

온톨로지 언어의 비교 연구: W3C OWL과 ISO 토픽맵을 중심으로*

A Comparison of Ontology Languages: Focusing on W3C OWL and ISO Topic Maps

오 삼 균(Sam-Gyun Oh)**

초 록

이 연구의 목적은 국제표준 온톨로지 언어로 간주되고 있는 W3C의 웹 온톨로지 언어(OWL)와 ISO 토픽맵 (Topic Map)의 핵심개념을 상세히 기술하고, 각 언어의 의미표현력에 대해서 비교분석하는 것이다. 본 논문의 구성은, 첫째 온톨로지의 기반을 이루고 있는 URI와 네임스페이스에 대해서 기술하는 것이고, 둘째 토픽맵의 핵심 개념인 토픽(Topic), 연계(Association), 어커런스(Occurrence) 등에 대한 상세한 설명을 제공하는 것이고, 셋째 토픽맵이 검색결과를 의미 있는 그룹으로 묶어서 보여 주어야 한다는 목적을 효율적으로 성취하는 방법에 대해서 기술하는 것이고, 마지막으로 토픽맵과 OWL의 의미표현력의 차이점에 대해서 상세하게 비교분석한 내용으로 되어 있다.

ABSTRACT

The purpose of this study is to describe major concepts related to W3C OWL and ISO Topic Maps and to provide the result of comparison and analysis regarding semantic expression power between two ontology languages. This paper is comprised of the following parts: 1) describing URI and namespace concepts that are fundamental building block of effective ontology construction; 2) offering detailed explanation of major Topic Map concepts such as topics, associations, and occurrences; 3) providing how to accomplish the second purpose of cataloging(grouping related items when displaying the search result) using Topic Map; and 4) finally explaining the difference between two ontology languages in terms of semantic expression power.

키워드: 온톨로지, 토픽맵
Ontology, Topic Map, W3C, OWL, ISO

* 본 연구는 성균학술비 지원으로 진행되었음.

** 성균관대학교 문헌정보학과 부교수(samoh@skku.edu)

논문접수일자 2004년 11월 30일 논문심사일자 2004년 12월 2일 게재확정일자 2004년 12월 20일

1. 연구의 필요성 및 목적

시맨틱웹 기술에 기반을 둔 온톨로지의 생성이 활발해지고 있는 상황이다. 국외에서는 W3C에서 제정한 OWL 기반 온톨로지와 ISO 토픽맵 기반 온톨로지가 서로 보완과 경쟁관계를 건전하게 유지하고 있는데, 유독 국내에서는 일방적으로 OWL 기반 온톨로지의 생성에 연구가 집중되는 경향이 있다. 이 두 온톨로지 언어는 서로의 장점과 단점이 있기 때문에 적절한 경쟁 관계를 유지하면서 상호의 발전을 촉진하는 것이 국내의 지식정보관리에 필요한 일로 간주된다. 이런 관점에서 이 두 언어의 핵심 개념과 기능에 대한 소개, 두 언어를 “의미 표현력(semantic expression power)”의 관점에서 비교분석함으로써, 국내의 건전한 시맨틱웹의 시대를 펼쳐가는 구도에 도움을 주는 것이 이 연구의 목적이다.

2. 온톨로지 기본 하부구조

시맨틱웹의 하부구조를 이루고 있는 기술은 식별체계, XML 이름공간(namespace), 그리고 XML 구문체계라 할 수 있다. 먼저 XML 구문체계가 데이터의 교환에 유익할 뿐 아니라 온톨로지의 표현 구문체계로도 가장 적절한 것으로 인식되고 있고, 널리 활용되고 있는 실정이다. 그러나 글로벌 식별체계는 제대로 정립되어 있지 않기 때문에 이에 대한 문제점을 파악하고자 한다.

2.1 “Identity” 관련 핵심사항

컴퓨터가 같은 사물과 개념에 대해서 언급하고 있다는 것을 어떻게 이해할 수 있는가? 시맨틱웹이 되기 위해서는 식별체계의 명확성이 가장 중요한 관건 중의 하나이다. 분명한 것은 이 식별체계를 자연어 기반 이름으로 할 수는 없다는 것이다. 자연어는 동의어 및 같은 의미의 다국어 존재하기 때문에 식별체계로서는 적절치 못하다. “식별자(Identifier)”를 사용하는 것이 해결책이다. 모든 구조화된 정보시스템에서는 다 식별자를 사용했으나, 문제는 서로 다른 식별체계를 사용했기 때문에 정보를 웹 환경에서 서로 연결하거나 통합하는데 도움이 되지 않고 있다. 전 세계가 식별체계를 공유하지 않고서는 시맨틱웹의 목표인 맞춤 정보서비스, 지능형 검색서비스, 끊임없는 정보의 연계 등이 불가능하다.

국제적으로 통용되는 대표적인 식별체계로는 URN과 PSI가 있고, URN은 IETF(Internet Engineering Task Force)에 의해서, PSI는 ISO Topic Maps 커뮤니티에서 시작한 것이다. 이 두 식별체계의 장점과 단점에 대해서 언급하고 국내의 시맨틱웹 기반구조를 다지기 위한 국내 식별체계의 전략을 제시하고자 한다. W3C RDF/OWL과 ISO Topic Maps 기반 온톨로지 구축은 불변의 식별자를 기본으로 하고 있다. 관리대상인 자원, 속성, 클래스에 불변의 식별자를 부과하지 않으면, 시맨틱웹은 거의 불가능하다(Pepper & Schwab 2003).

2.2 URN(Uniform Resource Name)

URN은 자원, 속성, 클래스에 자유롭게 부

과할 수 있는 범용적인 식별체계이나, URN을 살아 있는 URL로 바꿔 주는 변환 서버나 더블링크어에서 사용하고 있는 지속적인 URL (Persistent URL)과 같은 투자가 필요하기 때문에, 그 개념은 뛰어나지만 널리 활용되지 않고 있는 실정이다.

URN 개념을 성공적으로 적용한 사례는 미국출판협회와 국제출판협회의 주관으로 시작된 Digital Object Identifier(DOI) 식별체계이다. DOI는 Center for National Research Institute(CNRI)에서 개발한 핸들(handle) 변환 시스템을 활용하여 출판과 관련된 모든 자원에 DOI를 부착하여 지적재산권을 관리하기 위한 식별체계이다. 이 식별체계를 활용하여 유익한 서비스를 제공하는 있는 대표적인 기관은 CrossRef라 할 수 있으며(<http://www.crossref.org>), 현재 각 분야의 주요 논문을 취급하고 있는 출판사들이 자신들의 메타데이터에 DOI를 부착하여 CrossRef 데이터베이스로 방출함으로써, 각 논문들 간의 인용정보를 자동적으로 연결해주는 유익한 서비스를 제공하고 있다. 그러나 DOI를 부착하는 것이 무료가 아니고, 메타데이터를 등록하고 관리해야 한다는 점이 이 체제의 급속한 확산에 장애요인으로 존재하고 있다.

국내에서도 URN의 필요성을 인식하고 한국전산원이 주축이 되어 국가 Universal Content Identifier(이하, UCI) 식별체계를 확립하였다(허정희 2003). 국내에서도 URN 개념을 도입하면 유익한 상황과 관리해야 할 자원들이 많기 때문에 그 많은 것들에 유료의 DOI를 부착하는 것보다는 자체 변환 서버를 갖추는 것이 필요하다고 판단되어 현재 UCI 식별

시스템을 시범적으로 운영하고 있다. 이 UCI 시스템을 많은 기관에서 적극적으로 활용한다면 국내 시맨틱웹의 하부구조를 튼튼히 형성하는데 큰 도움이 될 것으로 보인다. 이 UCI 시스템을 확장하여 메타데이터 요소와 온톨로지의 등록서비스를 제공하는 방향으로 전진된다면 더 큰 이익을 줄 것이다. 또한 등록된 모든 요소들에 관한 정보가 RDF/OWL과 Topic Maps으로 표현되어 방출될 수 있는 인터페이스를 갖추어야 한다.

2.3 PSI

PSI는 공개적이고 분산적이면서 고유의 식별자를 전 세계가 공유할 수 있는 식별체계이다. 원래는 토픽맵의 필요에 의해서 제정되었지만, 그 용도는 토픽맵에 국한된 것이 아니고 시맨틱웹 환경에서 용이하게 적용될 수 있다. 그 핵심내용은 URN 대신에 URL을 식별자(identifier)로 사용하자는 것이다. 이 PSI는 '공용주제식별자'(Published Subject Identifier)의 의미와 '공용주제지시자'(Published Subject Indicator)라는 양면성을 지니고 있다. 공용주제식별자는 단순한 URL로서 컴퓨터를 위한 것이고, '공용주제지시자'는 이용자를 위한 것이다. 이용자가 '공용주제식별자'를 클릭하면 이 식별자의 정의와 용례를 기술한 웹 페이지가 꼭 보여야 하고, 이것을 '공용주제지시자'라 한다. 이러한 접근법의 시도는 URN 체계의 도입된 지 상당한 기간이 흘렀지만 그 보급이 확산되지 않는 이유가 상·하 전달식에 근거한 표준은 이용이 저조하다는데 근거를 둔 것이며, 식별체계의 신속한 확산은 시맨틱웹의 필수조건이기 때문이다. 다만 URL을 식별자

로 사용하는 단점은 '공용주제식별자'로 연결된 웹 페이지가 더 이상 존재하지 않을 가능성이 있다는 점이다. 각 기관에서 식별자로 사용하는 웹 페이지와 도메인 이름에 대한 유지를 보장하면 큰 문제없이 저렴하게 식별체계 문제를 해결할 수 있다.

참고로, URN과 URL을 포함하여 URI (Uniform Resource Identifier)라 한다. 그러므로 PSI를 포함하여, 고유의 식별체계를 가지고 있는 모든 것은 URI의 범주에 속한다고 볼 수 있다. 컴퓨터는 단순히 URL을 비교하면 되고, 이용자가 정확하게 각 PSI에 의해서 표현된 개념, 자원, 속성, 클래스 등이 무엇인지 식별할 수 있도록 기술되어야 한다. 현재 PSI에 구체적인 내용은 OASIS TC에서 다루고 있으며 PSI에 관한 필수사항은 다음과 같다(OASIS 2004).

1. 각 '공용주제식별자'는 단일의 URI가 할당되어야 한다.
2. 각 '공용주제식별자'를 클릭하면 이용자가 이해할 수 있도록 기술된 단일의 '공용주제지시자'와 연결되어야 한다.
3. 각 '공용주제지시자'는 정의와 용례에 대한 기술사항도 있겠지만, 무엇보다도 고유의 '공용주제식별자'로 사용될 URI가 반드시 할당되어야 한다.

모든 기관이 URN 체계를 다 수용할 정도의 하부구조를 가진 것이 아니기 때문에 국내에서는 한국전산원에서 개발한 UCI의 적극 활용과 보다 용이한 PSI의 활용을 적극 권한다. 이 두 식별체계의 목적이 같기 때문에 서로 보완관계에 있는 것이고, 중요한 것은 각 기관에

서 고유의 식별체계를 적극적으로 널리 활용하여 정보의 통합과 공유를 용이하게 수행할 수 있는 하부구조를 조속히 다지는 것이다.

2. 4 XML 이름공간(XML namespaces)

XML 이름공간은 의미 있는 URI 군집을 대표하는 명칭으로 간주하면 된다. 예를 들어, 각 메타데이터 요소의 URI는 이름공간에 요소명을 합해서 된 것이다. 이름공간과 URI 정책이 각 기관마다 확립되면, 의미충돌을 방지할 수 있으며 메타데이터의 요소가 기계가독형으로 표현되는 장점을 갖추게 되어 기관 간의 데이터의 통합과 정보의 공유가 용이하게 된다 (Bray 1999). 국내에서도 시맨틱웹의 기반을 견고히 다지기 위해서는 무엇보다도 각 기관에서 이름공간과 URI 정책을 조속히 수립해야 한다.

3. 모범적인 식별자 할당정책 사례

국제적으로 시맨틱웹 기술을 적용하여 이름공간과 URI 정책을 가장 모범적으로 적용하고 있는 사례를 더블링크어의 용어관리에서 볼 수 있고(DCMI 2004), 현재 더블링크어는 3개의 네임스페이스를 사용하고 있다(Powell 2001). 첫 번째 이름공간은 단순 더블링크어를 위한 것이고, 두 번째 이름공간은 새로 추가된 요소, 한정어 및 인코딩스킴을 관리하기 위한 것이고, 마지막 네임스페이스는 자료유형의 값을 위한 것이다.

- 단순 더블링크어 이름공간:

<http://purl.org/dc/elements/1.1/>

- 더블린코어 추가요소, 한정어, 인코딩스킴 이름공간: <http://purl.org/dc/terms/>
- 더블린코어 자료유형 이름공간: <http://purl.org/dc/dcmitype/>

단순 더블린코어 요소는 1996년에 ISO에 의해 국제표준으로 인정된 15개 요소로 국한되어 있다. 이에 해당하는 대표적인 URI-URI를 소개하면 다음과 같다.

- 제목 URI: <http://purl.org/dc/elements/1.1/title>
- 생성자 URI: <http://purl.org/dc/elements/1.1/creator>
- 주제 URI: <http://purl.org/dc/elements/1.1/subject>
- 요약정보 URI: <http://purl.org/dc/elements/1.1/description>

더블린코어 스키마에 'audience(자원대상)'와 'provenance(자원의 출처 및 변천과정)'라는 주요소가 최근에 첨가되었는데, 이 요소들이 한정어와 인코딩스킴의 이름공간으로 배정되었다. 여기에서 우리가 파악할 수 있는 사실은 더블린코어에서 앞으로 새로 제정하는 더블린코어 요소는 주요소나 한정어에 관계없이 <http://purl.org/dc/terms/> 이름공간에 할당하겠다는 정책을 세운 것이다. 다음은 더블린코어의 하위요소와 추가된 요소들에 관한 대표적인 URI를 소개한다.

- 추가요소 - 자원대상 URI: <http://purl.org/dc/terms/audience>

- 추가요소 - 자원출처 및 변천 URI: <http://purl.org/dc/terms/provenance>
- 제목의 하위요소 - 대체제목 URI: <http://purl.org/dc/terms/alternative>
- 요약정보의 하위요소 - 초록정보 URI: <http://purl.org/dc/terms/abstract>
- 날짜의 하위요소 - 제작일 URI: <http://purl.org/dc/terms/created>
- 식별자의 하위요소 - 서지정보 URI: <http://purl.org/dc/terms/bibliographicCitation>
- 관계의 하위요소 - 전체/부분관계 URI: <http://purl.org/dc/terms/isPartOf>
- 범위의 하위요소 - 시간적 범위 URI: <http://purl.org/dc/terms/temporal>
- 자원대상의 하위요소 - 교육수준 URI: <http://purl.org/dc/terms/educationLevel>

인코딩스킴명의 이름공간은 한정어와 동일하다. 더블린코어에서는 모든 메타데이터 요소는 주요소, 하위요소, 인코딩스킴을 동일한 이름공간에서 관리하기로 결정한 것이다. 더블린코어는 전 세계에서 가장 권위 있는 메타데이터 전문가들이 모여서 합의한 결과로 이뤄진 것이 때문에, 국내 URI와 이름공간 정책에 참조할 수 있는 좋은 사례라 판단된다. 대표적인 인코딩스킴의 URI를 제시하면 다음과 같다.

- 주제 DDC 스킴 URI: <http://purl.org/dc/terms/DDC>
- 포맷 IMT 스킴 URI: <http://purl.org/dc/terms/IMT>
- 자료유형 DCMIType 스킴 URI: <http://purl.org/dc/terms/DCMIType>

- 국가명 ISO3166 스킴 URI:
http://purl.org/dc/terms/ISO3166
- 공간영역 DCMI Box 스킴 URI:
http://purl.org/dc/terms/Box
- 시간범위 DCMI Period 스킴 URI:
http://purl.org/dc/terms/Period
- 날짜 W3CDTF 스킴 URI:
http://purl.org/dc/terms/W3CDTF
- 언어 ISO639-2 스킴 URI:
http://purl.org/dc/terms/ISO639-2

또한 URI는 메타데이터 요소를 구분하기 위해서만 사용하는 것이 아니라, 온톨로지 구축에도 사용되어야 한다. 온톨로지에 대한 다양한 정의가 있지만, 본 연구에서는 시소러스의 약점을 보완하기 위하여 개념들을 보다 명확히 표현하기 위해서 개발된 개념의 체계를 가리키는 것으로 정의한다. 시맨틱웹 환경에서는 W3C에서 제정한 OWL이나 ISO에서 제정한 토픽맵을 활용하여 온톨로지를 생성하게 된다. URI 정책의 관점에서 보면, 각 도메인에 관한 통제어 리스트나 온톨로지에 하나의 이름 공간을 할당하는 것이 바람직하다고 본다. 앞으로 많은 온톨로지가 구축될 것으로 기대되고 있기 때문에, 각 온톨로지는 고유의 이름공간이 할당되어야 서로 온톨로지 간의 의미 충돌이 없이 정보의 공유가 이뤄질 것이고, 또한 온톨로지 관리에도 용이하다. 각 기관에서만 중복이 없이 이름공간과 URI를 사용하면 각 기관의 도메인 이름이 다르기 때문에 식별자의 중복가능성은 없다. 온톨로지 생성과정에서 이미 할당된 모든 URI는 웹상에서 그 정의를 참조할 수 있어야 하고, 제대로 정의된 개념들은

다시 정의하지 말고 재활용해야 한다. 국내에서도 온톨로지를 구축할 때 고유의 이름공간과 URI를 적용하여 어느 기관에서 혹은 어느 분야에서 개발한 온톨로지인지와 책임 여부를 투명하게 관리하여 성공적인 구축사례가 점점 확산되도록 기반을 마련해야 한다.

지금까지 언급한 대로 각 기관에서 모든 메타데이터 요소에 고유의 URI와 XML 네임스페이스로 관리를 할 경우에 시맨틱웹을 구축하기 위한 하부구조를 이루는데 큰 도움이 되는 것은 사실이다. 그러나 각 기관에서 다양하게 설계되고 있는 모든 메타데이터 스키마를 보다 효율적으로 이용자들이 참조하기 위해서는 한국전산원에서 개발한 UCI 시스템이 적극적으로 활용되는 것이 바람직 하고, 용이하게 사용할 수 있는 PSI 정보를 한 곳으로 모아 주는 포털 사이트가 필요하다.

4. 토픽맵의 핵심 개념: Topic, Association, Occurrence

토픽맵은 책 뒷부분의 색인을 전자화하려는 노력에서 시작되었다. 제대로 구축된 색인은 독자가 원하는 정보에 직접 접근할 수 있도록 큰 도움을 제공한다. 색인의 구조를 살펴보면, 중요하다고 여겨지는 개념들을 토픽으로 열거하고 있으며, 토픽들 간의 관계가 표시되어 있고, 또한 각 토픽에 관한 정보를 담고 있는 쪽번호를 제공하므로 토픽을 쪽번호와 연결시키는 역할을 하고 있다. 토픽맵의 가장 기본적인 개념은 토픽(topic), 어커런스(occurrence), 연계(association) 이다. 그 외에 식별자(iden-

tity), 범위(scope) 등의 개념 또한 토픽맵을 이해하는데 중요하다. 여기서는 토픽맵에 관련한 주요 개념들에 대한 설명을 기술하고자 한다.

4. 1 토픽

토픽맵에서 토픽은 가장 일반적이면서도 포괄적으로 정의되어 있다. 즉, 토픽맵에서 지원하는 토픽은 사람, 개체(entity), 개념 등이며, 이들의 존재 유무에 상관이 없다. 또한, 이들 토픽의 특징 존재 유무에도 상관이 없으며 특징에 대한 어떠한 주장도 가능하다. 결국, 어떠한 것도 토픽으로의 전환이 가능하다는 것이다.

토픽맵 안의 토픽은 현실의 주제를 표현한다. 특정 주제를 표현하는 토픽이 컴퓨터 안에서 생성 된다는 것은 이 주제가 토픽을 통하여 기계가 이해하고 처리 할 수 있는 객체(object)로 전환되는 것을 의미한다. 주제가 토픽으로 전환된 후에는 이 토픽을 다른 토픽과 많은 연계 관계를 생성할 수 있다. 이러한 연계는 토픽으로 변환되기 전의 주제에는 불가능하다. 그 이유는 주제는 컴퓨터 시스템 밖에 존재하는, 즉 기계가 이해 할 수 있는 객체의 형태로 존재 하지 않기 때문이다. 주제와 토픽간의 차이점을 이해하는 것은 매우 중요하다. 주제는 '어떠한 것(thing)이라 할 수 있고, 토픽은 이러한 '어떠한 것'이 기계가 이해하고 처리할 수 있는 형태로 표현된 것이다. 기술적 용어를 쓰자면, 토픽은 주제를 구체화(reification)한 것이다. 즉, 추상적 개념이 구체화된, 컴퓨터가 처리 할 수 있는 형태로 전환된 것이다. 예를 들면, '이순신'이란 사람이 대한 모든 것을 '주제'이고, 그것을 컴퓨터에 '이순신'이라는 문자

열로 표현하여 저장하면 그것은 '토픽'화 된 것이다. 그리고 컴퓨터는 그 문자열이 '이순신'에 대해서 모든 것을 대표하는 것으로 이해한다. 토픽은 다음과 같은 3가지의 특성을 가진다.

- 토픽 이름 또는 기본 이름(base name): 인간 가독을 위한 주제의 이름. 토픽은 이름이 없을 수도 있고, 하나 이상의 이름을 가질 수도 있다.
- 토픽 연계 안에서의 역할: 다른 토픽과 연결되었을 때 토픽이 하는 역할을 규정한다.
- 토픽 어커런스: 토픽과 관련된 정보 자원에 대한 링크를 말한다. 토픽은 하나 이상의 정보 자원과 연결될 수 있고, 이 경우 이러한 정보 자원을 토픽의 어커런스라 부른다. 어떤 정보 자원이 특정 토픽에 관한 정보를 제공할 때, 이 정보 자원을 토픽의 어커런스로 만들 수 있다.

토픽은 하나 이상의 토픽 타입 즉, 토픽 클래스의 인스턴스(instance)일 수 있으며, 어떠한 토픽 타입에 속하지 않을 수도 있다. 예를 들어, "대구시"라는 토픽은 "시"라는 토픽 타입의 인스턴스이며, "대한민국"이라는 토픽은 "나라"라는 토픽 타입의 인스턴스이다. 토픽과 토픽 타입 간의 관계는 객체지향형 설계(object-oriented design)의 클래스 인스턴스 관계와 비슷하다. 토픽 타입도 토픽맵 안에서는 토픽으로 정의 된다. 즉, "나라", "시"를 토픽 타입으로 사용하려면, 먼저 "나라"와 "시"를 토픽 타입으로 정의해야 한다.

모든 토픽에는 '식별자'가 부여된다. 어떤 것을 토픽으로 지정 하느냐는 토픽맵이 쓰이는 애플리케이션의 종류에 달려 있다. 법에 관련

된 문서를 출판하는 회사의 관점에서 보면, 조문, 판례, 법정 등이 중요한 토픽 타입이 될 수 있다. 소프트웨어 개발사의 관점에서 보면, 함수, 변수, 객체, 방법론 등이 토픽 타입으로 쓰일 수 있다. 아래는 토픽 타입과 토픽에 대한 정의를 XTM(XML 토픽맵 구문구조)으로 표현한 것이다(표 1).

다음은 위의 XML 구문을 도식화한 것이다(그림 1).

4. 1. 1 이름(기본 이름)

이름은 토픽의 3가지 특성중의 하나이다. 토픽

이름은 인간이 토픽을 이해하는데 도움을 주는데 목적이 있으며, 이름이 없더라도 각각의 토픽에는 식별자가 있으므로 토픽을 분별하는 데는 무리는 없다. 토픽은 여러 개의 이름을 가질 수 있어서, 이 이름들을 용도에 맞게 사용할 수 있다. 가령, 특정 토픽의 이름을 여러 언어로 표현하여 다른 언어로 그 토픽맵이 번역될 때 사용할 수 있다. 이외에도 스타일, 도메인, 지리, 역사학적 기간 등의 문맥(context)에 맞게 특정 토픽에 여러 이름을 지정하여 사용할 수 있다.

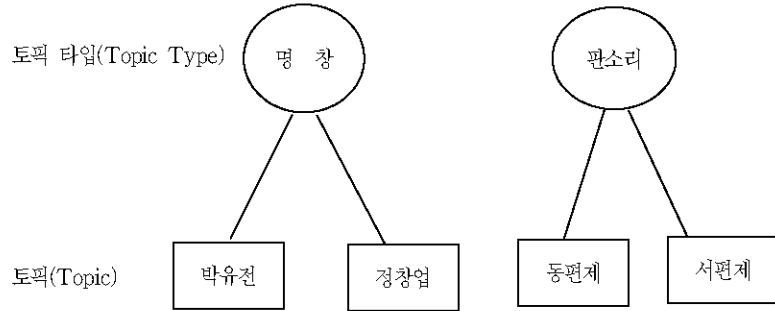
토픽의 이름은 여러 형태로 존재할 수 있다.

(표 1) 토픽과 토픽 타입의 XTM 구문 표현

```

<topicMap id="한국판소리">
<!--토픽타입 정의 ----->
  <topic id="명창">
    <baseName><baseNameString>판소리명창</baseNameString></baseName>
  </topic>
  <topic id="판소리">
    <baseName><baseNameString>한국판소리</baseNameString></baseName>
  </topic>
  <topic id="출생지">
    <baseName><baseNameString>출생지역명</baseNameString></baseName>
  </topic>
<!-- 토픽 정의 ----->
  <topic id="서편제">
    <instanceOf><TopicRef xlink:href="#판소리"/></instanceOf>
  </topic>
  <topic id="동편제">
    <instanceOf><TopicRef xlink:href="#판소리"/></instanceOf>
  </topic>
  <topic id="박유진">
    <instanceOf><TopicRef xlink:href="#명창"/></instanceOf>
  </topic>
  <topic id="정창업">
    <instanceOf><TopicRef xlink:href="#명창"/></instanceOf>
  </topic>
</topicMap>

```

(그림 1) 토픽과 토픽 타입의 관계

예를 들면, 공식적 이름, 상징적 이름, 별명, 애칭 등의 표현이 다 가능하다. 토픽맵 표준은 사용되고 있는 모든 이름의 종류들을 포함하는 것을 권장하지는 않는다. 다만, 가장 보편적으로 쓰이거나 중요하다고 고려되는 이름에 대해서는 표준화시키려 한다. 또한, 애플리케이션의 필요에 따라 다른 이름을 이용할 수 있도록 확장성을 지원한다는데 그 목적이 있다. 기본 이름의 예로는,

- “경북”, “경상북도”(동의어)
- “제주도”, “Jeju Island”(번역) 등 있다.

이외에도 지역적인 이름을 반영하여 방언 같은 것 또한 기본 이름으로 사용할 수 있다. 기본 이름의 XTM 구문은 표 2와 같다.

각각의 기본 이름은 변형이름(variant name)을 가질 수 있다. 변형이름은 표준에 의하여 출력용 이름(display name), 정렬용 이름(sort name)과 애플리케이션의 필요에 따라 사용할 수 있다. 변형이름의 예로는,

- “한국” 토픽의 표시이름: 대한민국
- 출력 불가능:
“http://abc.com/nonprintable.html” 등.
- 1024x768:
http://www.koreanpictures.org/images/logo.gif

토픽의 변형이름은 변수(parameter)에 의해 정의된다. 토픽 범위는 토픽 특성의 유효성을 정의하듯이 변수가 변형이름의 유효성을 정

(표 2)토픽 기본이름의 XTM 구문 표현

```

<topic id="경북">
  <baseName>
    <baseNameString>경북</baseNameString>
  </baseName>
  <baseName>
    <baseNameString>경상북도</baseNameString>
  </baseName>
</topic>
    
```

의한다. 토픽 변형이름은 하위 변형이름을 가질 수 있다. 토픽의 출력용 이름은 문자열이거나, 이미 존재하고 있는 자원을 참조하는 형식을 취할 수 있다. 참조 자원은 토픽의 이름을 보여주는 데이터, 즉 아이콘, 오디오 파일 등을 포함한다. 토픽의 정렬용 이름은 토픽을 정렬하는데 쓰이는 정렬 키(key)를 정의하는 문자열이고, 정렬 키는 특정 토픽이 다른 토픽과 함께 나열될 때의 위치를 계산하는데 쓰인다.

토픽의 출력용 이름과 정렬용 이름 자체를 표현하는 것은 이미 topicmaps.org에서 정의한 '공용주제'이다. 아래 표 3은 토픽 변형이름의 XTM 구문 표현의 한 예이다.

4. 2 어커런스

어커런스는 토픽을 정보 자원과 연결시키는 역할을 한다. 토픽맵이 토픽에 관한 정보를 제

(표 3)토픽의 변형이름 XTM 구문 표현

```

<topic id="한-국">
  <baseName>
    <baseNameString>한국</baseNameString>
    <variant>
      <parameters>
        <"subjectIndicatorRef xlink:href=http://www.topicmaps.org/xtm/1.0/core.xtm#sort"/>
      </parameters>
      <variantName>
        <resourceData>대한민국</resourceData>
      </variantName>
    </variant>
  </baseName>
</topic>
<topic id="대-한-민-국">
  <baseName>
    <baseNameString>대한민국</baseNameString>
    <variant>
      <parameters>
        <subjectIndicatorRef xlink:href="http://www.topicmaps.org/xtm/1.0/core.xtm#display"/>
      </parameters>
      <variant>
        <parameters>
          <topicRef xlink:href="#s1024x768"/>
        </parameters>
        <!-- 1024x768로 보여주기 -->
      </variantName>
      <resourceRef xlink:href="http://www.koreanpictures.org/images/logo.gif"/>
    </variantName>
  </variant>
</variant>
</baseName>
</topic>

```

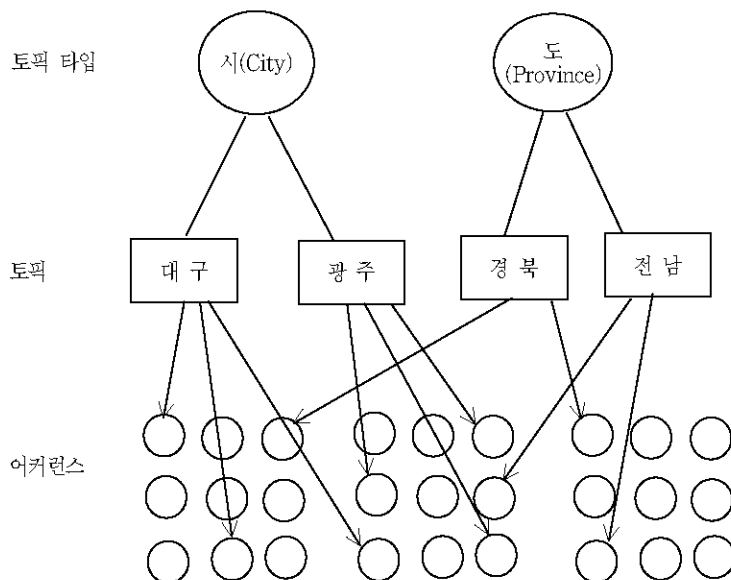
공할 때, 각 토픽에 관한 어커런스로 토픽에 관한 어떤 정보가 존재하는지를 확인할 수 있다. 어커런스는 토픽의 3가지 특성 중의 하나이다. 자원과 토픽 간에는 1:M(이하, 일대다)의 관계가 성립 될 수 있으나, N:M(이하, 다대다)의 관계는 성립 되지 않는다.

토픽 어커런스는 토픽과 자원을 연결하는 URI 값을 지니거나, 문자열 값을 취한다. URI 값을 취할 때는 XLink/XPointer URL을 이용하여 자원과 토픽을 연결한다. 문자열 값을 취할 때는, 'resourceData' 태그를 사용하여 실제 값을 입력하게 된다. 이를 통해 인터넷을 통해 참조할 수 있는 모든 자원을 토픽의 어커런스의 값으로 사용할 수 있다. 중요한 사항 중의 하나가 어커런스 또한 타입을 이용해 분류 할 수 있다는 것이다. 토픽 어커런스를 타입화 할 수 있다는 사실은 후에 언급된 목록의 제2목적의 실현과 긴밀한 관계를 지니고 있다.

어커런스 타입의 예로는 이미지, 삽화, 신문 기사, 비평, 홈페이지, 정의 등이 가능하다. 다음은 토픽 어커런스를 XTM으로 표현한 예이다 (표 4).

4.3 연 계

연계는 토픽맵 안에 정의된 토픽 간의 관계를 설정하는데 사용된다. 이를 통해 토픽맵 안에서 토픽이 갖는 문맥(context) 정보를 제공한다. 토픽간의 관계 설정은 지식을 모델링 하는데 필수적인 요소이다. 하나의 관계가 연결할 수 있는 토픽의 수는 무한정이다. 귀납적 관계(자신과 자신과의 관계), 두 토픽간의 관계(binary), 세 토픽간의 관계(ternary) 관계 등 애플리케이션의 필요에 따라 연결되는 토픽의 수가 다르게 설정 될 수 있다. 연계는 하나의 타입의 인스턴스로 정의될 수 있고, 그렇지



(그림 2)토픽 타입, 토픽, 어커런스의 관계

(표 4) 토픽 어커런스의 XTM 구문

```

<topic id="대구">
  <baseName>
    <baseNameString>대구</baseNameString>
  </baseName>
  <---- 어커런스 정의 ---->
  <occurrence>
    <---- 어커런스 타입 정의 ---->
    <instanceOf><topicRef xlink:href="#기사"/></instanceOf>
    <resourceRef
      xlink:href="http://www.infoKorea.co.kr/cities.html#daegu.html"/>
    </resourceRef>
  </occurrence>
  <occurrence>
    <instanceOf><topicRef xlink:href="#인구"/></instanceOf>
    <resourceData>2,500,000</resourceData>
  </occurrence>
</topic>

```

않을 수도 있다. 연계 타입 또한 토픽이다. 토픽과는 다르게, 연계는 하나 이상의 타입 인스턴스로 정의될 수 없다.

토픽맵 안에 정의된 연계는 방향성을 갖지 않는다. 연계는 토픽 사이에 존재할 수 있는 관계의 의미를 기술하는 것이다. 그러므로 이러한 정보나 설명은 방향성을 갖지 않을 수 있다. “나라”와 “도시” 토픽들 사이에 존재하는 “부분-전체” 관계나, “도시”와 “도시” 토픽들 사이에 존재할 수 있는 “근접도시” 관계에서 보는 바와 같이, 이 둘 관계들에는 방향성이 없다. 그러나 “부분-전체” 관계에서 나라 토픽이 도시 토픽을 포함하는 것인지, 도시 토픽이 나라 토픽을 포함하는 것인지는 분명치 않다. 그 나라와 도시를 아는 사람의 관점에서 보면 분명하지만, 이들의 이름을 모르는 사람이나, 컴퓨터의 관점에서는 이 “부분-전체” 관계는 분명치 않다. 토픽맵에서 연계를 설정할 때 역할을 활용하여 이 부분을 분명케 한다. 아래의 그림

에서 연계 역할(“여객선”, “공항”)이 “ABC”, “광주”, “제주” 토픽사이에서 관계(“항로”)를 연결하여 준다. 이 연계에 대한 설명은 다음과 같다.

- ABC는 광주에 있는 공항과 제주에 있는 공항을 연결하는 여객선을 운영한다.
- 광주에 있는 공항을 ABC 여객선이 제주에 있는 공항과 연결 한다.

XTM 구문을 이용해 위의 관계를 표현하면 다음과 같다.

4. 4 ‘주제식별자’와 ‘주제지시자’(Subject Identity and Subject Indicator)

토픽맵의 목적은 한 토픽을 통해 표현되는 주제에 관련된 모든 지식과 정보에 접근할 수 있도록 연결시키고, 또한 토픽들 간의 관계를 통해서 토픽의 의미를 분명하게 규명하는 것이

(표 5) 토픽 연계의 XTM 구문

```

<association>
  <instanceOf>
    <topicRef xlink:href="#항로"/>
  </instanceOf>
  <member>
    <roleSpec><topicRef xlink:href="여객선"/></roleSpec>
    <topicRef xlink:href="#ABC"/>
  </member>
  <member>
    <roleSpec><topicRef xlink:href="공항"/></roleSpec>
    <topicRef xlink:href="#광주"/>
  </member>
  <member>
    <roleSpec><topicRef xlink:href="공항"/></roleSpec>
    <topicRef xlink:href="#제주"/>
  </member>
</association>
    
```

다. 두 토픽맵이 합쳐질 경우 같은 주제를 하나 이상의 토픽으로 표현할 경우가 있다. 이러한 경우 그 두개의 토픽을 분별하는 방법이 요구 된다. 예를 들어, 노르웨이, 프랑스, 독일의 출판사가 토픽맵을 병합하려 할 때, "Italia", "l'Italie", "Italien"라는 토픽들이 개별적인 토픽맵에 존재 할 때는 중복되는 토픽이 없었으나, 병합 시에는 이들 토픽을 분별하는 방법이 요구 된다.

'주제식별자'는 이러한 주제의 식별을 위한 기능이다. 주제가 직접 접근이 가능한 자원인 경우에는 이 주제식별자의 값은 자원의 URL 주소가 될 수 있다. 그러나 대부분의 자원인 사람이나 개념은 직접적으로 컴퓨터에 표현할 수 없기 때문에, '주제지시자'를 이용하여 그 개념을 표현하는 방식을 취한다. 주제지시자는 "주제의 식별을 위한 설명을 제공 하는 웹 페이지"이다. 주제지시자 또한 자원이므로 보통 주제 식별자로 쓰일 수 있는 URI 주소를 갖는

다. 주제식별자는 컴퓨터 처리를 위한 것이고, 주제지시자는 그 주제에 대한 사람의 이해를 돕기 위한 것이다.

하나 이상의 '주제지시자' 또는 동일한 '주제식별자'를 공유하는 두 토픽은 이들 두 토픽의 특성인 이름, 어커런스, 또는 관계를 합한 하나의 토픽과 의미적으로 동일하다고 간주 된다. 실제로, 토픽맵 병합이 완성되면 이들 두 토픽의 특성을 병합한 하나의 토픽이 생성되는 것이다.

공적으로 유용한 문서가 '주제지시자'로 사용될 수 있다. 예를 들어, 국가 이름을 나타내는 ISO 표준 문서가 주제지시자로 쓰인다. 토픽맵 내외에 정의된 간단한 정의도 주제지시자로 쓰일 수 있다. '공용주제지시자'는 지식의 교환과 병합을 위해 공식적으로 알려진 곳에 보존하여 관리되는 주제지시자이다. 토픽맵이 널리 유용하게 쓰이기 위해서는 공적 주제지시자를 공유하는 것이 필요하다. 왜냐하면, 가령

한 토픽맵이 다른 조직에 전해졌을 때, 받는 쪽에서 사용된 어커런스(자원)와 주어진 토픽맵에서 사용하는 어커런스가 서로 상응한다는 보장이 없기 때문이다. 현재 OASIS의 주도하에 공적 주제의 사용과 문서화에 대한 권장사항이 개발되고 있다.

4.5 범위

토픽맵 모델은 특정 토픽에 관하여 세 가지를 기술할 수 있도록 허용한다. 즉 이름, 관계 및 어커런스의 토픽 특성 등이다. 토픽 특성의 지정은 항상 특정 문맥 내에서 의미를 갖게 된다. 이 문맥은 명백하거나 암시적인 것 모두를 포함한다. 토픽맵은 범위의 개념을 사용하여 문맥을 표현한다. 일반적으로, 토픽의 범위는 다음과 같은 경우에 사용된다.

- 언어: 토픽 이름과 어커런스는 이들이 사용되는 언어라는 범위에 의해 한정된다. 이들 언어가 영어, 독일어, 한국어 등의 언어일 수도 있고, 어떤 지방의 독특한 방언일 수도 있다.
- 접근 권리: 토픽 어커런스와 관계는 다른 자원이나 토픽에 접근할 수 있는 방법을 제공한다. 이러한 어커런스와 관계에 대한 범위는 정보 비밀성의 수준을 지정하거나, 접근이 허용된 사용자의 종류를 지정할 수 있다. 예를 들면, “confidential”, “public”, “registered user”, “unregistered user” 등에 관한 것이다.
- 뷰(views): 토픽 연계와 토픽 어커런스는 특정 토픽에 대한 문맥 정보(context information)를 제공한다. 토픽 범위는 이러한 정보가 어떠한 문맥에서 유용한지를 지정할

수 있다. 예를 들면, 이용자의 기술 수준(“expert”, “intermediate”, “beginner”), 취미 (“영화”, “연극”, “여행”) 등이다. 이용자 기술 수준을 표현하는 문맥의 경우, 이용자의 기술수준에 적합한 토픽만을 제공할 수 있다.

토픽맵의 범위는 대용량 토픽맵에서 필요한 정보만을 선택할 수 있게 한다. 기술적으로 말하자면, 범위는 토픽 특성의 유효성의 한계를 지정한다. 그리고 범위는 토픽 참조의 집합이다. 토픽맵 범위의 집합은 범위를 지정하는데 쓰인 토픽들의 집합이다. 토픽 집합안의 토픽 참조를 ‘범위를 제한하는 토픽’ 또는 테마라고 한다. 만약, 토픽에 어떠한 범위가 지정되어 있지 않으면, 이 토픽은 범위에 제약이 없다는 뜻이다. 범위 집합에 대한 해석과 사용은 개개의 응용 프로그램의 필요에 달려 있으며, 토픽맵 표준 자체가 미리 정의된 범위 집합을 제공하거나, 해석에 대한 지침은 주지는 않는다. 다음은 토픽맵 범위에 대한 XTM 구문이다(표 6).

4.6 클래스 체계(Class hierarchy)

토픽, 어커런스, 관계는 클래스의 인스턴스이며, 토픽맵에서 클래스 또한 토픽으로 정의된다. 클래스 토픽 또한 특정 클래스의 인스턴스로 정의될 수 있다. 결국, 토픽은 클래스이며 또한 인스턴스로 정의될 수 있다는 것과, 특정 토픽이 클래스로 정의되기 위해서는 최소한 하나의 클래스 인스턴스가 존재해야 함을 의미한다. 이 표현의 의미는, “나라”라는 토픽 클래스가 되려면, 최소한 하나의 나라가 이 토픽의 인스턴스로 정의되어야 한다는 것이다. 토픽

(표 6) 토픽맵 범위의 XTM 구문

```

<topic id="brit-virgin-islands">
  <baseName>
    <scope><topicRef xlink:href="#english"/></scope>
    <baseNameString>Brit. Virgin Islands</baseNameString>
  </baseName>
  <baseName>
    <scope><topicRef xlink:href="#german"/></scope>
    <baseNameString>Brit. Jungferninseln</baseNameString>
  </baseName>
  <occurrence>
    <instanceOf><topicRef xlink:href="#article"/></instanceOf>
    <scope><topicRef xlink:href="#public"/></scope>
    <resourceRef xlink:href="http://bviwelcome.com/intro.html"/>
  </occurrence>
</topic>
<association>
  <instanceOf><topicRef xlink:href="#geo-containment"/></instanceOf>
  <scope><topicRef xlink:href="#geography"/></scope>
  <member>
    <roleSpec><topicRef xlink:href="#container"/></roleSpec>
    <topicRef xlink:href="#caribbean"/>
  </member>
  <member>
    <roleSpec><topicRef xlink:href="#containee"/></roleSpec>
    <topicRef xlink:href="#brit-virgin-islands"/>
  </member>
</association>

```

맵 제작시스템이 토픽 저자에게 새로운 토픽이 생성될 때마다 현존하는 토픽 클래스의 리스트를 제공한다던 큰 유익을 줄 것이다. 그러나 이러한 시스템은 클래스로 사용되고 있는 토픽 클래스만을 보여줄 수 있다. 즉, 클래스로 사용될 수 있으나 현재 사용되고 있지 않는 토픽은 저자에게 제공되지 않는다는 뜻이다. 이러한 토픽맵 시스템에 관련한 표준으로 '토픽맵 제한언어'(Topic Map Constraint Language)가 사용될 것이다.

토픽 클래스들 간의 체계는 superclass-subclass 관계를 이용하여 정의되어 있다. 클래스

체계의 전형적인 응용분야로는 텍사노미(taxonomy)와 주제 분류 스키마가 있다. Superclass-subclass 관계는 'superclass-subclass' 클래스와 'superclass', 'subclass' 역할 간의 관계로 정의된다. Superclass-subclass 클래스와 'superclass', 'subclass' 역할은 표준화된 공적주제로 정의되어 있다. 토픽은 동시에 superclass, 그리고 subclass의 역할을 할 수 있다. 토픽 클래스, 관계 클래스, 어커런스 클래스 모두가 클래스 체계화할 수 있다. 아래는 클래스 체계를 XTM 구문을 이용해 표현한 예이다(표 7).

(표 7) 토픽맵 클래스 체계의 XTM 구문

```

<association>
  <instanceOf><subjectIndicatorRef xlink:href=
    "http://www.topicmaps.org/xtm/1.0/core.xtm#superclass-subclass"/>
  </instanceOf>
  <member>
    <roleSpec><subjectIndicatorRef xlink:href=
      "http://www.topicmaps.org/xtm/1.0/core.xtm#superclass"/>
    </roleSpec>
    <topicRef xlink:href="#나라"/>
  </member>
  <member>
    <roleSpec><subjectIndicatorRef xlink:href=
      "http://www.topicmaps.org/xtm/1.0/core.xtm#superclass"/>
    </roleSpec>
    <topicRef xlink:href="#대한민국"/>
  </member>
</association>

```

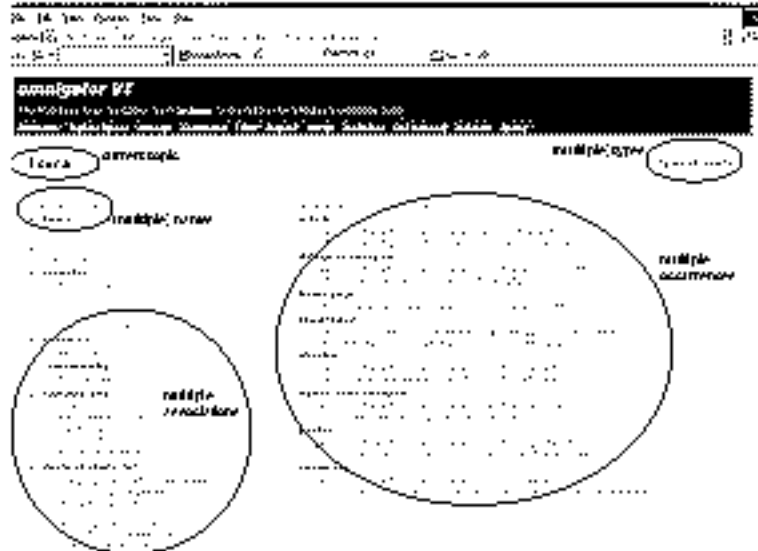
5. 토픽맵과 목록의 제 2 목적

목록의 제 1 목적은 정보를 찾아 주는 것이고, 제 2 목적은 검색결과를 단순하게 년도순, 제목순 등으로 정렬하는 것이 아니라, 의미 있는 그룹으로 재편성하여 보여 주어야 한다는 것이다. 현 목록 시스템이 제 1 목적에 대한 성과는 이론 편인데 제 2 목적에 대해서는 큰 진전을 보지 못했다. 세계도서관협회(IFLA)에서 제정한 FRBR(Functional Requirements for Bibliographic Relationships)이 이러한 문제점을 대처하기 위한 데이터 모델을 제시하였고, 최근 FRBR에 근거한 시스템 구축과 평가에 관한 논문이 발표된 것은 고무적인 일이고, 이 분야에 대한 지속적인 연구가 필요하다(조재인 2004).

주제를 중심으로 모든 자원을 융통성 있게 연계시킬 수 있는 프레임워크를 지닌 토픽맵은

목록의 제 2 목적을 이루기 위한 중요한 도구가 될 수 있다고 판단된다. 다음의 그림은 이에 대한 구체적인 예를 제시하고 있다.

그림 3에서 관련된 정보와 서로 거미줄처럼 연계된 상태를 보여 주고 있다. 상세하게 설명하자면, 현재 'Tosca'가 토픽(주제)과 연계된 정보를 보여 주는 것으로써 먼저 오른쪽 상단에 보면 Tosca는 Opera의 한 종류(Type)라고 정의되었고, 또한 Tosca에 다양한 이름이 표현될 수 있다는 것과 Tosca는 La Tosca에 근거한 것이라는 연계관계(based on), Tosca는 Puccini에 의해서 작곡되었다는 연계관계(composed by), Tosca 오페라 작품에서 등장하는 다양한 아리아(aria) 정보들과의 연계관계(contains aria), 그 작품에 등장하는 죽음과 관련된 캐릭터(character) 정보들과의 연계관계(death of character) 등이 동시에 표현 되어 있다. 이렇게 연계된 토픽을 클릭하



(그림 3) 토릭엠을 이용한 목록의 제2목적의 실현

면 그것이 새로운 토릭이 되고 이제 그 토릭과 연계된 정보들로 모든 정보가 즉시로 변환된다.

토릭엠의 강점이 여기에서 평주는 것이 아니라, 오른쪽 하단을 보면 현 토릭과 연결된 모든 자원을 보여 주는 데 논문, 오해라 비평, 홈페이지, 포스터 등으로 그룹 시켜서 검색결과를 보여 주고 있다. 이것을 토릭엠에서는 어커런스 타입이라 하는데 각 도메인에 적절한 어커런스 타입을 정의함으로써 이용자에게 유익하게 검색된 정보를 그룹해서 보여줄 수 있기 때문에 목록의 제 2 목적을 보다 충실하게 달성할 수 있게 된다.

어커런스 타입을 어떻게 세분하는 것이 가장 적절한지는 보다 체계적인 연구가 필요하다고 본다. 이미 고정된 어커런스 타입이 존재하는 것이 아니지만 분야별로 창조할 수 있는 Best Practice가 존재하고 있으며, 새로운 시

스템을 설계할 때 그 분야의 이용자에게 도움이 될 어커런스 타입을 정하여 검색결과를 그룹화 해서 보여 주면 이용자는 자기가 원하는 정보를 보다 용이하게 접근할 수 있다. 어커런스 타입도 토릭이기 때문에 모든 어커런스 타입은 PSI로 관리되어야 하고 다른 토릭엠 설계자들이 창조할 수 있도록 공유해야 한다.

토릭엠과 RDF/OWL의 표현력을 비교하기 위해서는 RDF/OWL에 대한 소개를 할 필요가 있으나, 이와 관련된 다수의 논문이 존재하기 때문에 RDF/OWL의 핵심개념은 다른 논문을 창조하기 바란다(Smith 2002; 오삼환 2002; Stevens 2003; Miller 1998). 이로써 RDF/OWL과 토릭엠에 관한 기본개념을 마친 것으로 하고 시맨틱 웹 기반 온톨로지 생성에 가장 유용하게 쓰이고 있는 이 두 기술의 의미표현력에 대한 비교와 분석 내용을 기술하고자 한다.

6. 토픽맵과 RDF/OWL의 의미표현력의 비교

토픽맵과 RDF/OWL과 같은 방대하고 복잡한 정보기술을 비교할 때는 서로 비교할 수 있는 대응되는 분야나 비슷한 영역으로 국한하는 것도 바람직하다고 본다. 그런 분야와 영역이 존재하기 때문에 이 논문에서는 이 관점에서 두 기술을 비교하기로 한다. 두 기술이 다 '식별자' 기반 정보기술이라는 점에서는 맥락을 같이 하고 있다. 토픽맵에서 표현 대상(thing)에 해당하는 것을 '주제'라 하지만 RDF/OWL에서는 '자원(resource)'이라 칭한다. 토픽맵은 '주제'를 표현하기 위해서 '토픽'을 사용하고 RDF/OWL에서는 '노드(node)'를 사용한다.

6. 1 단언/주장(Assertions)

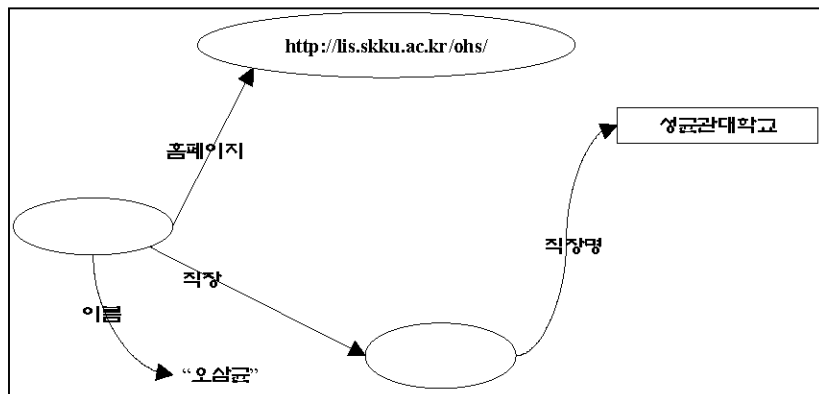
토픽맵과 RDF/OWL의 특성 중의 하나는 '주제'나 '자원'에 대해서 어떤 '단언'을 하거나 '주장'을 제시할 수 있다는 것이다. RDF/OWL

에서는 단 한가지 '주장'하는 방법을 허용하고 있다. 이것을 RDF 서술문(statement)이라 하는데 주어(resource), 술어(property), 목적어(object) 형태로 기술하며, 이 포맷으로 표현된 것을 트리플(triple)이라 한다. RDF 서술문을 노드를 사용해서 표현하면 다음과 같다.

그림 4는 RDF 서술문을 표현한 것으로 URI가 할당되지 않은 자원에 대해서 기술하는 것으로서 다음과 같은 다수의 RDF 서술문을 트리플로 표현하면 다음과 같다.

- ▶ 주어:annonymous1, 술어: 이름, 목적어: 오삼균
- ▶ 주어:annonymous1, 술어: 직장, 목적어: anonymous2
- ▶ 주어:annonymous1, 술어: 홈페이지, 목적어: http://lis.skku.ac.kr/ohs/
- ▶ 주어:annonymous2, 술어: 직장명, 목적어: 성균관대학교

반면 토픽맵에서는 '주제'에 대해서 '주장'이나 '단언'을 하는 세 가지 방법이 있다. 첫 번



(그림 4) RDF Assertion

제 방법은 토픽에 이름을 부과하는 것이다. 아래의 그림은 '오삼균'이라는 토픽을 생성한 것이다(그림 5).

두 번째 방법으로는 토픽에 '어커런스'를 연결하는 방식으로 '주제'에 대한 '단언'이나 '주장'을 표현하는 방식이다. 아래의 예는 '오삼균'이라는 토픽이 어커런스와 연결되어 있는데 그 자원과 토픽 간의 관계는 '홈페이지'라고 정의된 것을 그림으로 보여 주고 있다(그림 6).

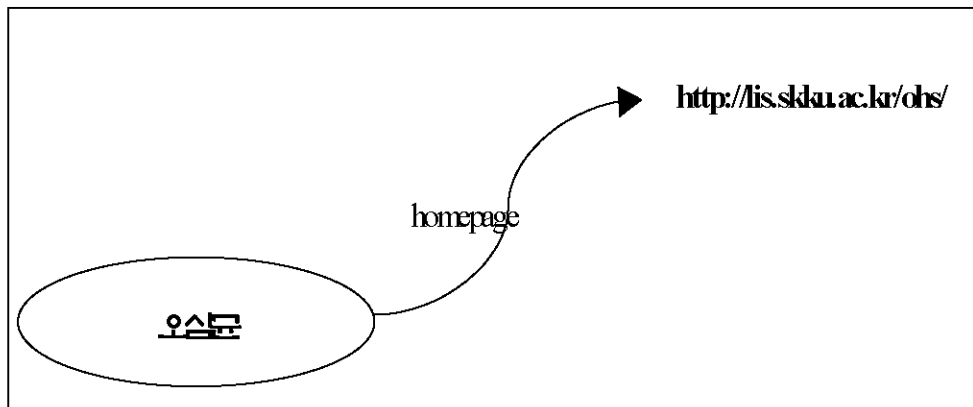
세 번째 방법은 연계를 통해서 토픽 간의 관계를 설정함으로써 '주제'에 대한 '주장'이나 '단언'을 표현하는 방법이다. 아래의 그림은 이미 전 그림에서 밝힌 '주장'에 새로운 정보를 추가하는 것으로 '성균관대학교'라는 토픽을 생성했고 이 토픽을 '오삼균'이라는 토픽과 연

계를 시켰고, 각 토픽을 역할을 규정하고 있다. '고용하다'라는 연계에 참여한 두 토픽의 역할은 '오삼균' 토픽은 '고용인'의 역할이고, '성균관대학교'는 '고용자'의 역할을 하고 있다는 사실을 표현하고 있다(그림 7).

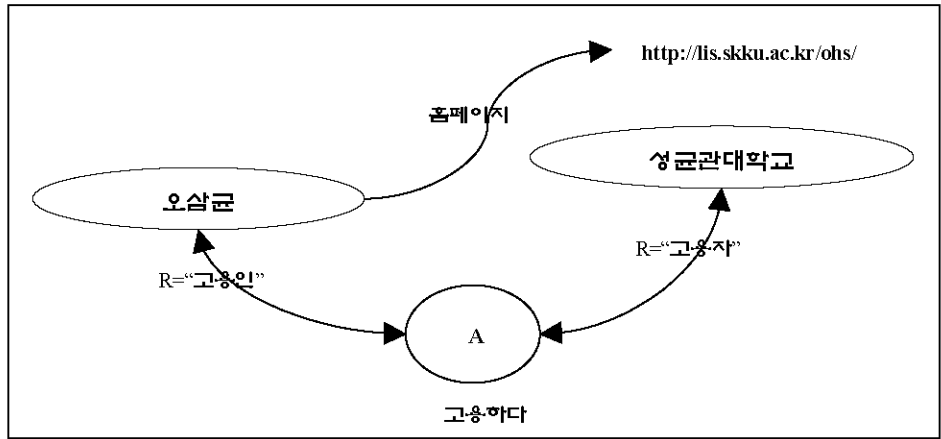
다음 그림은 토픽맵에서는 이진관계만 지원하는 것이 아니라 삼진관계를 지원하는 모습을 보이고 있다. '판매하다'라는 연계는 3개의 토픽이 동시에 참여하여 이뤄지는 관계를 보이고 있다. 자동차 판매가 이뤄지기 위해서는 구매자(토픽: 오삼균), 판매원(토픽: 정동일), 팔자동차(토픽: 카렌스) 등이 있어야 한다. 이와 같이 삼진관계를 사용한 '단언'이나 '주장'도 토픽맵에서 가능하다(그림 8).



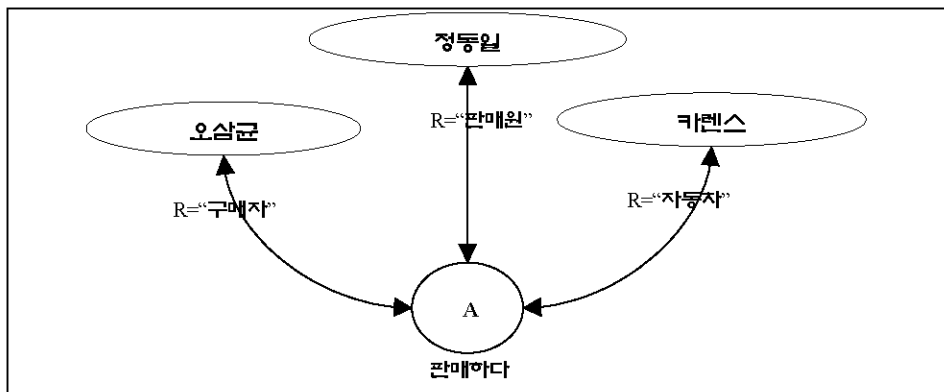
(그림 5) 토픽맵 Assertion 1



(그림 6) 토픽맵 Assertion 2



(그림 7) 토픽맵 Assertion 3



(그림 8) 토픽맵 Assertion 4

6. 2 식별자

W3C는 '자원'에 대해서 '식별자를 지니고 있는 어떤 것'이라 정의하고 있다. 그러나 이 정의에 대해서 Clack은 어느 곳에도 '식별자'가 무엇을 의미하는지 정의되어 있지 않은 점이 문제라고 제기한다. 여기에서 가장 중요한 문제는 과연 'URI가 무엇을 식별하느냐?'는

것이다. Berners-Lee는 'HTTP URI가 무엇을 식별하는가?', Booth는 '네 가지 용도의 URL', Hawke는 '모호성이 배제된 RDF 식별자'에 대한 논문들이 발표되었지만, 지금까지도 RDF는 'http://www.w3.org'와 같은 URI가 W3C를 식별하는 것인지 아니면 W3C에 관한 웹페이지를 식별하는 것인지에 대해서 분명한 입장을 취하지 않고 있다(Clark 2002:

Berner-Lee 2003; Booth 2002; Hawke 2003).

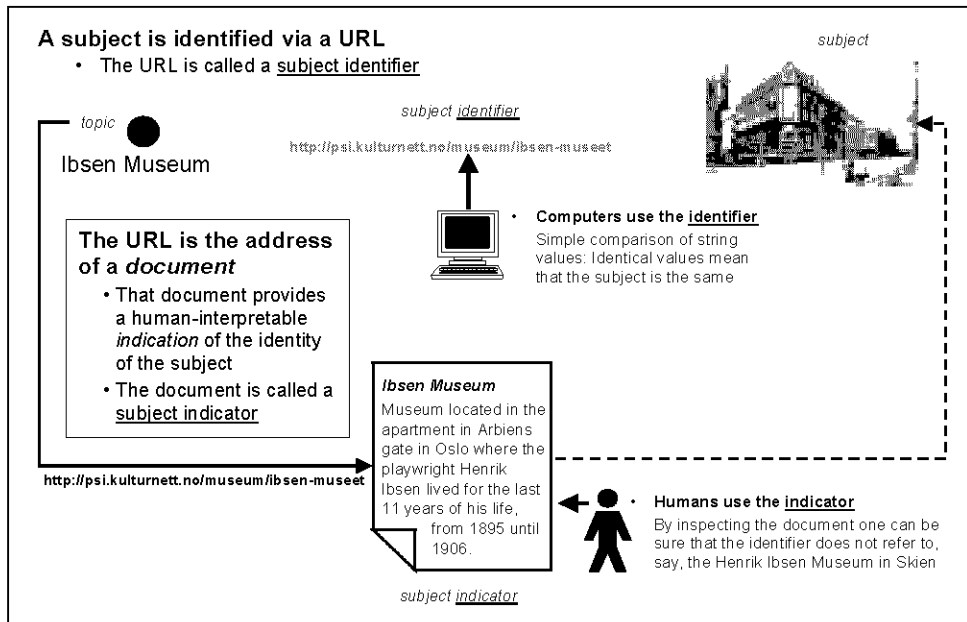
왜 명확한 식별체계가 중요한 것인가? 이 문제에 대한 명확한 답변 없이는 시맨틱웹의 도전을 해결할 길이 없고, 스케일이 큰 웹 서비스를 구축할 수 없기 때문이다. 또한 시맨틱웹이 추구하는 '글로벌 지식연합'도 불가능하고, 인간이나 소프트웨어 에이전트가 정보와 지식의 축적하여 정보의 홍수현상을 방지할 방법을 찾기 어렵게 된다. 다음은 이 개념을 도식화한 것이다.

그림 9에서 오른쪽 상단에 있는 그림이 오슬로에 있는 'Ibsen' 박물관의 모습이다. 박물관 자체에 식별자를 부과하고(<http://psi.kulturnett.no/museum/ibsen-museet>), 이 식별자는 컴퓨터 처리를 위한 것이다. 만약 이

식별자를 클릭 했을 경우에 이 박물관에 대한 소개를 자연어로 제공하면, 이 정보는 사람에게 유익을 주는 정보가 될 것이다. 이 문서를 '주제지시자'라 하고 이 두 개념을 분리해서 사용하는 것이 시맨틱웹을 구축하는데 필요하다는 것이다. 다시 말하면, 사람은 '주제지시자'를 사용하여 자원이나 주제에 대해서 이해를 하고, 컴퓨터는 '주제식별자'를 사용하여 URL을 비교함으로써 동일한 자원을 지칭하는 것인지에 대한 식별을 하게 된다. 이 두 개념에 대한 구분이 토픽맵에서는 명확한데 RDF는 없다.

6. 3 주제의 구체화(Reification)

온톨로지 표현에서 '구체화'의 의미에 대한 오해가 많은데, 간단하게 표현하면 '주제의 구



(그림 9) 토픽맵 주제식별자와 주제지시자(Pepper 2004)

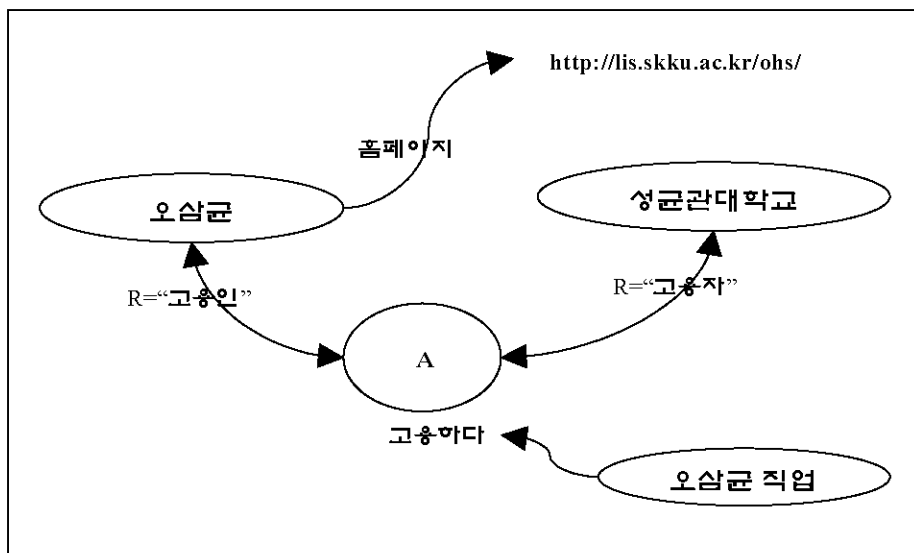
체화'는 '주장'에 관한 '주장'이다. 토픽맵에서 주제를 구체화하는 방법은 한 토픽을 생성하여 '주제식별자'를 부과하고 그 식별자가 이름, 어 커런스, 아니면 연계 중의 하나로 연결시키면 된다. 다음은 토픽맵 '구체화'를 도식화한 것이다. '오삼균 직업'이라는 토픽을 생성하여 '오삼균'과 '성균관대학교' 토픽 간의 '고용' 관계로 규정된 연계와 연결시킴으로 그 이미 생성된 '주장'이 '오삼균 직업'이라는 사실을 표현한 '구체화'의 한 예이다(그림 10).

그러나 RDF에서는 '구체화'의 방법이 다르고, 좀 어색하게 표현된다. RDF에서 '구체화'를 표현하기 위해서는 먼저 빈 노드를 하나 만들어 그 노드의 타입(rdf:type)을 rdf:statement로 규정해야 한다. 그리고 '구체화'를 표현하기 위한 특별 용어를 사용하는데, 주어의 표현에는 rdf:subject, 술어의 표현에는 rdf:property, 목적어의 표현에는 rdf:object를

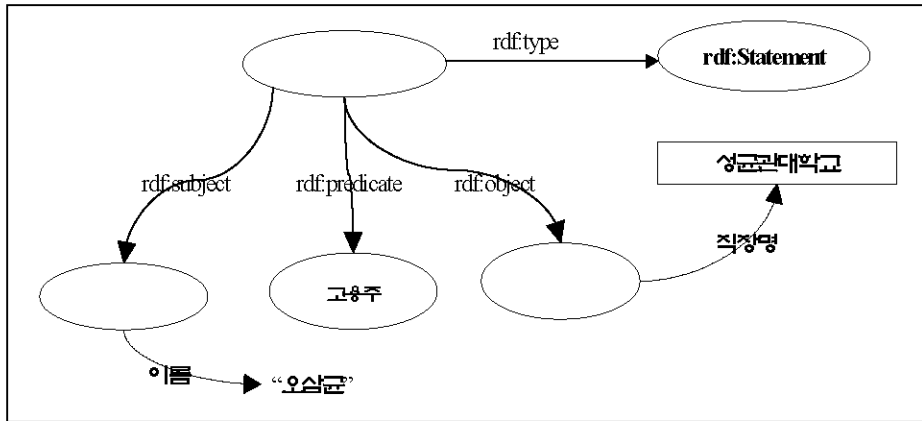
사용한다. 위와 똑 같은 내용을 RDF로 표현하면 아래와 같다(그림 11).

6. 4 제한사항(Qualification)

어떤 자원 또는 객체에 대해서 '단언'이나 '주장'을 할 때 종종 그 주장에 대한 제한 사항을 표현할 필요가 있다. 예를 들면, 그 주장을 펴는 기관이 누구인지, 그 주장이 어떤 상황에서 사실인지, 그 주장이 어떤 특정 언어로 표현되었다는 등의 일련의 조건 및 상황을 표현하는 것을 qualification이라 한다. 토픽맵에서는 이 기능이 '범위'로 표현된다. 기본적으로 토픽이 설정되면 아무 제한이 없는 '범위'가 자동적으로 부착된다. 특정 '범위'가 설정되면 기본 값을 대체한다. '범위'를 사용하여 토픽의 언어 표현할 수도 있고, 그 토픽과 연관된 어 커런스가 그 토픽에 대한 비평을 담고 있을 경



(그림 10) 토픽맵 Reification



(그림 11) RDF Reification

우 그 비평자의 이름을 '범위'로 표현할 수 있고, 또한 유명한 작품에 대한 저자가 다수 존재할 때에도 다른 저자에 관한 정보를 'scope'으로 처리할 수 있기 때문에, 이렇게 '범주화'된 토픽맵을 검색할 때 범위 정보가 필터링의 기준이 될 수 있다.

그러나 RDF에서는 이러한 제한 사항을 RDF 구체화 방법을 사용해서 표현할 수는 있으나 어색하고 자연스럽지 못한 면이 있다. 문제의 핵심은 RDF 서술문이 식별자가 없다는 것이다. RDF 모델이 제한을 위한 제한 방법을 직접적으로 지원하지 않기 때문에 '구체화'를 사용해서 제대로 '제한사항'을 표현할 수 없다는 것이다. 이점이 RDF와 토픽맵의 근본적인 차이점이다. 이 차이점이 토픽맵을 RDF로, 그리고 RDF를 토픽맵으로 모델화 하려는 많은 노력을 어렵게 만든 문제의 근원이다.

6. 5 타입과 하위타입(Types과 Subtypes)

우리가 표현하려는 중요한 '단언'이나 '주장'

중의 하나는 기술하려는 대상이 어떤 타입에 속하는가에 관한 것이다. 예를 들면, '모든 논문을 일종의 문서이나, 모든 문서가 다 논문은 아니다.'라는 사실을 표현하기 위해서는 타입을 사용해서 표현한다. 이런 경우에는 '논문'은 '문서'의 '하위타입'이 되는 것이다.

RDF에서는 '자원'에 관한 타입 정보가 'rdf:type'이라는 속성을 사용하여 표현되고, 하위 타입에 관련된 정보의 표현은 RDF 스키마의 속성 중의 하나인 rdfs:subClassOf를 사용한다. 토픽맵에서는 토픽 타입, 어커런스 타입, 연계 타입 등의 표현이 가능하고, 또한 클래스들 간의 계층구조는 subpertype-subtype 관계를 사용하여 표현할 수 있다.

6. 6 정보 통합(Merging)

'통합'은 분산된 정보를 같은 자원이나 주제에 대한 '단언'이나 '주장'들을 한 곳으로 모으는 작업을 의미한다. RDF에서는 노드의 값이 동일하거나 URI가 같은 경우에 통합시키고,

그 외에는 통합에 대한 필수사항은 없다. 고급 수준의 어휘를 지니고 있는 DAML+OIL과 OWL은 노드의 식별자를 서술문들의 집합에서 유추할 수 있는 기능을 제공함으로써 확장된 통합의 방법을 제공하고 있다. 토픽맵에서는 '토픽'의 동일한 '식별자' 값을 지니고 있는 것들을 통합한다. 그 외에 통합에 관한 다른 요구사항은 없다.

7. 결론

지금까지의 비교를 보면 RDF와 토픽맵이 서로 유사하면서도 또한 다른 점이 있다는 사실을 알게 된다. 가장 핵심적인 차이점을 정리하면 다음과 같다.

- ▶ UIR가 자원이나 주제를 식별하는데 사용되는 방식에서의 차이점(RDF는 한 방식으로, 토픽맵은 두 방식으로: subjectIdentifier와 subjectIndicator)
- ▶ 토픽맵은 세 가지 '주장'과 '단언'하는 방법이 있는 반면에 RDF에서는 한 방법을 제공한다.
- ▶ '구체화'의 표현 방법과 '주장'과 '단언'에 대한 '제한사항'을 표현하는 방법에서의 차이점.

이러한 3가지 차이점이 토픽맵과 RDF를 하나의 기술로 통합하는 것을 아주 어렵게 하

고 있는 것을 사실이다. 두 용도로 사용되는 URI와 세 가지 다른 방법의 '주장'이나 '단언'은 쉽게 해결되는 사항인데, '구체화'와 qualification에 관한 사항은 RDF 서술문 체계의 변화 없이 가능해 보이지 않는다. 분석에 의하면, 토픽맵이 RDF에 비해서 RDF 모델이 표현하는 것보다 더 풍부한 내용을 담을 수 있다는 점에서 고급수준의 언어이라는 사실은 자명하다. 결론적으로 RDF는 보다 자원에 관한 메타데이터의 상호운용을 증진시키기 위한 표현이고, 토픽맵은 자원을 주제를 중심으로 융통성 있고 효율적으로 연결시켜서 유의한 브라우징과 검색을 가능케 하는데 초점이 맞춰진 기술이다.

RDF/OWL과 토픽맵은 서로 경쟁적 관계에 있는 것처럼 보이나 사실은 보완적 관계에 있다고 보아야 옳다. 현재 W3C와 ISO 간에 상호운용을 위한 Task Force 팀이 구성되어 데이터 호환에 관한 모든 문제를 협의 중에 있다. RDF/OWL은 추론의 능력이 뛰어나 triple로 구성된 지식 데이터베이스에서 추론을 통한 새로운 지식의 창출과 지식의 연계를 가능케 하며, 토픽맵은 전 세계의 지식과 정보를 주제를 중심으로 융통성 있게 엮으므로 아주 유의한 정보와 지식서비스를 가능케 한다. 앞으로 데이터의 상호변환이 가능케 될 것이므로 두 기술을 최대도 활용할 수 있는 시대가 오고 있으니, 문헌정보학의 연구에 큰 도움이 될 것으로 생각된다.

참 고 문 헌

- 오삼균. 2002. 시맨틱웹 기술과 활용방안. 『정보관리학회지』, 19(4): 297-320.
- 조재인. 2003. 연관 저록의 구조적 접근을 위한 목록 체계 구축 및 평가에 관한 연구: 객체지향적 접근. 『한국문헌정보학회』, 37(3): 15-34.
- 허정희, 강상욱, 윤세진, 오삼균, 서태설, 김희석. 2003. 『국가 URN기반 UCI 식별체계의 식별메타데이터 연구』. 한국전산원.
- Berners-Lee, Tim. What do HTTP URIs Identify? February 15 2003. URL: <<http://www.w3.org/DesignIssues/HTTP-URI>>.
- Booth, David. Four Uses of a URL: Name, Concept, Web Location and Document Instance. January 28 2003. <http://www.w3.org/2002/11/dbooth-names/dbooth-names_clean.htm>.
- Bray, T., D. Hollander, and A. Layman. 1999. "Namespaces in XML." <<http://www.w3.org/TR/REC-xml-names/>>.
- Clark, Kendall Grant. Identity Crisis. XML.com, September 11, 2002. URL: <<http://www.xml.com/pub/a/2002/09/11/deviant.html>>
- DCMI. 1999. "Dublin Core Qualifiers in RDF." <<http://purl.org/dc/terms/>>.
- Garshol, LM. 2004. Metadata? Thesaurus? Texonomies, Topic Maps?: Making sense of it all, XML 2004 Conference and Exposition. April 19-21, 20024 Amsterdam, Netherland.
- <<http://www.ontopia.net/topicmaps/materials/tm-vs-thesauri.html>>.
- Hawke, Sandro. Disabiguating RDF Identifiers. January 4 2003. URL: <<http://www.w3.org/2002/12/rdf-identifiers/>>.
- Miller, E. 1998. "An introduction to the Resource Description Framework." <<http://www.dlib.org/dlib/may98/miller/05miller.html>>.
- Organization for the Advancement of Structured Information Standards. 2004. <http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=tm-pubsubj>.
- Pepper, S. and L. Garshol. 2004. Toward seamless knowledge: integrating public sector portals in Norway. XML 2004 Conference and Exposition. April 19-21, 20024 Amsterdam, Netherland.
- Pepper, Steve & Schwab, Sylvia. 2003. Curing the Web's Identity Crisis: Subject Indicators for RDF. URL: <<http://www.ontopia.net/topicmaps/materials/identitycrisis.html>>.
- Pepper, S. and L. Garshol. 2002. The XML Papers: Lessons Learned on Apply-

- ing Topic Maps. XML 2002 Conference and Exposition. December 8-13, 2002. Baltimore, Maryland, USA.
- Powell, A., H. Wagner, S. Weibel, T. Baker, T. Matola, and E. Miller. 2001. "Namespace Policy for the Dublin Core Metadata Initiative (DCMI)." <http://dublincore.org/documents/2001/10/26/dcmi-namespace/>.
- Smith, M., D. McGuinness, R. Volz, and C. Welty. 2002. "Web Ontology Language(OWL) Guide Version 1.0." <http://www.w3.org/TR/owl-guide/>.
- Stevens, Roberts, et al. 2003. Building ontologies in DAML + OIL. *Comparative and Functional Genomics*, 4(1): 134-141.