인 D2R 서버는 생성된 매핑 정보를 참고하여 RDB의 데이터를 RDF 형태의 온톨로지로 변환하며, 이 매핑 정보를 참고하여 SPARQL을 SQL로 변환한다. 질의문의 변환에 의한 매핑 모델은 데이터의 최신성을 보장하며 추가적인 트리플 저장소가 필요하지 않

다. 따라서 데이터의 속성 정보가 다양하고 데이터의 변형과 데이터 간의 연결관계가 지속적으로 변화하는 동적인 RDB에 적용할 수 있는 Non-R2RML 방식의 변환 모형들이 제안되었다(Bizer, 2003; Saboo et al., 2009; 성하정 외, 2014).

RDB를 온톨로지로 변환시키는 것과 관련하여 이경호와 이준승(2006)이 XML문서의 데이터를 분석해 유사도를 기반으로 단말 노드 간 매칭을 계산하고 문맥 정보를 반영하는 제안된 경로 유사도 비교를 통해 후보매칭 관계 중에서 최종 매칭 결과를 선택하는 방식의 온톨로지 변환 모형을 제시한 바 있으며, 박여삼, 장옥배, 한성국(2008)은 XTM 1.0에 기반해 레거시 데이터를 토픽맵으로 변환한 후 온톨로지를 구축하는 모형을 제시하였다. 최미영과 문창주(2012)는 RDB의 테이블 데이터를 RDF/RDFs로 변환하는 모형을 제시하였으며, 최지웅과 김명호(2014)는 단일 연결 구성을 가진 레거시 RDB로부터 OWL 온톨로지를 자동으로 생성할 수 있는 규칙을 제안하였고, 성하정 외(2014)는 RDB 구성요소의 의미관계를 고려한 RDB to RDF 매핑시스템을 제안하였다. 그러나 이경호와 이준승의 연구는 노드의 경로가 유사할 경우 유사도 계산이 어려워 XML 문서 생성시 데이터 중복을 방지하는 별도의 정제 작업이 필요하다는 문제점이 있으며, 박여삼, 장옥배, 한성국의 연구는 레거시 데이터가 실시간으로 온톨로지를 생성하지 못하는 한계를 가지고 있다. 최미영과 문창주의 연구는 관계가 맺어진 엔터티의 널(Null) 데이터를 삭제해야 하는 추가작업을 필요로 하는 등의 문제가 있으며, 최지웅과 김명호의 연구는 복잡한 관계로

구성된 데이터에 적용하기 어렵다. 성하정 외의 연구는 표준화된 매핑언어인 R2RML을 사용하여 언어의 재사용성을 높이고 RDB에 저장된 데이터를 단순히 RDF로 변환하는 R2RML을 기능적으로 보완하였다는 점에 의의가 있으나, 저장된 데이터의 변형과 연결관계가 지속적으로 변화하는 RDB 데이터를 대상으로 적용한 것은 아니라는 한계를 갖는다.

본 연구는 데이터의 변형이 빈번하고 새로운 데이터의 추가나 데이터들 간의 연결관계가 지속적으로 변화하는 관계형 데이터베이스인 “구조적학술용어사전(Structural Terminology Net: 이하 STNet, <http://stnet.re.kr>)”을 대상으로, Non-R2RML 방식을 적용하여 RDB 데이터를 RDF 형태의 온톨로지로 변환한 데이터가 R2RML 방식을 적용한 변환 데이터에 비해 실제로 풍부한 의미정보를 표현하고 있는지를 확인하고자 하며, 또한 튜플 당 변환 시간과 질의응답 시간을 측정하여 R2RML 방식과 어느 정도 차이가 나는지를 비교해 보고자 하였다.

본 연구는 메타데이터 형태로 구축된 RDB의 데이터를 시멘틱 웹에서 사용할 수 있도록 하기 위해 필수적으로 거쳐야 하는 RDB의 온톨로지 변환 과정에 관한 연구이다. 따라서 본 연구는 R2RML 방식과 Non-R2RML 방식의 기술적, 논리적 차이를 밝히기 위한 연구라기 보다는, RDB 기반의 “STNet”을 사례로 Non-R2RML 방식을 적용하는 것이 풍부한 의미관계 표현 뿐 아니라 변환 속도 측면에서도 적절한지를 확인하기 위한 연구에 해당한다고 할 수 있다. 즉 데이터의 변형이 빈번하고 새로운 데이터의 추가나 데이터들 간의 연결관계가 지속적으로 변화하는 동적인 RDB의 경우 스키

마에만 의존하는 R2RML 방식에 비해 레이블의 속성 정보까지 변환하는 Non-R2RML 방식을 적용하는 것이 풍부한 의미관계 표현 뿐 아니라 변환 속도 측면에서도 어떠한 결과를 나타내는지 검증하기 위한 것이다. 특히 본 연구는 테스트베드를 따로 만들어 실험을 한 것이 아니라, 실제 구축된 데이터베이스를 대상으로 데이터를 변환하고 측정할 결과를 제시하였다. 따라서 본 연구 결과는 실제 운용되는 시소러스와 같은 지식조직 데이터베이스, 학술정보 메타데이터, 링크드데이터를 시멘틱 웹으로 진전시키는데 있어서 많은 기여를 할 수 있을 것이다.

1.2 연구 방법

R2RML 방식의 변환에는 RDB의 데이터를 추출하여 RDF 구조로 연결시키는 RDB to RDF 매핑과 매핑을 통해 추출된 R2RML 파일을 해석하는 R2RML 분석기, 매핑된 규칙을 통해 RDB 데이터를 읽어오는 SQL 생성기, RDB 질의 결과를 RDF로 변환하는 RDB to RDF 변환기를 사용한다. 이렇게 변환된 결과는 최종적으로 트리플 스토어에 저장되어 SPARQL을 통한 질의가 가능해진다. Non-R2RML 방식의 변환에는 D2R Query(이하 D2RQ)를 통해 RDB 스키마에서 매핑 규칙을 생성하고, 매핑 규칙을 통해 추출된 매핑 파일을 SQL 쿼리로 매핑하는 D2RQ 엔진을 사용하여 데이터베이스 테이블 단위 혹은 view 단위로 RDF를 생성한다.

질의 실행을 위해 R2RML 방식에서는 Virtuoso를 사용하였으며, Non-R2RML 방식에서

는 D2RQ를 사용하였다. Virtuoso와 D2RQ를 사용한 이유는 Hert, Reif, & Gall(2011)의 연구에서 두 방식의 매핑언어 모두 스키마 수준에서의 트리플 생성을 지원하는 F13 'static metadata' 부문과 RDF 그래프를 지원하는 F10 'named graphs' 부문을 제외한 다른 13개 분야는 모두 지원하는 것으로 평가되어 유사한 기능을 갖춘 것으로 나타났기 때문이다([부록] 참조).

변환 데이터의 규모 측정은 RDB 데이터의 테이블 명과 컬럼 명 사이에 관계정보가 추가되어 변환된 결과를 측정하는 것으로, 구체적으로는 매핑 파일이 RDF로 표현될 수 있는 표현의 개수를 측정한다. RDB의 데이터는 동일하지만 레이블 속성을 변환 데이터에 추가하는 것은 변환 방식에 따라 그 결과가 다를 수 있기 때문에 표현 개수의 측정을 통해 RDF 데이터에 포함된 표현의 풍부성과 추론 가능성 정도를 판단해 볼 수 있다. 레이블의 속성 정보가 변환 데이터에 추가될 경우 변환 데이터의 규모가 커지고 표현의 풍부성이 좋아지며, 이에 따라 추론 가능성 또한 커진다(성하정 외, 2014).

변환 시간과 질의 응답 시간의 측정은 STNet과 같은 실제 구현된 동적 RDB를 대상으로 Non-R2RML 방식과 R2RML 방식을 적용했을 때 시스템 성능 면에서 어떤 차이를 보이는지 확인하기 위한 것이다. 변환 시간의 비교를 위해 테스트 데이터의 튜플당 RDF 트리플 변환 시간을 측정하였으며, 질의 응답 시간은 변환된 RDF 데이터에 SPARQL Endpoint를 통하여 전달한 동시 접속자의 쿼리의 응답 속도를 측정하였다.

2. 구조적학술용어사전 “STNet”

2.1 STNet의 데이터 서술구조 및 현황

STNet은 인문학, 사회과학 및 예술체육 분야의 학술용어를 유사한 속성을 가진 클래스(개념범주)에 따라 분류하고, 그 클래스가 가지는 속성을 체계화 한 다음, 그 속성에 따라 해당 학술 용어 하나하나의 의미를 정의하고, 또 관계명을 통해 용어, 클래스, 속성 사이의 의미 관계를 서술한 데이터베이스 시스템이다. STNet은 한국학술지인용색인(Korea Citation Index, 이하 KCI)에 등재된 인문학, 사회과학, 예술체육 분야 논문 중 2007년에서 2012년 9월까지 발행된 245,409편에 수록된 저자키워드 약 1백만 개에서 불용어 등을 제외하고 출현빈도가 2

이상인 용어 약 5만 여개를 토대로 의미목록을 구축하였다. 의미목록을 구축하기 위한 메타데이터 항목은 크게 클래스 정보, 용어 정보, 관계 정보로 나누어진다. 속성값은 객체(object) 값과 텍스트(text) 값, 코드(code) 값으로 구분되며, 객체값은 대체로 용어로 표현된다(〈표 1〉 참조). 따라서 STNet의 용어는 해당 클래스의 속성을 통해서 표현되며, 정의나 외래어 등의 서술을 통해 용어 스스로의 정보가 풍부해지고, 관계명을 통해 다른 용어와 의미적 연결관계가 맺어지는 구조를 가진다(〈표 2〉 참조).

2015년 4월 30일 현재 STNet에는 개념속성이 부여된 49,675개 학술용어의 의미목록이 구축되어 있으며, 목록 구축과정에서 개념범주(클래스) 속성 값으로 입력된 용어 97,238개를 포

〈표 1〉 STNet의 메타데이터 구조

구분	항목	내용
클래스 정보	클래스명	STNet 클래스 텍사노미 기준
	클래스 속성	속성값 유형: 객체(object), 텍스트(text), 코드(code)
용어 정보	용어명	구축 대상 용어
	연구분야	용어가 속한 논문 게재 학술지의 분야(한국연구재단 연구분야분류표 기준)
	정의	구축 대상 용어의 정의(해당 논문, 전문용어사전 또는 백과사전 활용)
	동의어, 한자어, 외국어, 음차어	
관계 정보	관계명	구축 대상 용어가 다른 용어와 갖는 관계

〈표 2〉 STNet의 데이터 서술 구조

Subject	Predicate	Object
용어	클래스	속성
	속성	속성 값(객체, 텍스트, 코드)
용어	용어설명(연구분야, 정의, 동의어, 외국어 등)	속성 값
용어	관계명	용어

함하여 중복용어 10,270개를 제외한 총 136,643개의 용어가 수록되어 있다. STNet은 상위 클래스, 중위 클래스, 하위 클래스로 체계화된 텍사노미에 따라 용어에 클래스를 부여하고 있으며, 관계명을 통해 클래스와 클래스, 클래스와 속성, 속성과 다른 속성, 용어와 용어 사이의 의미적 연결관계를 서술하고 있다. 텍사노미의 클래스와 관계명을 활용하여 구축하여 조직된 STNet의 의미쌍은 총 322,979개이다(〈표 3〉 참조).

2.2 STNet의 클래스 텍사노미

STNet에서는 용어의 개념을 체계화한 클래스 텍사노미를 통해 각각의 용어에 클래스가 부여되고 해당 클래스의 속성에 따라 의미목록이 구축된다. 텍사노미는 최상위에 7개의 기본 클래스 그 아래에 28개의 중위 클래스, 그리고 120개의 하위 클래스(1)와 하위 클래스(2)로 구성되어 있다. 하위 클래스(2)는 하위 클래스(1)를 세분한 것이다. 클래스마다 각각의 코드와 클래스 명칭을 가지고 있다(〈표 4〉, 〈표 5〉 참조).

〈표 3〉 STNet 데이터 현황(2015년 4월 30일 기준)

데이터 유형	데이터 수
STNet 수록 용어	136,643
개념범주	176
개념범주별 속성	217
개념속성이 부여된 용어(의미목록이 구축된 용어)	49,675
개념속성 값으로 서술된 용어	97,238
관계유형	100
용어 관계쌍	322,979
연구분야 적용 용어	73,563
한자어	86,628
외국어	2,031
용어정의	67,226

〈표 4〉 STNet의 텍사노미 체계(예시)

기본 클래스		중위 클래스		하위 클래스	
코드	명칭	코드	명칭	하위 클래스(1)	하위 클래스(2)
A	개체	A01	인간	8개	19개
		A02	기관/조직	4개	
		A03	자연물	3개	
		A04	인공물	9개	
B	활동/변화	B01	행위/행동/역할	6개	
		B02	변화(행동)	6개	
C	특성/현상	C01	특성/성질	7개	
		C02	심리	2개	
		C03	현상/이슈	5개	

기본 클래스		중위 클래스		하위 클래스	
코드	명칭	코드	명칭	하위 클래스(1)	하위 클래스(2)
D	이론/방법	D01	이론(사상, 이념, 주의, 법칙)	4개	
		D02	제도(체제)	3개	
		D03	방법	4개	
		D04	기법/전략	4개	
E	형식/기준	E01	형식/유형/양식/장르	5개	
		E02	모형/기준	6개	
		E03	언어/각국어	2개	
		E04	공간	3개	
X	일반/공통	X01	지명	6개	
		X02	시대/시간	3개	
		X03	관계/상호작용	6개	8개
Y	인스턴스	Y01	인명	2개	
		Y02	창작물명	6개	
		Y03	사건명	4개	
		Y04	기념물(문화재)명	2개	
		Y05	법률/제도명	3개	
		Y06	기관/단체명	4개	
		Y07	상품명	3개	
		Y99	기타개체명		
Z99	기타주제어				
기본 클래스 7개		중위 클래스 29개		하위(1) 120개, 하위(2) 27개	

〈표 5〉 하위 클래스(1)과 하위 클래스(2) - 예시

하위 클래스 (1)		하위 클래스 (2)	
a01-04	인간(사회집단)	a01-04-01	민족/인종/종족
		a01-04-02	국민
		a01-04-03	인간(거주상황)
		a01-04-04	인간(사회계층)
		a01-04-05	인간(세대)
		a01-04-06	공동체

2.3 STNet의 관계 유형

STNet의 용어 간 의미적 관계는 크게 두 가지 종류로 나누어진다. 하나는 저자키워드와 그 저자키워드가 속한 클래스의 속성값이 연결되는 관계이며, 다른 하나는 저자키워드와 연관된 다

른 저자키워드 또는 저자키워드와 기타 연관된 학술용어 간에 맺어지는 관계이다. 2015년 4월 30일 현재, STNet에서 사용하고 있는 용어 단위 간 관계명은 역관계를 포함하여 총 100개이다(〈표 6〉 참조).

〈표 6〉 STNet의 용어 간 관계명 개요(2015년 4월 30일 기준)

관계유형		관계명(54개)	역관계명(46개)
계층관계	속-종	isA	includesSpecific
	부분 - 전체	BT	NT
		hasMember 외 5개	isMemberOf 외 5개
사례	hasInstance	isInstanceOf	
동의관계		LT	PT
		Use	UF
		hasSynonym	
반의관계		hasAntonym	
연관관계	개념적	RT_X	RT_Y
		RT	
		hasIssue 외 6개	isIssueIn 외 6개
	물리적	hasTributary	isTributaryOf
		isConnectedTo	
	기능적	apply 외 5개	isAppliedBy 외 5개
	공간적	traverses	isTraversedBy
		isAdjacentTo	
		surrounds	isSurroundsBy
	시간적	co-occurs-with	
		precedes	follows
	개념 속성 연관	hasLocation	
		hasEra	
		hasCreator 외 16개	hasWork 외 16개

3. STNet 데이터의 RDF 변환

3.1 RDF 변환을 위한 데이터 구조 설계 및 데이터 추출

3.1.1 RDF 변환 구조 설계

STNet의 데이터를 RDF로 변환하기 위해서는 STNet의 클래스와 함께 해당 클래스에 연결되어 나타나는 관계 유형 모두를 '주어부(Subject)-술어부(Predicate)-목적어부(Object)'의 트리플(Triple) 구조로 표현해 줄 수 있어야 한다. 이러한 구조 변환을 위해 STNet 시스템의 데이

터 변환 구조를 다음과 같이 설계하였다.

- 온톨로지 클래스(Class)와 속성(Property) 정의: STNet 시스템에서 이미 분류되어 있는 클래스가 owl:Class로의 대응이 가능하므로, 클래스에 오브젝트(object), 텍스트(text), 코드(code) 값으로 입력되는 개념 속성 필드 및 용어 단위 관계명을 온톨로지 속성 rdf:Property로 대응하여 변환할 수 있도록 하였다.
- 클래스 타입(rdf:type), 계층구조(rdf:subClassOf), 연결되는 속성(rdf:Property) 정의: STNet이 클래스 정의와 클래스 속성

값, 연관된 관계명 등의 데이터 입력값을 이미 가지고 있기 때문에 해당 내용을 트리플 구조에 대응하여 변환할 수 있도록 하였다.

- 온톨로지 속성(owl:ObjectProperty, owl:Datatype Property)에 해당하는 내용의 유형 구분, 계층구조(rdf:subPropertyOf) 및 범위(rdfs:Domain, rdfs:Range)의 정의와 제한: STNet의 X 용어가 속한 클래스는 rdfs:domain으로, Y 용어가 속한 개념 클래스는 rdfs:range로 대응하여 범위를 정의하고 제한할 수 있도록 하였다.

3.1.2 RDF 데이터 추출

STNet의 RDB 테이블은 저장하고 있는 데이터의 유형에 따라 DT 계열, RT 계열, IT 계열의 테이블 그룹으로 구성되어 있다. DT 계열은

주요 데이터가 저장되어 있는 테이블로써 용어, 용어정의, 외래어, 클래스 및 속성정의 등의 테이블로 구성되어 있으며, RT 계열 테이블에는 용어-클래스 간 연결정보, 용어-속성 간 연결정보, 용어-용어 간 연결정보가 있고, IT 계열 테이블은 용어-클래스-속성 정보, 용어-클래스-값 정보의 테이블로 구성되어 있다. STNet을 구성하고 있는 위와 같은 테이블 구조로부터 주어부는 DT 계열 테이블로, 술어부는 RT 계열 테이블로, 목적어부는 IT 계열 테이블로 대응시켜 변환 대상 데이터를 추출하였다(〈그림 1〉 참조).

STNet RDB 데이터 중 “실존인물” 클래스에 속한 “혜경궁 홍씨”를 사례로 하여 추출한 SQL 쿼리와 결과를 정리해 보면 〈그림 2〉와 같다.



〈그림 1〉 주요 테이블 그룹 및 ERD

```

SELECT al.* FROM (
  SELECT t.term_name as subject, p.predicate_name as predicate, tt.term_name as objects
  FROM it_term it, dt_term t, dt_class c, dt_predicate p, dt_term tt
  WHERE
    it.term_id = t.term_id AND
    it.predicate_id = p.predicate_id AND
    it.class_id = c.class_id AND
    it.it_value = tt.term_id union
  SELECT DISTINCT t.term_name as subject, p.predicate_name as predicate, it.it_value as objects
  FROM it_varchar it, dt_term t, dt_class c, dt_predicate p, dt_term tt
  WHERE
    it.term_id = t.term_id AND
    it.predicate_id = p.predicate_id AND
    it.class_id = c.class_id ) al
WHERE subject = '해경궁 흥씨':
    
```

⚡ SUBJECT	⚡ PREDICATE	⚡ OBJECTS
1 해경궁 흥씨	생물년	1735-1815
2 해경궁 흥씨	시대	조선 후기
3 해경궁 흥씨	저작	한중록
4 해경궁 흥씨	직업	왕후

〈그림 2〉 “해경궁 흥씨”에 대한 데이터 추출 쿼리 및 결과

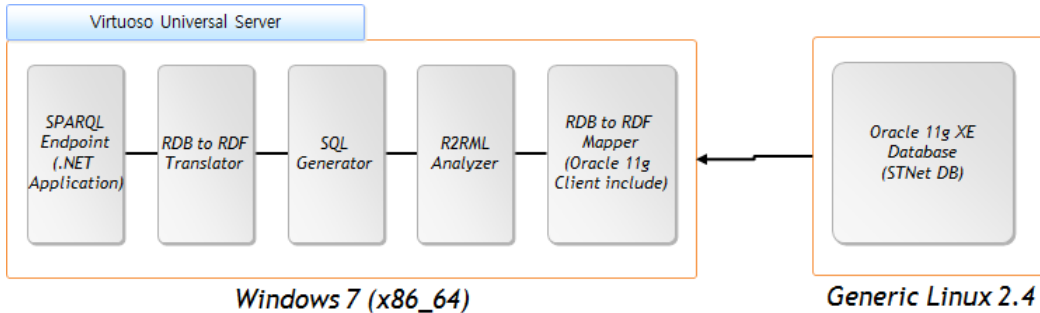
3.2 R2RML 방식의 RDF 변환

R2RML 방식의 매핑을 위해 STNet 시스템의 테이블에서 데이터를 추출하여 View 형태의 테이블 데이터를 만들어 RDF로 변환할 수 있게 하였으며, 최종적인 변환 작업은 RDB to RDF 매퍼, R2RML 분석기, SQL 생성기, RDB to RDF 변환기를 통해 수행하였다. RDB to RDF 매퍼는 RDB 구성요소와 RDF 구성요소를 매핑하며, R2RML 분석기는 RDB to RDF 매퍼에서 정의한 규칙을 바탕으로 R2RML 문서를 분석하고, SQL 생성기는 분석된 데이터를 기반으로 실제 질의문을 생성하고 해당 질의문을 통해 RDB의 데이터를 추출하며, RDB to RDF 변환기를 통해 추출된 데이터를 RDF로

변환하는 기능을 가진다(〈그림 3〉 참조).

R2RML을 통해 STNet의 데이터를 RDF로 변환하고 검증하는 과정은 네 단계로 나누어 진행되었다.

- 첫 단계로 R2RML 방식은 테이블 형태의 데이터를 RDF로 변환하는 구조를 취하고 있으므로, 전형적인 RDB의 테이블 구조로 만들기 위해 쿼리를 통해 STNet의 변환 대상 데이터로부터 View 테이블을 생성하였다. 한편, View 형태의 테이블에서 데이터를 추출할 때는 primary key가 정의되어 있어야 하므로 STNet RDB에서 제공하는 rownum을 이용해 임의의 primary key를 부여하였으며, 각 객체별 URI 생성을 위해 임의의 namespace를 생성해 적용하였다.



〈그림 3〉 R2RML 변환을 위한 시스템 구성

- 두 번째로 R2RML 매핑언어를 기계적으로 생성하기 위하여 Virtuoso를 사용하여 R2RML 변환 스크립트를 생성하였다(〈그림 4〉 참조). Virtuoso는 OPENLINK사의 트리플 저장소로써 SPARQL Endpoint, HTTP를 통한 데이터 제공, Linked Data 발행 등의 기능을 제공한다(<http://virtuoso.openlinksw.com/> 참조).
- 세 번째로 역시 Virtuoso를 사용하여 두 번째 단계에서 생성된 R2RML 변환 스크립트의 데이터를 RDF로 변환하였다.
- 마지막으로 변환된 RDF 데이터를 대상으로 SPARQL 질의를 실행하여 응답 시간

을 측정하였다. 이때 Virtuoso에서 제공하는 SPARQL Endpoint를 사용하였으며, SPARQL 질의어는 전체 데이터의 주어부, 술어부, 목적어부를 조회하도록 하였다(〈그림 5〉 참조).

3.3 D2R 서버에 의한 Non-R2RML 방식의 RDF 변환

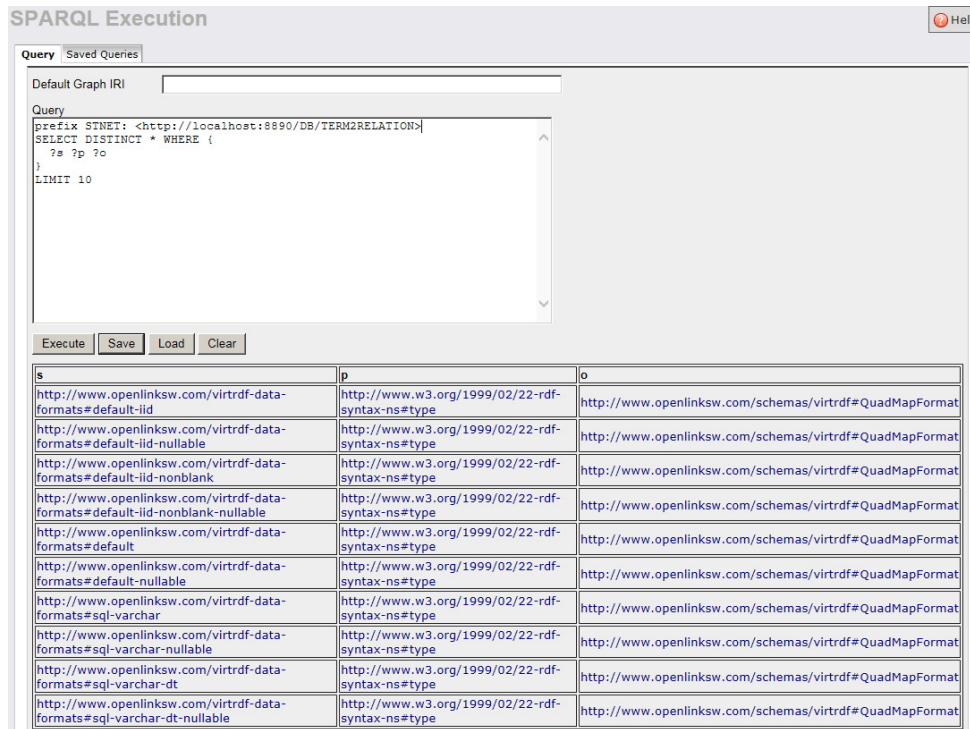
STNet의 RDB에 저장된 데이터를 Non-R2RML 방식으로 매핑하기 위해 D2R서버(<http://d2rq.org/>)를 사용하였다. D2R 서버 시스템은 베를린 자유대학에서 개발한 미들웨어 시스

```

R2RML Graph
@prefix rr: <http://www.w3.org/ns/r2rml#> .
@prefix DB: <http://localhost:8890/schemas/DB/> .
@prefix db-stat: <http://localhost:8890/DB/stat#> .
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix void: <http://rdfs.org/ns/void#> .
@prefix scovo: <http://purl.org/NET/scovo#> .
@prefix owl: <http://www.w3.org/2002/07/owl#> .

<#TriplesMapTERM2RELATION> a rr:TriplesMap; rr:logicalTable [ rr:tableSchema "DB" ; rr:tableOwner "STNet" ;
rr:tableName "TERM2RELATION" ];
rr:subjectMap [ rr:termType rr:IRI ; rr:template "http://localhost:8890/DB/term2relation/PRIMARY_KEY={PRIMARY_KEY}";
rr:class DB:TERM2RELATION; rr:graph <http://localhost:8890/DB#> ];
rr:predicateObjectMap [ rr:predicateMap [ rr:constant DB:primary_key ]; rr:objectMap [ rr:column "PRIMARY_KEY" ]; ];
rr:predicateObjectMap [ rr:predicateMap [ rr:constant DB:subject_uri ]; rr:objectMap [ rr:column "SUBJECT_URI" ]; ];
rr:predicateObjectMap [ rr:predicateMap [ rr:constant DB:subject ]; rr:objectMap [ rr:column "SUBJECT" ]; ];
rr:predicateObjectMap [ rr:predicateMap [ rr:constant DB:predicate ]; rr:objectMap [ rr:column "PREDICATE" ]; ];
rr:predicateObjectMap [ rr:predicateMap [ rr:constant DB:objects ]; rr:objectMap [ rr:column "OBJECTS" ]; ];
rr:predicateObjectMap [ rr:predicateMap [ rr:constant DB:object_uri ]; rr:objectMap [ rr:column "OBJECT_URI" ]; ].
    
```

〈그림 4〉 스크립트를 통해 생성된 R2RML(예시)



〈그림 5〉 SPARQL을 통한 R2RML 데이터 질의 결과(예시)

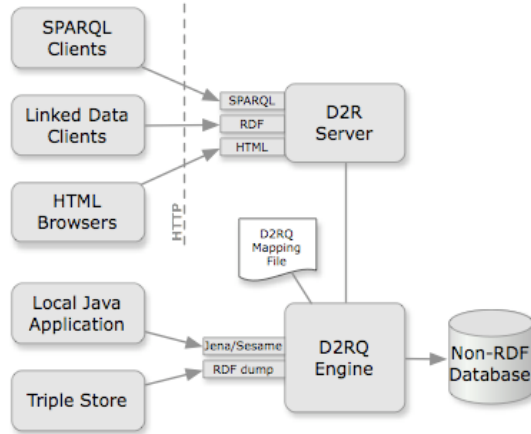
템으로 별도의 자체 엔진을 통해 RDB 내의 쿼리들을 RDF 저장소 없이 RDF로 변환하여 질의 접근점을 제공한다.

D2R 서버는 자체의 D2RQ 매핑 언어를 사용하여 응용 프로그램의 특정 데이터베이스 스키마와 RDFS, 또는 OWL 온톨로지 간에 매핑을 할 수 있도록 지원한다. D2RQ의 매핑 속성 값은 데이터베이스 테이블 단위, 혹은 view 단위로 RDF를 생성하고 각 변환 단위에 대한 생성 방법을 지정한다. D2RQ의 기본 객체인 ClassMap은 변환하고자 하는 RDB 테이블의 집합으로부터 매핑을 한다. 각 ClassMap 객체의 구성과 속성 값은 데이터베이스의 테이블 데이터나 SQL을 통한 가상테이블을 이용하여 생성하며, 개별 테이블의 속성이나 테이블간의 연결관계에 대한

설명을 포함한다. D2R 서버는 데이터베이스의 테이블 구조에서 D2RQ 매핑을 생성하는 도구를 포함하고 있다. 이 도구를 통해 속성 이름과 같은 클래스 이름과 열 이름으로 테이블 이름을 사용하여 변환 대상 데이터베이스에 대한 새 RDF 어휘를 생성한다. 이러한 D2R 서버의 모듈과 구조를 모형화하면 〈그림 6〉과 같다.

D2R 서버를 통해 STNet의 데이터를 RDF로 변환하여 측정하는 과정은 세 단계로 나누어 진행되었다.

- 첫 단계로 STNet의 RDB에 접속하여 변환하고자 하는 데이터를 정의하고 이에 대한 속성값을 설정한 후, D2RQ 형태로 변환된 스크립트를 통해 STNet의 RDB 데이터를 RDF 데이터로 변환하였다(〈그림 7〉)



〈그림 6〉 D2R 서버의 시스템 모형(<http://d2rq.org/d2r-server>)

```
# Table ONOMA_DT_CLASS2PREDICATE
map:ONOMA_DT_CLASS2PREDICATE a d2rq:ClassMap;
  d2rq:dataStorage map:database;
  d2rq:uriPattern "ONOMA/DT_CLASS2PREDICATE/@@ONOMA_DT_CLASS2PREDICATE.CLASS2PREDICATE_ID@@";
  d2rq:class vocab:ONOMA_DT_CLASS2PREDICATE;
  d2rq:classDefinitionLabel "ONOMA_DT_CLASS2PREDICATE";
  .
map:ONOMA_DT_CLASS2PREDICATE_label a d2rq:PropertyBridge;
  d2rq:belongsToClassMap map:ONOMA_DT_CLASS2PREDICATE;
  d2rq:property rdfs:label;
  d2rq:pattern "DT_CLASS2PREDICATE #@ONOMA_DT_CLASS2PREDICATE.CLASS2PREDICATE_ID@";
  .
map:ONOMA_DT_CLASS2PREDICATE_CLASS2PREDICATE_ID a d2rq:PropertyBridge;
  d2rq:belongsToClassMap map:ONOMA_DT_CLASS2PREDICATE;
  d2rq:property vocab:ONOMA_DT_CLASS2PREDICATE_CLASS2PREDICATE_ID;
  d2rq:propertyDefinitionLabel "DT_CLASS2PREDICATE_CLASS2PREDICATE_ID";
  d2rq:column "ONOMA_DT_CLASS2PREDICATE.CLASS2PREDICATE_ID";
  d2rq:datatype xsd:decimal;
  .
map:ONOMA_DT_CLASS2PREDICATE_CLASS_ID a d2rq:PropertyBridge;
  d2rq:belongsToClassMap map:ONOMA_DT_CLASS2PREDICATE;
  d2rq:property vocab:ONOMA_DT_CLASS2PREDICATE_CLASS_ID;
  d2rq:propertyDefinitionLabel "DT_CLASS2PREDICATE_CLASS_ID";
  d2rq:column "ONOMA_DT_CLASS2PREDICATE.CLASS_ID";
  d2rq:datatype xsd:decimal;
  .
map:ONOMA_DT_CLASS2PREDICATE_PREDICATE_ID a d2rq:PropertyBridge;
  d2rq:belongsToClassMap map:ONOMA_DT_CLASS2PREDICATE;
  d2rq:property vocab:ONOMA_DT_CLASS2PREDICATE_PREDICATE_ID;
  d2rq:propertyDefinitionLabel "DT_CLASS2PREDICATE_PREDICATE_ID";
  d2rq:column "ONOMA_DT_CLASS2PREDICATE.PREDICATE_ID";
  d2rq:datatype xsd:decimal;
  .
map:ONOMA_DT_CLASS2PREDICATE_PREDICATE_RANK a d2rq:PropertyBridge;
  d2rq:belongsToClassMap map:ONOMA_DT_CLASS2PREDICATE;
  d2rq:property vocab:ONOMA_DT_CLASS2PREDICATE_PREDICATE_RANK;
  d2rq:propertyDefinitionLabel "DT_CLASS2PREDICATE_PREDICATE_RANK";
  d2rq:column "ONOMA_DT_CLASS2PREDICATE.PREDICATE_RANK";
  .
```

〈그림 7〉 STNet 데이터의 D2RQ 매핑언어 적용(예시)

참조). 이때 빠른 데이터의 변환을 위해 D2R 서버에서 기본형으로 제공하는 RDF의 표현인 n-triple 구조를 사용하였다.

- 두 번째로 D2RQ 매핑언어 생성 후 해당 내용을 검증하기 위해 D2R 서버에서 제공하는 d2r-query를 검증도구로 하여 오류를 확인하고 수정하였으며 (〈그림 8〉 참조), 검

증이 완료된 데이터를 대상으로 RDF로의 변환을 실행하여 n-triple 파일을 생성하였다.

- 마지막으로 생성된 RDF 파일을 D2R 서버에 탑재한 후 http를 통해 SPARQL 질의를 실행하여 변환된 결과의 응답 속도를 측정하였다(〈그림 9〉 참조).

```

C:\hd2rq-0.8.1>d2r-query stnet.ttl "SELECT * < ?s ?p ?o > LIMIT 10"
-----
| s | p | o |
-----
| <ONOMA/TERM2TERM/1031> | vocab:ONOMA_TERM2TERM_HUMANITIES | "0"^^xsd:decimal |
| <ONOMA/TERM2TERM/1031> | vocab:ONOMA_TERM2TERM_DR_CHECK | "1"^^xsd:decimal |
| <ONOMA/TERM2TERM/1031> | vocab:ONOMA_TERM2TERM_B13 | "53"^^xsd:decimal |
| <ONOMA/TERM2TERM/1031> | vocab:ONOMA_TERM2TERM_B00 | "5"^^xsd:decimal |
| <ONOMA/TERM2TERM/1031> | vocab:ONOMA_TERM2TERM_ABG | "B" |
| <ONOMA/TERM2TERM/1031> | vocab:ONOMA_TERM2TERM_SOCIETY | "62"^^xsd:decimal |
| <ONOMA/TERM2TERM/1031> | vocab:ONOMA_TERM2TERM_SOCIETYAVERAGE | "1" |
| <ONOMA/TERM2TERM/1031> | rdfs:label | "TERM2TERM #1031" |
| <ONOMA/TERM2TERM/1031> | vocab:ONOMA_TERM2TERM_ONEST_CHECK | "1"^^xsd:decimal |
| <ONOMA/TERM2TERM/1031> | vocab:ONOMA_TERM2TERM_VECH | "0"^^xsd:decimal |
    
```

<그림 8> D2RQ를 통한 변환 검증

Snorql: Exploring http://localhost:18080/sparql

SPARQL:

```

PREFIX rdt: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
PREFIX vocab: <http://local.host:18080/resource/vocab/>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX rdfa: <http://local.host:18080/resource/#>
PREFIX foaf: <http://local.host:18080/resource/>

SELECT DISTINCT * WHERE {
  ?s ?p ?o
}
LIMIT 10
        
```

Results:

SPARQL results:

s	p	o
db:ONOMA/IT_TERM/188925	vocab:ONOMA_IT_TERM_PREDICATE_ID	163
db:ONOMA/IT_TERM/188925	vocab:ONOMA_IT_TERM_CLASS_ID	201
db:ONOMA/IT_TERM/188925	vocab:ONOMA_IT_TERM_IT_INDEX	1
db:ONOMA/IT_TERM/188925	rdfs:label	"IT_TERM #188925"
db:ONOMA/IT_TERM/188925	vocab:ONOMA_IT_TERM_IT_VALUE	6200
db:ONOMA/IT_TERM/188925	vocab:ONOMA_IT_TERM_TERM_ID	33262
db:ONOMA/IT_TERM/188925	vocab:ONOMA_IT_TERM_IT_TERM_ID	188925
db:ONOMA/IT_TERM/188925	rdf:type	vocab:ONOMA_IT_TERM
db:ONOMA/IT_TERM/188924	vocab:ONOMA_IT_TERM_PREDICATE_ID	160
db:ONOMA/IT_TERM/188924	vocab:ONOMA_IT_TERM_CLASS_ID	201

<그림 9> SPARQL을 통한 Non-R2RML 데이터 질의 결과(예시)

4. 변환 결과 측정

Virtuoso를 이용한 스크립트 생성 후 RDF로 변환하는 R2RML 매핑 방식과 D2RQ 매핑언어를 생성하여 변환하는 Non-R2RML 매핑 방식을 통해 변환된 각각의 결과를 평가하기 위해, 변환 데이터의 규모와 변환 시간, 질의에 대한 응답속도를 측정하였다. 평가를 위해 사용한 테스트 데이터는 의미목록이 구축된 49,675개 용

어이다.

변환 규모의 평가는 RDF에 포함된 표현의 풍부성과 추론 가능성 정도를 판단할 수 있도록 생성된 트리플 데이터의 수를 측정 기준으로 삼았다. 변환 시간과 질의 응답 시간의 평가는 STNet과 같은 실제 구현된 동적 RDB를 대상으로 Non-R2RML 방식과 R2RML 방식을 적용했을 때 시스템 성능 면에서 어떤 차이를 보이는지 확인하기 위한 것으로서 테스트 데이

터의 튜플당 RDF 트리플 변환 시간과 동시에 접속자의 쿼리에 대한 응답 속도를 측정하였다. 변환 시간과 응답 속도에 대한 측정 장비는 <표 7>과 같다.

4.1 변환 데이터의 규모

R2RML 방식의 경우 RDB의 스키마 정보만을 변환하는 반면, D2R 서버를 이용한 Non-R2RML 방식에서는 속성값에 object property로 정의되어 있는 용어도 변환한다. R2RML 변환 방식을 통해서서는 355,430개의 트리플 데이터가 생성되었으며 D2R 서버를 이용한 Non-R2RML 변환 방

식을 통해서서는 489,298개의 트리플 데이터가 생성되어 본 연구에서도 Non-R2RML 방식이 더 많은 온톨로지를 변환한 것으로 나타났다. 예로 <그림 10>의 트리플 변환 결과를 보면, R2RML 방식의 경우 용어 “민주주의”에 대하여 “권력분산(http://www.skku.edu/ONOMA/IT_TERM/49027)”이 “hasMethod” 관계를 통해 트리플로 연결되어 있지만 “권력분산”을 주어로 하는 역관계(isMethodOf)의 트리플은 존재하지 않는다. 그러나 Non-R2RML 방식에서는 동일한 사례에 대해 “권력분산”을 주어로 하고 “isMethodOf”로 “민주주의”와 연결된 트리플도 생성되었음을 보여주고 있다. 따라서 데이터의 변형이 빈번

<표 7> RDBMS와 변환 및 측정 장비 사양

구분	항목	사양
RDBMS	CPU	Intel(R) 2 core cpu
	Memory	4GB
	HDD	100GB SSD
	OS	64bit CentOS
	RDBMS	Oracle 11g XE
변환 및 측정 장비	CPU	Intel(R) Core(TM) i5-2520M 2.50Ghz
	Memory	4GB
	HDD	250GB SSD
	OS	64bit Windows

	s	p	o
R2RML (Virtuoso)	http://www.skku.edu/ONOMA/IT_TERM/104	http://www.skku.edu/vocab/ONOMA_IT_TERM_OBJECT_PROPERTY#hasMethod	http://www.skku.edu/ONOMA/IT_TERM/49027
	http://www.skku.edu/ONOMA/IT_TERM/104	http://www.skku.edu/vocab/ONOMA_IT_TERM_OBJECT_PROPERTY#hasProperty	http://www.skku.edu/ONOMA/IT_TERM/166627
	http://www.skku.edu/ONOMA/IT_TERM/104	http://www.skku.edu/vocab/ONOMA_IT_TERM_OBJECT_PROPERTY#isAffectedBy	http://www.skku.edu/ONOMA/IT_TERM/172985
	http://www.skku.edu/ONOMA/IT_TERM/104	http://www.skku.edu/vocab/ONOMA_IT_TERM_OBJECT_PROPERTY#isBranchOf	http://www.skku.edu/ONOMA/IT_TERM/10694
	http://www.skku.edu/ONOMA/IT_TERM/104	http://www.skku.edu/vocab/ONOMA_IT_TERM_OBJECT_PROPERTY#isIngredientOf	http://www.skku.edu/ONOMA/IT_TERM/40703
	http://www.skku.edu/ONOMA/IT_TERM/104	http://www.skku.edu/vocab/ONOMA_IT_TERM_OBJECT_PROPERTY#isIssueln	http://www.skku.edu/ONOMA/IT_TERM/90197
Non-R2RML (D2R 서버)	http://www.skku.edu/ONOMA/IT_TERM/104	http://www.w3.org/1999/02/22-rdf-syntax-ns#type	http://www.skku.edu/vocab/ONOMA_IT_TERM/
	http://www.skku.edu/ONOMA/IT_TERM/104	http://www.w3.org/2000/01/rdf-schema#label	*IT_TERM #민주주의
	http://www.skku.edu/ONOMA/IT_TERM/104	http://www.skku.edu/vocab/ONOMA_IT_TERM_CLASS_ID	"596" < http://www.w3.org/2001/XMLSchema#decimal >
	http://www.skku.edu/ONOMA/IT_TERM/104	http://www.skku.edu/vocab/ONOMA_IT_TERM_OBJECT_PROPERTY#hasMethod	< http://www.skku.edu/ONOMA/IT_TERM/49027 >
	http://www.skku.edu/ONOMA/IT_TERM/104	http://www.skku.edu/vocab/ONOMA_IT_TERM_OBJECT_PROPERTY#hasProperty	< http://www.skku.edu/ONOMA/IT_TERM/166627 >
	http://www.skku.edu/ONOMA/IT_TERM/104	http://www.skku.edu/vocab/ONOMA_IT_TERM_OBJECT_PROPERTY#isAffectedBy	< http://www.skku.edu/ONOMA/IT_TERM/172985 >
	http://www.skku.edu/ONOMA/IT_TERM/104	http://www.skku.edu/vocab/ONOMA_IT_TERM_OBJECT_PROPERTY#isBranchOf	< http://www.skku.edu/ONOMA/IT_TERM/10694 >
	http://www.skku.edu/ONOMA/IT_TERM/104	http://www.skku.edu/vocab/ONOMA_IT_TERM_OBJECT_PROPERTY#isIngredientOf	< http://www.skku.edu/ONOMA/IT_TERM/40703 >
	http://www.skku.edu/ONOMA/IT_TERM/104	http://www.skku.edu/vocab/ONOMA_IT_TERM_OBJECT_PROPERTY#isIssueln	< http://www.skku.edu/ONOMA/IT_TERM/90197 >
	...		
	http://www.skku.edu/ONOMA/IT_TERM/49027	http://www.w3.org/1999/02/22-rdf-syntax-ns#type	< http://www.skku.edu/vocab/ONOMA_IT_TERM/
	http://www.skku.edu/ONOMA/IT_TERM/49027	http://www.w3.org/2000/01/rdf-schema#label	*IT_TERM #권력 분산
http://www.skku.edu/ONOMA/IT_TERM/49027	http://www.skku.edu/vocab/ONOMA_IT_TERM_CLASS_ID	"286" < http://www.w3.org/2001/XMLSchema#decimal >	
http://www.skku.edu/ONOMA/IT_TERM/49027	http://www.skku.edu/vocab/ONOMA_IT_TERM_OBJECT_PROPERTY#hasMethod	< http://www.skku.edu/ONOMA/IT_TERM/104 >	
http://www.skku.edu/ONOMA/IT_TERM/49027	http://www.skku.edu/vocab/ONOMA_IT_TERM_OBJECT_PROPERTY#hasSynonym	< http://www.skku.edu/ONOMA/IT_TERM/40181 >	

<그림 10> Virtuoso와 D2R서버의 Triple 변환 결과(예시)

하고 새로운 데이터의 추가나 데이터들 간의 연결관계가 지속적으로 변화하는 STNet과 같은 RDB에는 Non-R2RML 방식이 표현의 풍부성과 추론 가능성 정도를 높이는 변환을 수행한다고 할 수 있다.

4.2 변환 시간

RDB의 스키마 정보만을 변환하는 R2RML 방식과 속성값에 object property로 정의되어 있는 용어도 변환하는 Non-R2RML 방식은 총 데이터 변환에 걸리는 시간에 차이가 날 수밖에 없다. 따라서 RDB 테이블의 각 행을 말하는 튜플당 변환시간을 측정하여 매핑 파일 작성 단계를 제외한 규모의 변환 시간을 비교하였다. 측정을 위해 Virtuoso를 이용해 변환 스크립트를 생성 후 RDF로 변환하는 R2RML 방식과 D2RQ 매핑 언어를 생성하여 변환하는 Non-R2RML 방식의 데이터 변환 시간을 총 5회에 걸쳐 반복 실험을 수행하였다. 측정 결과 Virtuoso를 사용하는 R2RML 방식이 평균 0.0010691275초, D2RQ를 사용한 Non-R2RML 방식이 평균 0.0010075659초가 소요되어 D2RQ를 사용한 Non-R2RML 방식이 빠른 것으로 나타났다(〈표 8〉 참조).

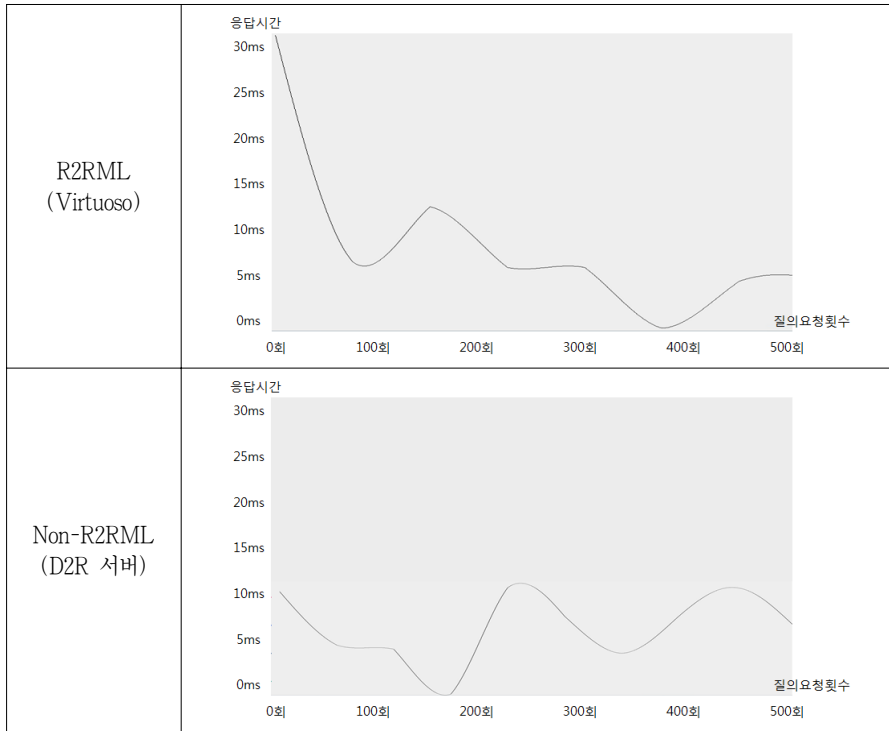
한편, 총 변환시간의 측정 결과는 Virtuoso를 사용한 R2RML 방식이 평균 6분 20초, D2RQ를 사용한 Non-R2RML 방식은 평균 8분 13초가 소요되어 Virtuoso를 사용하는 R2RML 방식이 빠른 것으로 나타났다.

4.3 질의 응답 시간

두 방식의 질의 응답 시간 비교를 위해 R2RML 기반의 Virtuoso와 Non-R2RML 기반의 D2R 서버에 각각 동시접속자 5명이 1초당 10회의 질의요청을 할 수 있도록 설정 한 후, 두 방식의 응답속도를 측정하였다. 이때 변환 방식에 따른 트리플 규모가 다르므로 동일한 질의 응답 조건에서의 속도를 측정하기 위해 전체 데이터 중 1,000건의 데이터를 출력하는 SPARQL 쿼리를 제출한 후 응답결과가 출력되는 시간까지를 Apache JMeter(<http://jmeter.apache.org>)를 통해 http 기반에서 측정하였다. 질의 응답 시간의 측정 결과, Virtuoso의 경우 RDF를 내부 메모리로 모두 로딩하는 초기 로딩시간이 오래 걸렸다. 초기 로딩시간을 제외한 질의응답 시간의 측정에서는 R2RML 방식의 최대치가 약 15ms, Non-R2RML 방식의 D2R 서버의 최

〈표 8〉 Virtuoso와 D2R서버의 RDF 변환시간(단위: 초)

회차	R2RML (Virtuoso)		Non-R2RML (D2RQ)	
	총 변환시간	튜플당 변환시간	총 변환시간	튜플당 변환시간
1	358	0.0010072307	507	0.0010361784
2	390	0.0010972625	502	0.0010259596
3	388	0.0010916355	478	0.0009769098
4	401	0.0011282109	495	0.0010116534
5	363	0.0010212981	483	0.0009871285
평균	380	0.0010691275	493	0.0010075659



<그림 11> Virtuoso와 D2R서버의 질의응답 시간(단위: 1/1000초(ms))

대치는 약 10ms로 나타났으나 최소치의 경우에는 차이가 거의 없었다. 두 방식 모두 질의요청 횟수 300회 후반부터는 안정적으로 기능하면서 유사한 유형의 속도 값이 측정되어 별 차이를 보이지 않는 것으로 나타났다(<그림 11> 참조).

5. 결론

본 연구에서는 RDB 기반의 구조적학술용어사전 “STNet”의 데이터 중 클래스가 설정되어 있는 용어와 관련 데이터를 대상으로 Virtuoso를 이용한 R2RML 방식과 D2R 서버를 이용한 Non-R2RML 방식의 RDF 온톨로지 변환 결

과를 측정하였다. 두 방식 모두 STNet의 데이터를 분석해 클래스와 관계유형을 정리하고 용어데이터와 클래스 간의 인스턴스를 추출하여 트리플 구조를 생성하는 과정으로 설계되었다.

두 가지 방식의 변환 작업에 대한 결과 평가는, 변환이 완료된 온톨로지 데이터의 규모와 변환 과정에 걸리는 시간, 그리고 질의에 대한 응답시간을 측정하는 방식으로 이루어졌다. 변환 규모의 평가는 RDF에 포함된 표현의 풍부성과 추론 가능성 정도를 판단할 수 있도록 생성된 트리플 데이터의 수를 측정 기준으로 삼았으며, 변환 시간과 질의 응답 시간은 시스템의 성능 면에서 어떤 차이를 보이는지 확인하기 위하여 매핑파일 단계를 제외한 규모의 데

이터를 대상으로 튜플당 RDF 트리플 변환 시간과 동시 접속자의 쿼리에 대한 응답 속도를 측정하여 비교하였다.

변환 규모의 평가 결과 Non-R2RML 방식이 더 많은 온톨로지를 변환하였으며, 특히 역관계의 RDF 온톨로지도 생성하는 것으로 나타났다. 튜플 당 변환 시간의 측정 결과 D2RQ를 사용한 Non-R2RML 방식이 Virtuoso를 사용하는 R2RML 방식에 비해 미세하나마 빠른 것으로 나타났다. 질의응답 시간은 두 방식 모두 질의요청 횟수 300회 후반부터는 안정적으로 기능하면서 유사한 유형의 속도 값이 측정되어 별 차이를 보이지 않는 것으로 나타났다. 따라서 데이터의 변형이 빈번하고 새로운 데이터의 추가나 데이터들 간의 연결관계가 지속적으로 변화하는 STNet과 같은 동적인 RDB를 시멘틱웹으로 발전시키고, 또 원래의 DB가 추구하는 표현의 풍부성을 구현하고 추론 가능성을 높이기 위해서는 Non-R2RML 방식이 적절한 것으로 평가할 수 있을 것이다.

본 연구는 RDB 기반의 구조적학술용어사전 "STNet"을 구축하고 이를 의미검색이 가능한 지식베이스로 발전시키는 과정에 있어서 '용어선

정에 관한 연구(고영만 외, 2013)', '텍사노미 구축에 관한 연구(고영만, 김비연, 민혜령, 2014)', '관계유형에 관한 연구(고영만, 송민선, 이승준, 2015)'에 이어지는 네 번째 단계의 연구이며, 연구 결과는 직접적으로 STNet을 시멘틱 웹으로 진전시키는 데 활용될 수 있을 것이다. 따라서 본 연구는 R2RML 방식과 Non-R2RML 방식의 기술적, 논리적 차이를 밝히기 위한 연구라기보다는, RDB 기반의 "STNet"을 사례로 Non-R2RML 방식을 적용하는 것이 풍부한 의미관계 표현 뿐 아니라 변환 속도 측면에서도 적절한 지를 확인하기 위한 연구에 해당한다고 할 수 있다.

본 연구는 시소러스와 같은 지식조직 데이터 베이스, 학술정보 메타데이터, 링크드데이터를 시멘틱 웹으로 진전시키는데 간접적인 기여를 할 수 있을 것이다. 그렇지만 본 연구가 RDB를 지식베이스로 발전시키는 연구 분야에 더 많은 기여를 하기 위해서는, 본 연구 결과를 반영하여 STNet의 데이터를 온톨로지로 변환한 후 추론시스템을 탑재해 의미 검색 결과를 측정하는 후속연구가 이루어지는 것이 반드시 필요하다.

참 고 문 헌

- 고영만 (2006). 시소러스 기반 온톨로지에 관한 연구. 정보관리, 5, 5-22.
- 고영만, 송민선, 김비연, 민혜령 (2013). 인문학 및 사회과학 분야 국내 학술논문의 저자키워드 출현빈도와 피인용횟수의 상관관계 연구. 정보관리학회지, 30(2), 227-243.
<http://dx.doi.org/10.3743/KOSIM.2013.30.2.227>
- 고영만, 김비연, 민혜령 (2014). 한국학술지인용색인(KCI)의 인문학, 사회과학, 예술체육 분야 저자키워드의 의미적, 형태적 분석에 의한 개념범주 텍사노미 연구. 한국문헌정보학회지, 48(4), 297-

322. <http://dx.doi.org/10.4275/KSLIS.2014.48.4.297>
- 고영만, 송민선, 이승준 (2015). 한국학술지인용색인(KCI)의 인문학, 사회과학, 예술체육 분야 저자키 워드의 의미적 관계 유형 최적화 연구. *한국문헌정보학회지*, 49(1), 45-67.
- 남영준 (2009). 재단구축 용어데이터레지스트리의 개념관계 유형 정형화를 통한추론 규칙 적용방안 연구. 서울: 한국학술진흥재단, 정책연구-2008-007-지식 확산.
- 박여삼, 장옥배, 한성국 (2008). X-TOP: 레거시 시스템상에서 온톨로지 구축을 위한 토폴맵 플랫폼의 설계와 구현. *정보과학회논문지: 컴퓨팅의 실제 및 레터*, 14(2), 130-142.
- 성하정, 김장원, 이석훈, 백두권 (2014). 관계형 데이터베이스 구성 요소의 의미 관계를 고려한 RDB to RDF 매핑 시스템. *정보처리학회논문지: 소프트웨어 및 데이터 공학*, 3(1), 19-30.
- 양준석, 김기성, 김형주 (2010). 관계형 데이터베이스 뷰 정의로부터 온톨로지 클래스와 계층 관계 생성 기법. *정보과학회논문지: 데이터베이스*, 37(6), 333-342.
- 이경호, 이준승 (2006). XML 문서의 변환을 위한 온톨로지 갱신 기반 XML 스키마 매칭. *정보과학회 논문지: 데이터베이스*, 33(7), 727-740.
- 최미영, 문창주 (2012). ER2iDM을 이용한 관계형 데이터베이스로부터 시멘틱 웹 구축 방법. *한국정보 기술학회논문지*, 10(10), 189-200.
- 최지웅, 김명호 (2014). 관계형 데이터베이스로부터 생성된 OWL 온톨로지를 위한 탐색기. *한국콘텐츠 학회논문지*, 14(10), 438-453. <http://dx.doi.org/10.5392/JKCA.2014.14.10.438>
- 한성국, 이현실 (2006). 시소리스를 활용한 온톨로지 구축방안 연구. *한국비블리아학회지*, 17(1), 286-303.
- Bizer, C. (2003, May). D2R MAP - A database to RDF mapping language. Poster Presentation In the Twelfth International World Wide Web Conference (WWW2003), Budapest, Hungary. poster presentation
- Bumans, G. (2010). Mapping between relational databases and OWL ontologies: An example. *scientific papers, University of Latvia, Computer Science and Information Technologies*, 756, 99-117.
- Hert, M., Rief, G., & Gall, H. C. (2011, September). A comparison of RDB-to-RDF mapping languages. In *I-SEMANTICS 2011, 7th Int. Conf. on Semantic Systems*, Graz, Austria, 25-32.
- Michel, F., Montagnat, J., & Faron-Zucker, C. (2014). A survey of RDB to RDF translation approaches and tools. [Research Report] I3S. <hal-00903568v2> Retrieved from <https://hal.archives-ouvertes.fr/hal-00903568v2/document>
- Saboo, S., Halb, W., Hellmann, S., Idehen, K., Thilbodeau, T., Auer, S., Sequeda, J., & Ezzat, A. (2009). A survey of current approaches for mapping of relational databades to RDF. Technical Report 2009.

Apache JMeter. Retrieved from <http://jmeter.apache.org/>

D2RQ. Retrieved from <http://d2rq.org/>

Linked open vocabularies. Retrieved from <http://lov.okfn.org>

OPENLINK software. Retrieved from <http://virtuoso.openlinksw.com/>

• 국문 참고문헌에 대한 영문 표기

(English translation of references written in Korean)

- Choi, Ji Woong, & Kim, Myung Ho (2014). Navigator for OWL ontologies generated from relational databases. *The Journal of the Korea Contents Association*, 14(10), 438-453.
- Choi, Mi-Young, & Moon, Chang-Joo (2012). Method for building the semantic web from relational database using ER2iDM. *Journal of Korean Institute of Information Technology (KIIT)*, 10(10), 189-200.
- Han, Sung-Kook, & Lee, Hyun-Sil (2006). A study of ontology construction using thesaurus: Transformation of thesaurus into SKOS. *Journal of the Korean Biblia Society for Library and Information Science*, 17(1), 286-303.
- Ko, Young Man (2006). A study on the ontology based on thesaurus. *Information Management*, 5, 5-22.
- Ko, Young Man, Song, Min-Sun, Kim, Bee-Yeon, & Min, Hye-Ryoung (2013). A study on the correlation between the appearance frequency of author keyword and the number of citation in the humanities and social science journal Articles of the Korea Citation Index (KCI). *Journal of the Korean Society for Information Management*, 30(2), 227-243.
- Ko, Young Man, Kim, Bee-Yeon, & Min, Hye-Ryoung (2014). A study on a conceptual taxonomy of author keywords of humanities, social Sciences, and art and sport in the Korea Citation Index (KCI) by analysis of its meaning and lexical morpheme. *Journal of the Korean Society for Library and Information Science*, 48(4), 297-322.
- Ko, Young Man, Song, Min-Sun, & Lee, Seung-Jun (2015). A study on the optimization of semantic relation of author keywords in humanities, social sciences, and art and sport of the Korea Citation Index(KCI). *Journal of the Korean Society for Library and Information Science*, 49(1), 45-67.
- Lee, Kyong-Ho, & Lee, Jun-Seung (2006). XML schema matching based on ontology update for the transformation of XML documents. *Journal of Korean Institute of Information Scientists and Engineers (KIISE): Databases*, 33(7), 727-740.

- Nam, Young Joon (2009). A study on the inference rule of application conceptual registry of the term data registry of KRF. Seoul: National Research Foundation of Korea, Policy Studies-2008-007.
- Park, Yeo-Sam, Chang, Ok-Bae, & Han, Sung-Kook (2008). X-TOP: Design and implementation of topicMaps platform for ontology construction on legacy systems. *Journal of Korean Institute of Information Scientists and Engineers(KIISE): Computing Practices and Letters*, 14(2), 130-142.
- Sung, Ha Jung, Gim, Jang Won, Lee, Suk Hoon, & Baik, Doo Kwon (2014). An RDB to RDF mapping system considering semantic relations of RDB components. *KIPS transactions on software and data engineering*, 3(1), 19-30.
- Yang, Junseok, Kim, Kisung, & Kim, Hyung-Joo (2010). Generating ontology classes and hierarchical relationships from relational database view definitions. *Journal of Korean Institute of Information Scientists and Engineers (KIISE): Databases*, 37(6), 333-342.

[부록] RDB-to-RDF 매핑 언어 비교(Hert, Rief, & Gall, 2011)

	F1	F2	F3	F4	F5	F6	F7	F8	F9	F10	F11	F12	F13	F14	F15
Direct Mapping	(✓)	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✓
eD2R	✓	✓	✓	✓	✓	✓	✓	✓	✓	✗	✓	(✓)	✗	✓	✗
R ₂ O	✓	✗	✓	✓	✓	✓	✓	✓	✓	✗	✓	(✓)	✗	(✓)	✗
Relational.OWL	(✓)	✗	✓	✗	✗	✗	✗	✗	✓	✗	✓	(✓)	✗	✗	✓
Virtuoso	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	(✓)	✗	✓	✗
D2RQ	✓	✓	✓	✓	✓	✓	✓	✓	✓	✗	✓	(✓)	✓	✓	✗
Triplify	✓	✓	✓	✓	✓	✓	✓	✓	✓	✗	✗	(✓)	✗	✓	✗
R2RML	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	(✓)	✓	✓	✗
R3M	(✓)	✓	✓	(✓)	✓	✓	✓	✓	✓	✗	✗	✓	✗	✓	✓

✓ = full support (✓) = partial support ✗ = no support

측정 항목	내용
F1 Logical Table to Class	View 테이블과 같은 논리 테이블의 지원
F2 M:N Relationships	다대다의 연결구조 지원
F3 Project Attributes	테이블의 속성 지원
F4 Select Conditions	특정 조건별 변환 지원
F5 User-defined Instance URIs	URI 사용자 정의 지원
F6 Literal to URI	문자열 처리를 위한 URI 지원
F7 Vocabulary Reuse	테이블에 기반한 어휘 재사용
F8 Transformation Functions	문자열 값의 변환 기능 지원(날짜, 온도 등)
F9 Datatypes	데이터 값의 명시적 선언을 통한 유형 지원
F10 Named Graphs	RDF 그래프 지원
F11 Blank Nodes	RDF 그래프 내의 빈 값(노드) 지원
F12 Integrity Constraints	무결성 제약조건 지원(PK, FK 등)
F13 Static Metadata	Static 메타데이터 선언 지원
F14 One Table to n Classes	RDB가 정규화 되지 않았거나 개념분화가 필요한 경우 하나의 테이블에서 복수의 클래스로 매핑 지원
F15 Write Support	RDF를 통한 RDB 데이터의 저장 지원