

공리정의를 이용한 인명접근점제어 온톨로지 기술에 관한 연구

A Study on the Description of Personal Name Access Point Control Ontology Using Axiom Definition

강 현 민(HyenMin Kang)*

목 차

- | | |
|-------------------|------------------|
| 1. 서 론 | 3.2 접근점간 연결 |
| 2. 이론적 배경 | 4. 공리정의와 온톨로지 기술 |
| 2.1 공리(Axiom)의 개념 | 4.1 공리정의 |
| 2.2 RDF/OWL의 공리정의 | 4.2 온톨로지 기술 |
| 3. 인명접근점제어 온톨로지 | 5. 결 론 |
| 3.1 인스턴스와 접근점 | |

초 록

본 연구는 온톨로지를 기반으로 접근점을 제어하기 위해, 미국 작가 마크 트웨인을 대상으로 인물과 이름 간의 다양한 관계와 특질을 RDF/OWL의 공리정의를 통해 인명접근점제어 온톨로지를 기술하였다. 적용된 공리정의는 인물·이름클래스 간 서로 소 공리정의, 객체속성의 정의역·치역 공리 정의, 인물·이름클래스의 객체속성 값 출현횟수 제약 공리정의, 인물과 이름 개체 간 역함수 관계 공리정의, 인스턴스와 리터럴 간 데이터속성 공리정의 등이다. 그 결과 온톨로지 기반에서 전거와 표목의 개념을 배제하고 모든 표현형식을 대등하게 다룬 대등형접근점이 기존의 제어기능을 수용할 수 있음을 확인하였고, 다양한 공리정의로 보다 강화된 표현력으로 공유성과 유일성이 광역적으로 확보된 인명접근점제어 온톨로지를 기술할 수 있었다.

ABSTRACT

This study tries to describe personal name access point control ontology for the American novelist Mark Twain using RDF/OWL axiom to control access point based on the ontology. The Axiom used in this study are disjoint with class, domain and range, property cardinality, inverse functional property, individual and literal data property. As a result, in the ontology environment we can accept various access points as equal access points exclusive of authority heading and heading concept. It can successfully describe Mark Twain's personal name access point control ontology and display using the OntoGraf.

키워드: 접근점 제어, 대등형접근점, 온톨로지, 공리

Access Point Control, Equal Access Point, Ontology, Axiom

* 행정안전부 국가기록원(xianmin70@yahoo.co.kr)

논문접수일자: 2012년 4월 22일 최초심사일자: 2012년 4월 23일 게재확정일자: 2012년 5월 14일
한국문헌정보학회지, 46(2): 157-174, 2012. [http://dx.doi.org/10.4275/KSLIS.2012.46.2.157]

1. 서론

전통적으로 카드목록시대 뿐만 아니라 MARC 포맷의 자동화목록시대에서도 인명전거레코드는 특정 저자의 대표이름형식을 선정하여 이를 표준형식으로 삼아 특정 저자의 다양한 이름으로 저작된 모든 관련 문헌들을 한데 모으는 집중의 주요 수단으로 사용해 왔다. 이를 위해 기본표목, 상호참조 등의 연결장치를 통해 여러 형식의 채택표목들과 채택되지 않은 형식들을 제어하여 대표이름형식으로 한데 모으는 수단으로 사용하였다.

그러나 대표이름형식인 기본표목의 저작단위 집중기능은 기본표목만의 고유한 기능이 아니며 기본표목의 선정여부와는 무관하게 모든 데이터 요소를 접근점으로 사용할 수 있고 특히 서지 정보의 국제적인 공유라는 관점에서 특정 형식의 표목을 기본표목으로 유지하는 것은 의미 없는 일이다(김태수 1996, 52-57). 전통적인 카드목록시대의 저록에서 전거와 표목의 개념과 용어가 사라지고 모든 표현형식이 대등한 접근점의 개념이 대두되면서 MARC포맷 기반의 전거레코드나 XML포맷의 메타데이터 영역에서 이를 적용하기 위한 다양한 시도가 있었다.

그러나 변화된 전거개념을 구현하기 위한 새로운 저록의 형식도 국가 간 저록작성 규칙의 상이성, 저록을 담는 매체의 물리적·구조적 한계성뿐만 아니라 관계형데이터 구조에 기반한 전거·접근점 DB는 로컬시스템을 벗어나 스키마가 서로 다른 이형DB로 이식이 어려우며, 의미의 일관성을 유지하면서 유일성이 확보된 데이터를 구축하기가 불가능하다. 결국 기존 저록의 근본적인 문제를 해결하지 못하고 있다.

반면, 웹 기반의 온톨로지 환경에서는 동일한 이름공간 내에서 동일한 어휘스카마를 사용하면 의미의 일관성, 데이터의 유일성 및 범용성을 확보할 수 있으며, 상이한 기관에서 구축한 온톨로지의 병합을 통하여 특정 인명의 고유한 식별과 의미를 유지하면서 데이터를 재활용하고 공유할 수 있는 장점이 있다. 그 사례로 생의학 분야의 OBO(The Open Biomedical Ontologies) 온톨로지는 현재 관리되고 있는 온톨로지 내 다양한 관계어휘와 정의들을 온톨로지화 한 것으로, 온톨로지 개발자나 이용자들이 코딩이나 주석을 달 때 혼돈을 피하고, 온톨로지 내에 사용된 관계어휘 표현의 공식적 정의를 통해 일관성과 명확성을 제공하기 위한 것이다. 이는 온톨로지 간 상호운용성을 개선하고 생물학과 의학에서 시공간적 차원에서 다양하게 일어나는 현상들에 대해 새로운 형태의 자동화된 추론을 지원하기 위한 것이다(Mungall 2009, 37).

본 연구에서는 기존 전거레코드의 전거표목과 이형표목의 개념을 배제하고 또한 제어형접근점이든 비제어형접근점이든 모든 표현형식을 대등한 접근점으로 인정하고 의미의 일관성, 식별의 유일성과 광역성이 확보된 저록의 기능을 온톨로지 환경에서 구현하고자 하였다. 먼저 RDF/OWL 관련 문헌을 통해 웹 온톨로지 언어의 정의와 특징에 대한 이론적 분석을 하였고, FRAD 개념모형을 참조하여 전거개념이 배제된 '대등형접근점' 개념을 적용하고자 하였다. 적용 대상에는 단체명, 서명, 주제명, 인명 등 다양한 대상이 있지만 전거분야에서 가장 널리 연구되고 있는 인명을 선정하였다. 이에 인명전거레코드를 대상으로 온톨로지 저작 어플리케이션을 사용하여 인명접근점제어 온톨로지를 기

술하였으며 다음과 같은 방법을 시도하였다.

첫째, FRBR의 제2집단의 '인물'개체와 FRAD의 '이름'개체를 각각 클래스로 정의하였다. 온톨로지 상의 클래스는 관계형 데이터베이스에서 각각 인물테이블과 이름테이블로 이해할 수 있기 때문이다.

둘째, 정의된 인물·이름클래스 내 인스턴스 간 다양한 관계를 기술하기 위한 17종의 객체속성을 정의하였다. 객체속성은 인스턴스 간 이항 관계를 의미적으로 '기술'하며 물리적으로도 상호 '연결과 접근' 기능을 수행할 수 있다.

셋째, RDF/OWL에서 제공하는 다양하고 강화된 표현력을 지닌 공리를 정의하여 특정인물의 상이한 이름형식을 효과적으로 제어하기 위한 인명접근점제어 온톨로지를 기술하였다. 사용된 공리정의는 1) 인물·이름클래스 간 서로소(疎) 공리정의, 2) 객체속성의 정의역·치역 공리 정의, 3) 인물·이름클래스의 객체속성 값출현횟수 제약 공리정의, 4) 인물과 이름 개체 간 역함수 관계 공리정의, 5) 인스턴스와 리터럴 간 데이터속성 공리정의 등을 선언하였다.

온톨로지 기술은 보다 세밀하고 강화된 표현력을 지닌 공리를 제공하는 OWL Full 수준의 웹 온톨로지 언어를 사용하였으며 온톨로지 저작도구인 Protégé 4.2 베타버전을 사용하여 인명접근점제어 온톨로지를 기술하였다. 이때 선택한 특정인물은 마크 트웨인이라는 필명으로 유명한 Clemens, Samuel Langhorne으로서 다양한 이름형식을 갖는 접근점들을 기술대상으로 삼았다. 마지막으로 온톨로지의 클래스와 인스턴스 간 관계를 시각화하는 뷰어 플러그인으로 OntoGraf를 사용하였다. 이 온톨로지에 사용된 URI는 'http://www.kanghm.org/acceescont

rol'이며 QNAME 접두사는 'ac'로 정의하였다.

본 연구는 온톨로지 마크업 언어인 RDF/OWL을 활용하여 서지세계의 대등한 접근점을 제어하고자 온톨로지를 기술한 것이다. 그러나 현재의 시맨틱 웹 온톨로지 기술(technology)과 표준화의 문제로 인해 기술된 온톨로지에 대한 시맨틱 질의는 시도하지 못하였다.

2. 이론적 배경

2.1 공리(Axiom)의 개념

공리(公理, Axiom)의 어원은, 그리스어인 ἀξίωμα(axioma)에서 왔으며, '가치가 있다고 간주되거나 그 자체로 명백하다'라는 의미를 가지고 있다. 공리는 이론체계 가운데에서 가장 기초적인 근거가 되는 명제(命題)이다. 어떤 다른 명제들을 증명하기 위한 전제로 이용되는 가장 기본적인 가정을 가리킨다. 지식이 참된 것이 되기 위해서는 근거가 필요하고 그 근거를 소급해 보면 더 이상 증명하기가 곤란한 명제에 다르게 되는데 이것이 바로 공리이다.

2004년 W3C의 권고안이 된 OWL(Web Ontology Language)은 웹 온톨로지 마크업 언어로서 RDF와 RDFS의 문제점을 보완하고 개념이나 자원의 관계를 정밀하게 나타내거나 추론 가능한 논리를 기술하기 위한 보다 세련된 어휘이다. OWL은 RDFS의 클래스와 속성의 의미를 그대로 수용하여 RDF/OWL라고도 불리며, 보다 풍부한 표현력을 지원하는 근원적 언어를 추가했다는 점에서 RDFS의 확장이라 할 수 있다(Grigoris Antoniou etc. 2008, 116).

〈표 1〉 공리의 유형

클래스 공리	객체속성 공리	데이터속성 공리
• oneOf, dataRange • disjointWith • equivalentClass (applied to class expressions) • rdfs:subClassOf (applied to class expressions) • minCardinality • maxCardinality • cardinality • unionOf • complementOf • intersectionOf • hasValue	• SubObjectPropertyOf • EquivalentObjectProperties • DisjointObjectProperties • InverseObjectProperties • ObjectPropertyDomain • ObjectPropertyRange • FunctionalObjectProperty • InverseFunctionalObjectProperty • ReflexiveObjectProperty • IrreflexiveObjectProperty • SymmetricObjectProperty • AsymmetricObjectProperty • TransitiveObjectProperty	• SubDataPropertyOf • EquivalentDataProperties • DisjointDataProperties • DataPropertyDomain • DataPropertyRange • FunctionalDataProperty

자료: W3C 2009

RDF/OWL에서의 공리란 추론의 기본이 되는 명제로서 증명을 할 수 없거나 증명이 필요하지 않는 ‘참’(true)으로 인정되는 문장을 말한다. RDF/OWL의 공리는 온톨로지에서 이름과 자연언어에 의한 설명이 아니라 형식논리(formal logic)를 토대로 표현한 기술 또는 정의를 말한다(Gruber 1993, 9). 이때 형식논리는 판단과 추리의 추상적 구조를 내용과 분리하여 순수 수학분야처럼 선형적 지식에 기반하여 논리를 형식언어로 표현하는 것을 말한다. RDF/OWL에서 제공하는 공리의 유형에는 클래스 공리, 객체속성 공리와 데이터속성 공리 등이 있다(〈표 1〉 참조).

2.2 RDF/OWL의 공리정의

RDF/OWL 온톨로지에서의 공리는 기본문(basic statements)이라고도 한다. 이 기본문은 정의문(definition statements), 선언문(assertion statements), 표현문(expression statements), 기술문(description statements)이라고도 하여

사전적인 공리의 개념보다 광의의 개념으로 사용된다. 본 논문에서는 ‘공리를 선언하여 온톨로지를 기술한다’는 의미로 ‘공리 정의’라는 용어로 통일하여 사용하였다. 주요 공리정의에는 클래스 공리정의, 객체속성 공리정의, 데이터속성 공리정의 등이 있다.

먼저 RDF/OWL의 클래스 공리정의에는 1) URI참조에 의한 클래스의 명명, 2) 클래스의 인스턴스가 되는 멤버개체의 열거, 3) 속성의 제약, 4) 2개 이상의 클래스 표현의 교집합, 5) 2개 이상의 클래스 표현의 합집합, 6) 클래스 표현의 부정 등 6가지 방법이 있다(Kanzaki 2005, 35-47)(〈표 2〉 참조).

URI참조에 의한 클래스의 명명은 클래스를 규정하는데 있어서 클래스 표현의 가장 기본적인 요소이다. 클래스의 인스턴스가 되는 멤버개체의 열거나 속성의 제약, 2개 이상의 클래스 표현의 교집합·합집합, 또는 클래스 표현의 부정 등은 ‘해당 클래스의 외연의 멤버이기 위한 조건’을 나타내는 클래스 공리정의 방법이다.

RDF/OWL의 객체속성 공리정의는 첫째,

〈표 2〉 클래스 공리정의 방법

구 분	클래스 공리 정의	정의유형
1	URI참조에 의한 클래스의 명명	클래스명명
2	클래스의 인스턴스가 되는 멤버개체의 열거	클래스멤버열거
3	속성의 제약	속성제약
4	2개 이상의 클래스 표현의 교집합	클래스논리조합
5	2개 이상의 클래스 표현의 합집합	
6	클래스 표현의 부정	

자료: Kanzaki 2005 재구성

관계하는 주어 클래스인 정의역과 목적어 클래스인 치역을 기술함으로써 객체속성의 외연(extension)을 정의할 수 있다. 즉, 어떤 객체속성을 취할 수 있는 클래스의 범위를 제한하고 그 객체속성이 취할 수 있는 인스턴스의 범위를 제한할 수 있다. 둘째, 독립형 클래스로 정의되며 객체속성 간 추이적 계층관계를 통한 외연 상속이 가능하다. 셋째, 동치관계, 반대관계 등 다른 객체속성 간 관계표현과 함수형, 역함수형 등 객체속성의 출현횟수에 대한 제약선언과 추이성, 대칭성 등 객체속성의 논리적 특질 등을 통해 강화된 표현력을 갖는 공리정의가 가능하다.

특히 객체속성 공리정의는 특정 클래스와 함께 사용되는 속성의 조건을 규정하는 클래스의 속성제약과는 달리, 어떤 클래스에 적용할 것 인지를 묻지 않는 전역적인 성질을 나타낸다. 객체속성의 공리정의 방식을 좀 더 상세히 살펴보면 다음과 같다.

첫째, OWL의 객체속성은 기본속성공리에 의해 정의된다. OWL의 객체속성은 RDFS의 `rdfs:domain`과 `rdfs:range`를 통한 기술방식과 동일한 방식으로 개체와 개체 간 관계를 기술하기 위한 속성으로 정의되며 이를 기본속성 공리정의라 한다.

둘째, 다른 객체속성과의 관계에 의한 정의이다. 객체속성은 계층관계(`rdfs:subPropertyOf`), 동치관계(`owl:equivalentProperty`), 반대관계(`owl:inverseOf`) 등을 사용하여 다른 객체속성과의 관계에 의해 공리를 정의할 수 있다.

셋째, 객체속성 출현횟수에 대한 전역적 제약을 통한 정의이다. 주어가 되는 개체의 객체속성의 출현횟수가 1로 규정된 함수형 속성과 이와 반대로 목적어가 되는 개체의 객체속성의 출현횟수가 1로 규정되는 역함수형 속성을 들 수 있다.

넷째, 객체속성의 논리적인 특질에 대한 정의이다. 속성이 추이형 속성(`owl:TransitiveProperty`), 대칭형 속성(`owl:SymmetricProperty`) 인지를 정의하면 이런 객체속성의 특질을 통해 관계추론을 수행할 수 있다.

3. 인명접근점제어 온톨로지

3.1 인스턴스와 접근점

온톨로지에서 특정인물에 대한 이름 인스턴스는 동일한 이름공간 내에서 유일한 문자열로 정의되어 동일 클래스 내에서 상호 배타적이며

독립적으로 존재한다. 이는 특정인물에 대한 모든 표현형식이 유일하며 대등한 문자열로 인정하는 최근의 전거제어 개념의 접근점과 유사하다. 따라서 이름클래스의 인스턴스도 대등한 접근점으로서 제어하기 위한 대상이다.

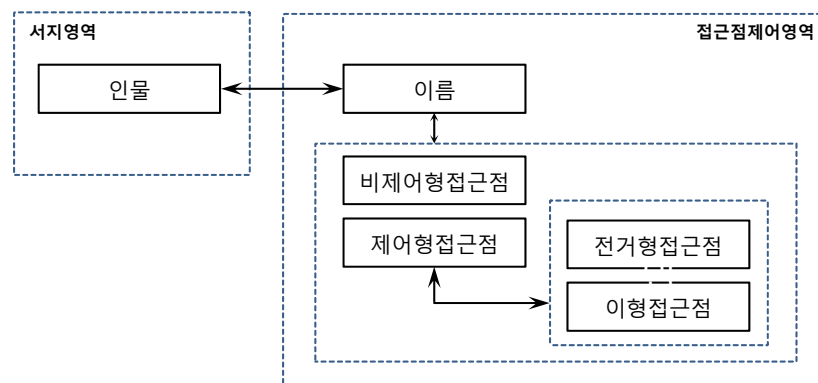
FRBR, FRAD 개념모형에는 23종의 서지세계 개체들이 존재한다. 온톨로지에서 이 개체들은 클래스로, 그 개체들의 특정사례들은 인스턴스로 정의할 수 있다. 공리정의에 의해 기술되는 인명접근점제어 온톨로지는 FRBR 개념모형의 인물개체를 하나의 클래스로, FRAD 개념모형의 이름개체만을 클래스로 정의하였으며 이때 정의역은 인물클래스이며 치역은 이름클래스이다. 특정인물의 사례와 특정이름의 사례를 각각 인스턴스로 정의하였다. 인물 인스턴스와 이름 인스턴스를 객체속성을 사용하여 연결하였다. 이때 객체속성은 술어기능을 하게 된다.

〈그림 1〉은 인명접근점제어 온톨로지 관련 서지세계의 개체 관계를 도식화 한 것이다. FRAD 개념모형의 접근점제어영역에는 이름, 비제어형접근점, 제어형접근점, 전거형접근점, 이형접근점

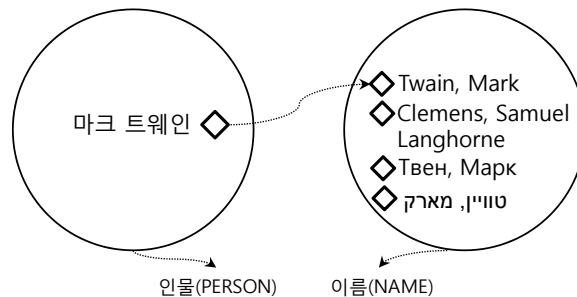
점 등의 개체들이 존재하는데, 인명접근점제어 온톨로지에서는 전거 및 표목이라는 제어형접근점 개념뿐만 아니라 비제어형접근점까지도 배제하고 오직 이름개체만을 유일한 '대등형접근점'을 인스턴스로 갖는 클래스로 정의하였다.

인명접근점제어 온톨로지 기술을 위해 본 연구에서는 마크 트웨인을 특정 인물사례 인스턴스로 선정하고, 그의 다양하고 상이한 형식의 이름들을 특정사례 인스턴스로서 공리정의를 통해 서로 연결하고자 하였다. 미국의 사실주의자 작가로서 1876년 톰 소여의 모험과 1884년 허클베리 핀의 모험이라는 소설을 쓴 마크 트웨인은 Twain, Mark이라는 필명을 사용했으며 본명은 Clemens, Samuel Langhorne이다. 또한 그는 여행가로서 Snodgrass, Quintus Curtius라는 이름을 사용했다. 이 이름들은 '馬克吐溫', '마크 트웨인', 'Klemens, Samüel', 'Твен, Марк', 'טוויין, קלאם' 등 다양한 언어와 문자형식으로 표현되며 이들 모두는 특정인물의 서지적 신원이 동일한 특정인물을 지시한다.

〈그림 2〉에서 인물클래스 내 특정인물의 서지적 신원인 마크 트웨인은 이름클래스 내에서



〈그림 1〉 인명접근점제어 온톨로지 관련 개체관계



〈그림 2〉 인명접근점제어 온톨로지 기본 개념도

표목이든 상이한 형식이든 모두 대등한 접근점으로서 하나의 인스턴스들로 존재한다. 이는 온톨로지 환경에서 특정 클래스의 인스턴스들은 전거와 표목의 개념을 배제하고 모든 표현 형식이 대등한 접근점으로서 공유성과 유일성을 동시에 확보하면서 효과적으로 제어할 수 있다는 것을 의미한다.

3.2 접근점간 연결

기계가독목록에서는 전거표목과 비전거표목

으로 나눌 수 있는데 전거표목은 1XX, 5XX 필드에 사용되며 비전거표목은 표목으로 채택되지 않은 형식으로서 전거레코드의 4XX 필드에 기술된다. 이들은 보라참조와 도보라참조 등 상호참조를 통해 전거표목으로 연결된다. 〈그림 3〉은 마크 트웨인의 인명전거레코드를 나타낸 것이다. 전거표목을 지시하는 1XX와 5XX, 비전을 표현하는 부호이자 일종의 어휘이기도 하다.

1XX, 4XX 및 5XX 필드를 제거하여 전거와 표목의 개념을 배제한다면 각 필드에 표기되어

LC control no.: n 79021164
LCCN permalink: <http://lcn.loc.gov/n79021164>

```
100 1_ |a Twain, Mark, |d 1835-1910
400 1_ |a Tvèn, Mark, |d 1835-1910
400 1_ |a Tuwen, Make, |d 1835-1910
400 0_ |a Make Teviin, |d 1835-1910
400 1_ |a Твен, Марк, |d 1835-1910
400 1_ |a קרמ, מַרְק, |d 1835-1910
400 1_ |a 馬克吐溫, |d 1835-1910
500 1_ |a Clemens, Samuel Langhorne, |d 1835-1910
500 1_ |a Snodgrass, Quintus Curtius, |d 1835-1910
500 1_ |a Conte, Louis de, |d 1835-1910
```

〈그림 3〉 MARC포맷의 마크 트웨인 전거레코드

있는 마크 트웨인의 본명, 필명, 어순, 철자, 표기언어 등 상이한 형식들은 모두 특정인물을 식별하는 문자열로 동일하게 사용될 수 있다. 다시 말해, 동일 인물에 대해 여러 형식의 이름이 대등한 접근점으로 남게 된다.

인명접근점제어 온톨로지에서는 이러한 상이한 형식의 대등한 접근점들을 이름클래스의 인스턴스로 정의하였다. 이때 인물클래스의 인스턴스와 이름클래스의 인스턴스를 연결하기 위해 객체속성을 사용하였다. 객체속성은 특정 클래스에 적용되고 있는 객체속성에 대해 클래스 정의 변경 없이 새로운 객체속성의 정의가 가능하다. 정의된 객체속성, 즉 어휘는 기존의 어휘를 확장하거나 상세화하여 새로운 어휘를 정의할 수도 있다는 것을 의미한다. 어휘는 누

구든지 정의할 수 있고, 또한 누가 정의한 어휘라도 다시 이용할 수 있다(Kanzaki Masahide 2005, 76). 따라서 1XX, 4XX 및 5XX 필드와 같은 부호의 기능 이외에 본명, 필명, 어순, 철자, 표기언어 등 의미의 기능을 모두 표현할 수 있는 다양한 어휘로 객체속성을 정의하였다.

FRAD와 RDA에서 정의되고 있는 인물개체와 이름개체 간의 관계를 어휘화하고자 하였다. FRAD와 RDA에는 의미는 동일하나 표현이 다른 관계지시자가 많다. 이 둘을 의미적 배타성을 유지하면서 고유한 표현으로 통합·정리하여 그 결과 <표 3>과 같이 17종의 객체속성 어휘를 생성하였다.

이 17종의 객체속성은 공리정의를 통해 인스턴스의 관계를 보다 세밀하게 표현할 수 있다.

<표 3> 인물·이름클래스 내 접근점 간 관계표현을 위한 객체속성 정의(17종)

구분	개체속성	의 미
1	OfficialName	공식명
2	RealName	본명
3	AlternativeLinguisticForm	대체언어형식
4	EarlierName	변경전이름
5	LaterName	변경후이름
6	SecularName	속세명
7	NameInReligion	종교적이름
8	Pseudonym	필명
9	SpellingVariations	철자이형형식
10	Transliteration	음역이형형식
11	PunctuationVariations	구두법이형형식
12	CapitalizationVariations	대문자이형형식
13	Inversions	도치이형형식
14	Permutations	어순이형형식
15	Bynames	별명이형형식
16	Nicknames	에칭이형형식
17	CourtesyNames	경칭이형형식

4. 공리정의와 온톨로지 기술

4.1 공리정의

4.1.1 인물·이름클래스 간 서로 소(疎) 공리 정의

인물클래스와 이름클래스의 모든 인스턴트들은 상호 배타적인 관계를 유지해야한다. 인물클래스 인스턴스가 이름클래스 인스턴스로 존재해서도 안 되며 이름클래스의 인스턴스가 인물클래스의 인스턴스로 동시에 존재해서는 안 된다는 의미이다.

이를 위한 공리정의에는 owl:disjointWith를 사용한다. owl:disjointWith는 클래스 공리를 기술하는 방법 중 클래스 속성 표현 중 하나로 주어클래스와 목적어클래스의 외연들이 완전 분리된 서로 소(疎)인 클래스임을 정의하는 공리이다. 즉 owl:disjointWith 속성 공리를 사용하여 인물클래스와 이름클래스 중 특정 클래스의 인스턴스가 다른 클래스의 인스턴스가 될 수 없음을 정의함으로써 주어가 되는 클래스와 목적어가 되는 클래스의 외연이 서로 소(疎)임을 나타낸다.

서지영역의 제2집단에 속하는 '인물'개체와 접근제어개체영역의 '이름'개체 간의 외연이 서로 소(疎)임을 owl:disjointWith 속성 공리정의를 사용한 RDF/XML 구문은 <그림 4>와 같다.

즉, 'ac:NAME클래스는 ac:PERSON클래스와 서로 소(疎)이다'를 공리 정의한 직렬화 구문이다. 이 공리정의를 통해 제2집단 인물클래스의 인스턴스로서 실존적 인물인 '마크 트웨인'은 이름클래스의 인스턴스로서 동일한 실존적 인물로서 '마크 트웨인'이 출현할 수 없다. 즉, 실존적 인물을 지칭하는 인물클래스의 '마크 트웨인'에 대해 이름클래스의 '마크 트웨인'은 인물을 문자로 표현하는 문자열로서, 이름의 여러 인스턴스 중 하나인 특정한 인스턴스일 뿐이다.

4.1.2 객체속성의 정의역·치역 공리 정의

인명접근제어를 위해 앞서 정의한 17종의 객체속성들은 인물클래스와 이름클래스의 인스턴스를 직접적으로 연결한다. 술어인 객체속성은 주어가 되는 인물클래스는 정의역으로, 목적어가 되는 이름클래스는 치역으로 공리정의한다. 이때 사용되는 공리정의는 <rdfs:domain>과 <rdfs:range>이다. 정의역과 치역을 정의하는 이유는 주어 인스턴스가 다른 클래스의 인스턴스가 될 수 없으며, 목적어 인스턴스도 다른 클래스의 인스턴스가 될 수 없도록 정의하기 위함이다. 주어와 목적어로 저작클래스나 단체클래스 등 다른 클래스에 속해있는 인스턴스가 올 수 없도록 하고 만약 인명접근제어 온톨로지를 기술할 때 17종의 객체속성의 주어와 목적어

```
<!-- http://www.kanghm.org/accesscontrol#NAME -->
<owl:Class rdf:about="&accesscontrol:NAME">
  <owl:disjointWith rdf:resource="&accesscontrol:PERSON"/>
</owl:Class>
```

<그림 4> 인물클래스와 이름클래스 간 서로 소(疎) 공리정의

가 다른 클래스의 인스턴스로 연결되면 리즈너가 온톨로지 기술 오류메시지를 보내게 된다.

〈그림 5〉는 서지영역의 제2집단에 속하는 '인물'개체와 접근점제어개체영역의 '이름'개체를 객체속성 `ac:Pseudonym`의 정의역과 치역을 〈`rdfs:domain`〉과 〈`rdfs:range`〉를 사용하여 공리를 정의한 RDF/XML 구문이다.

필명을 의미하는 객체속성 `ac:Pseudonym`은 정의역인 인물클래스의 인스턴스인 '마크 트웨인'을 주어로 하고 치역인 이름클래스의 인스턴스인 'Twain Mark'으로 연결할 수 있다. 즉, SPO 술어논리로 표현하면 '`ac:마크 트웨인 ac:Pseudonym ac:Twain Mark`'이다. 동일하게 17종의 모든 객체속성을 특정 인물 마크 트웨인에 대해 이름클래스의 인스턴스들만 연결할 수 있도록 정의할 수 있다.

4.1.3 인물·이름클래스의 객체속성 값 출현 횟수 제약

인명접근점제어 온톨로지를 기술하면서 17종의 객체속성에 대해 클래스에 대한 객체속성 값의 출현횟수를 제약하는 즉 관계차수를 제약하는 두 가지 공리정의가 필요하다.

첫째, 인물클래스 내 하나의 특정 인물 인스턴스는 이름클래스 중 하나의 기본형식 인스턴

스를 반드시 한번은 연결되어야 한다. 즉, 이름클래스 중 특정 이름 인스턴스를 기본형식으로 선정하여 인물클래스의 특정 인물 인스턴스를 객체속성과 반드시 한번은 연결해야 한다.

접근점제어레코드에서 1XX 필드처럼 전거형식이 아니라 표시용 디폴트 형식이며 이는 서지기관마다 1XX 필드의 형식을 달리할 수 있다. 1XX 필드는 하나의 특정형식을 전거형식으로 규정하지 않고 다양한 형식의 이름을 포함하여 접근점의 표시형식의 디폴트로 지시할 수 있다.

이와 마찬가지로 인물클래스에서 서지적 인물을 표시하기 위해서는 언어·문자의 표현형식에 상관없이 단지 기본값에 불과한 인스턴스가 반드시 하나는 필요하다. 이름클래스의 기본형접근점은 객체속성과 인물인스턴스를 연결할 때 선정할 수 있는데, 이름클래스 내에는 모든 인스턴스들은 채택형, 비채택형, 전거형, 이형 등 그 언어와 문자의 표현형식에 상관없이 모두 대등한 접근점인 인스턴스들이다.

이때 기본형접근점을 의미하고 표현하기 위해 본 연구에서는 객체속성 `ac:OfficialName`을 사용하였으며, 이름클래스의 다양한 인스턴스 중 하나를 연결하여 관계차수 1을 의미하는 `owl:qualifiedCardinality` 공리를 정의했다. `owl:qualif`

```
<!-- http://www.kanghm.org/accesscontrol#Pseudonym -->

<owl:ObjectProperty rdf:about="&accesscontrol:Pseudonym">
  <rdfs:subPropertyOf rdf:resource="&accesscontrol:KnownBy"/>
  <rdfs:range rdf:resource="&accesscontrol:NAME"/>
  <rdfs:domain rdf:resource="&accesscontrol:PERSON"/>
</owl:ObjectProperty>
```

〈그림 5〉 객체속성의 정의역·치역 공리정의

iedCardinality는 owl:Restriction 클래스를 정의역으로 하며 모든 인스턴스들에 대해서, 제약대상 속성의 출현횟수가 목적으로 지정된 값과 항상 같음을 나타내는데 만약 이 값이 1이면 해당 속성은 필수임을 의미한다.

〈그림 6〉은 정의역인 인물클래스의 인스턴스는 객체속성 ac:OfficialName을 통해 치역인 이름클래스의 인스턴스를 연결 시 관계차수가 반드시 '1'임을 나타낸 RDF/XML구문이다. 즉, '주어 ac:PERSON의 인스턴스는 목적어 ac:NAME의 인스턴스를 갖는데, 이때 객체속성 ac:OfficialName과 목적어 ac:NAME클래스와의 관계차수는 '1'이다'라는 의미이다. 이 공리정의를 통해 이름클래스의 인스턴스는 본명이든 필명이든 가명이든 널리 알려진 특징이름 하나를 기본형식으로 선택하여 반드시 하나는 가져야 한다는 것이 기술되었다.

둘째, 이름클래스 내 기본형식으로 지정된 인스턴스 이외의 나머지 인스턴스들은 출현횟수와는 상관없이 자유롭게 연결되어야한다. 즉, ac:OfficialName 이외의 나머지 16종의 객체

속성은 기본형식의 인스턴스를 제외하고 철자이형이든, 어순이형이든, 별명이형이든 상관없이 기타 상이한 형식의 인스턴스들과 연결하기 위해서는 관계차수가 0...n로 정의되어 출현횟수에 상관없이 자유롭게 연결될 수 있어야한다.

이를 위해 owl:someValuesFrom라는 공리를 정의하였다. owl:someValuesFrom으로 정의된 객체속성은 이름 인스턴스를 연결 안 될 수도 있으며 또는 복수로도 연결이 가능하다는 의미이다.

〈그림 7〉의 RDF/XML구문은 '주어 ac:PERSON의 인스턴스는 목적어 ac:NAME의 인스턴스를 갖는데 이때 객체속성 ac:SpellingVariations과 목적어 ac:NAME클래스와의 관계차수는 '0...n'이다'라는 의미이다. 즉, 객체속성 ac:SpellingVariations은 치역인 이름 인스턴스를 0 또는 1 이상 연결될 수 있다는 공리 정의이다.

만약 마크 트웨인이 철자이형식의 이름(예: T..Mark 또는 Twain, M. 등)이 있을 경우 철자이형식을 의미하는 객체속성 ac:SpellingVaria

```
<!-- http://www.kanghm.org/accesscontrol#OfficialName -->

<owl:ObjectProperty rdf:about="&accesscontrol:OfficialName">
  <rdfs:range rdf:resource="&accesscontrol:NAME"/>
  <rdfs:domain rdf:resource="&accesscontrol:PERSON"/>
  <rdfs:range>
    <owl:Restriction>
      <owl:onProperty rdf:resource="&accesscontrol:OfficialName"/>
      <owl:onClass rdf:resource="&accesscontrol:NAME"/>
      <owl:qualifiedCardinality rdf:datatype="&xsd:nonNegativeInteger">1</owl:qualifiedCardinality>
    </owl:Restriction>
  </rdfs:range>
</owl:ObjectProperty>
```

〈그림 6〉 인물클래스의 출현횟수 제약 공리정의(관계차수: 1)

```

<!-- http://www.kanghm.org/accesscontrol#SpellingVariations -->

<owl:ObjectProperty rdf:about="&accesscontrol:SpellingVariations">
  <rdfs:range rdf:resource="&accesscontrol:NAME"/>
  <rdfs:domain rdf:resource="&accesscontrol:PERSON"/>
  <rdfs:range>
    <owl:Restriction>
      <owl:onProperty rdf:resource="&accesscontrol:SpellingVariations"/>
      <owl:someValuesFrom rdf:resource="&accesscontrol:NAME"/>
    </owl:Restriction>
  </rdfs:range>
</owl:ObjectProperty>

```

〈그림 7〉 인물클래스의 출현횟수 제약 공리정의(관계차수: 0...n)

tions을 사용하여 1회 이상 연결이 가능하다. 이와 마찬가지로 오직 하나의 인스턴스만을 취할 수 있는 객체속성 ac:OfficialName을 제외한 나머지 16종의 객체속성들은 이름클래스의 여러 인스턴스들을 0...n 횟수로 연결할 수 있다.

이 공리정의는 특정인물이 다양한 유형의 상이한 형식이름을 갖는 경우 자유롭게 해당 객체속성을 선택하여 연결하는 것을 가능하게 해준다.

4.1.4 인물과 이름 개체 간 역함수 관계 공리정의

동일한 서지적 신원을 가진 인물이 다양한 형식의 필명을 사용할 경우, 이 다양한 형식의 필명을 가진 인물이 동일인이라는 것을 표현하기

위한 접근점제어가 필요하다. 이때 owl:InverseFunctionalProperty라는 공리를 정의한다. owl:InverseFunctionalProperty 공리정의는 모든 y에 대해서 P(x, y)의 x가 1개로 규정되며 y에 대해서 1개의 x가 대응하는 역함수 관계에 있음을 나타낸다(〈그림 8〉 참조).

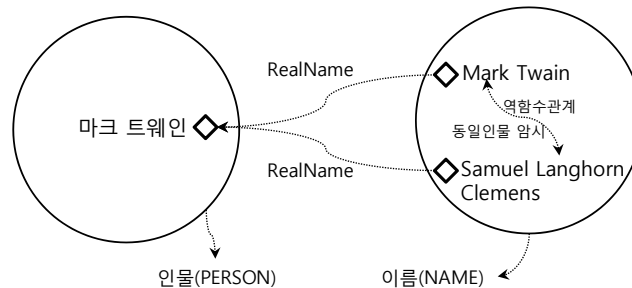
만약, 인물 마크 트웨인이 Mark Twain이라는 이름으로 Adventures of Huckleberry Finn이라는 저작을 창작하였고, Samuel Langhorn Clemens라는 이름으로 The Adventures of Tom Sawyer라는 저작을 창작하였다고 한다면, 온톨로지 기술 시 서로 다른 필명인 Mark Twain과 Samuel Langhorn Clemens가 동일한 서지적 신원임을 공리 정의하는 접근점 제어 과정이 필요하다.

```

<owl:ObjectProperty rdf:about="#RealName">
  <rdf:type rdf:resource="&owl:InverseFunctionalProperty"/>
  <rdfs:domain rdf:resource="#Name"/>
  <rdfs:range rdf:resource="#Person"/>
</owl:ObjectProperty>

```

〈그림 8〉 인물과 이름 개체 간 역함수 관계 공리정의



〈그림 9〉 동일인물 추론을 위한 역함수 공리정의

owl:InverseFunctionalProperty는 객체속성에 역함수 관계임을 정의하는 공리로서, 동일한 객체속성에 의해 주어로서 서로 다른 두개의 인스턴스가 목적어로서 동일한 인스턴스를 지시하면 이 서로 다른 주어는 동일한 인스턴스임을 추론한다.

〈그림 9〉에서 이름클래스 내 주어로서 서로 다른 필명인 Mark Twain과 Samuel Langhorn Clemens가 owl:InverseFunctionalProperty 공리정의가 된 객체속성 RealName에 의해 목적어로서 인물클래스 내 특정한 인스턴스를 동시에 지시하면 해당 주어들은 동일한 인물임을 묵시적으로 의미하게 된다.

이와 마찬가지로 모든 객체속성에 역함수 공리를 정의하여 상이한 형식의 이름을 동일 인물에 지시하게 되면 상이한 형식임에도 불구하고 동일한 서지적 신원임을 온톨로지로 기술할 수 있다.

4.1.5 인스턴스와 리터럴 간 데이터속성 공리정의

클래스, 객체속성, 데이터속성, 인스턴스 등 모든 자원은 특정 이름공간 내에서 전역적으로 유일하게 식별되어야한다. 인명접근점제어 온톨

로지를 구축할 때 역시 인물인스턴스와 이름인스턴스들도 광역적으로 개체가 식별되어야한다. 그러나 동명이인의 인물에 대해 이름으로 나타낼 경우, 문자의 표현형식이 동일하게 되어 온톨로지에서 유일성을 확보할 수 없다. 예를 들어 Mark Twain이라는 동일한 형식의 문자열을 갖는 동명이인의 이름이 URI가 'http://www.kanghm.org/accesscontrol#MarkTwain'으로 동일하게 정의되기 때문이다.

일반적으로 개체식별을 위해 사용되는 대표적인 문자열 체계중 하나는 식별기호이다. 식별기호는 '식별'이라는 기본적 기능을 통해 1) 도서, 정기간행물, 추상적 저작, 동영상 등 매체간 장벽, 2) 목록, 검색, 저작권 관리 등 기능적 장벽, 3) 단순한 것에서 복잡한 것으로의 메타데이터 수준 간 장벽, 4) 의미적 장벽, 5) 언어적 장벽 등을 뛰어 넘게 하는 5가지 유형의 상호운용성을 확보할 수 있다(Paskin 2006, 45).

이러한 식별기호의 특징을 바탕으로 본 연구에서는 데이터속성 공리정의를 사용하여 식별기호와 리터럴을 연결하여 인명접근점제어 온톨로지를 기술하였다. 즉, 데이터속성 정의를 통해 인스턴스 식별기호와 리터럴을 연결하여 식별기호를 문자열 형식으로 표현하는 구조를 적용하

였다. 인명접근점제어 온톨로지 내에서 개체 즉 인물 인스턴스와 이름 인스턴스 문자열을 광역적으로 유일성, 확장성, 의미성을 확보하는 범용적인 표기규칙으로서 두 단계 방식의 명명체계를 적용하였다.

첫 단계는 URI의 중복을 방지하기 위해 QNAME 접두사 뒤에 식별기호로서 각 클래스의 영문 첫 대문자에 일련의 숫자번호를 부여함으로써 인스턴스를 전역적으로 유일하게 식별할 수 있도록 하였다. 즉, 인물의 첫 번째 인스턴스는 P1, 인물의 두 번째 인스턴스는 P2로 정의되며 동일하게 이름의 첫 번째 인스턴스는 N1, 이름의 두 번째 인스턴스는 N2로 명명하는 방식이다. 인스턴스의 식별기호는 이름공간 URI 참조와 연결되는 로컬명으로서 'http://www.kanghm.org/accesscontrol#P1' 또는 'ac:P1'과 같이 유일하게 참조된다. 예를 들어, 한 명의 Mark Twain은 P1이라는 식별기호가 부여되고 URI는 'http://www.kanghm.org/accesscontrol#P1' 표현된다. 또 다른 Mark Twain에 대해서는 P2라는 식별기호가 부여되고 URI는 'http://www.kanghm.org/accesscontrol#P2'로 표현된다. 이를 QNAME 접두사를 사용하면 각각 ac:P1, ac:P2로 간단하게 표현할 수 있다. 이는 MARC 기반의 전거레코드에서 동명이인의 레코드 간 분리가 온톨로지 기반에서는 동일한 형식의 인스턴스에 서로 다른 URI 식별기호를 부여함으로써 쉽게 전역적인 분리가 가능하다는 것을 의미한다.

두 번째 단계는 RDF/OWL의 데이터속성을 사용하여 인스턴스와 리터럴을 연결하고 리터럴에 의미성이 있는 일반 언어형식의 문자열을 부여하였다. OWL 웹 온톨로지 마크업 언어는

리즈너가 자동으로 인식할 수 있을 뿐만 아니라 인간도 그 의미를 파악하기 쉬운 XML 기반의 언어이다. 그런데 위와 같은 방법은 인스턴스가 식별기호로 표현되면서 식별성과 유일성을 확보할 수 있으나 인스턴스의 의미성을 잃어버리게 된다. 즉, ac:P1, ac:P2, ac:N1, ac:N2가 각각 무엇을 의미하는지 인간은 알 수가 없다. 온톨로지 내에서 식별기호 기반으로 생성된 개체 간 복잡한 관계를 'P1 OfficialName N1'과 같이 화면에 그대로 디스플레이 할 경우 온톨로지 구축자 뿐만 아니라 이용자도 정보인식에 커다란 장애가 된다. 더욱이 인스턴스가 많아지고 온톨로지 내 공리정의가 증가할수록 이러한 의미파악 문제는 커진다.

이러한 문제점을 해결하기 위해 RDF/OWL의 데이터속성으로 'IsName'을 공리정의하고 인스턴스와 리터럴을 연결하였다. <그림 10>의 RDF/XML구문은 이름 인스턴스 ac:N1은 데이터속성 IsName을 통해 리터럴 'Twain, Mark'과 연결하고, 인물 인스턴스 ac:P1은 데이터속성 IsName을 통해 리터럴 '마크 트웨인'과 연결되고 있음을 의미한다.

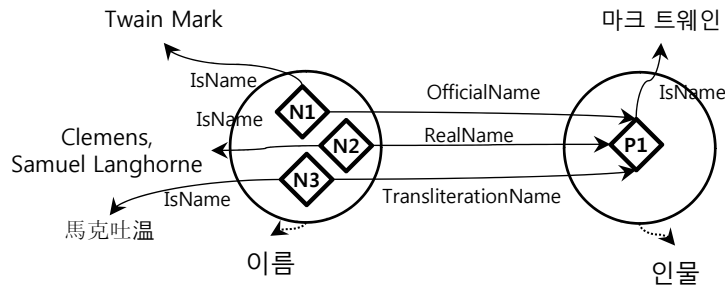
식별기호와 리터럴 간 연결구조를 개념적으로 도식화하면 <그림 11>과 같다. 식별기호로 명명된 N1, N2, N3 등은 이름클래스 내에서 유일성을 유지하면서 데이터속성 IsName을 통해 'Clemens, Samuel Langhorne', 'Twain Mark', '馬克吐溫' 등 다양한 형식의 문자열로 표현되고 있다. 인물 인스턴스 P1도 인물클래스 내에서 유일성을 유지하면서 데이터 속성 IsName을 통해 '마크 트웨인'이라는 문자열 표현이 가능하다.

이처럼 데이터속성 공리정의를 통한 식별기

```

<!-- http://www.kanghm.org/accesscontrol#N1 -->
<owl:Thing rdf:about="#N1">
  <rdf:type rdf:resource="#Name"/>
  <IsName rdf:datatype="&xsd:string">Twain, Mark</IsName>
</owl:Thing>
<!-- http://www.kanghm.org/accesscontrol#P1 -->
<owl:Thing rdf:about="#P1">
  <rdf:type rdf:resource="#Person"/>
  <IsName rdf:datatype="&xsd:string">마크 트웨인</IsName>
</owl:Thing>
    
```

〈그림 10〉 URI 식별기호와 리터럴 간 연결



〈그림 11〉 인스턴스와 리터럴 연결을 위한 데이터속성 공리정의

호와 리터럴 간 연결방식은 서지영역에서 동일한 표현형식을 갖고 있는 동명이인을 인스턴스의 중복 없이 동일한 이름공간을 갖고 있는 온톨로지에서 자유롭게 표현하고 식별할 수 있다.

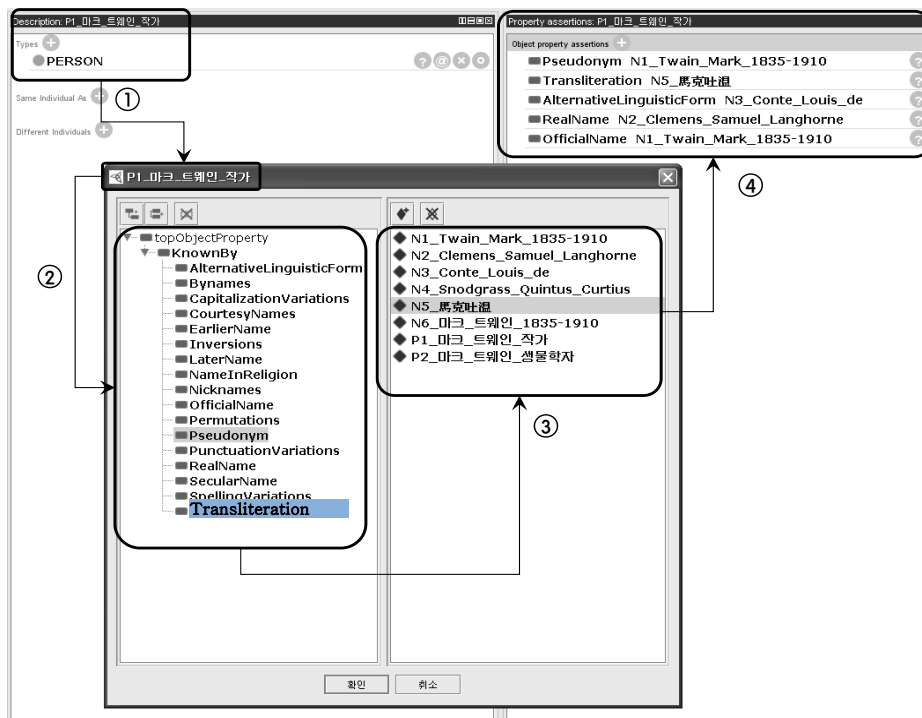
4.2 온톨로지 기술

지금까지 정의된 공리들을 바탕으로 온톨로지 저작도구인 Protégé를 이용해 마크 트웨인의 인명접근점제어 온톨로지를 기술하는 과정을 〈그림 12〉로 나타낸 것이다. 예를 들어 ① 인물클래스의 인스턴스인 P1(마크 트웨인)을 ② 17종의 객체속성 중 음역이형형식 어휘인 Transliteration을 이용하여 ③ 이름클래스의 인

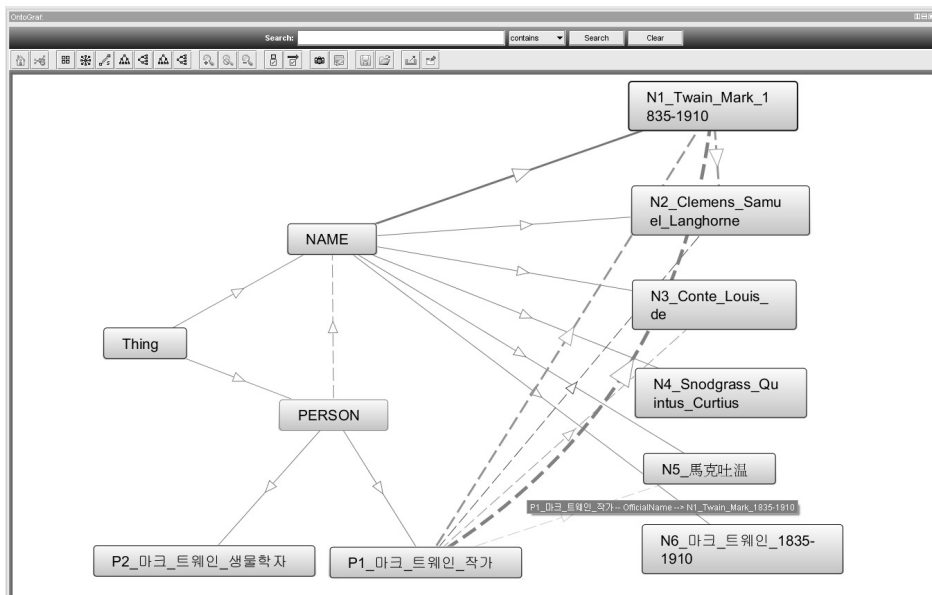
스턴스인 N5(馬克吐溫)와 연결하여 ④ 하나의 기술문장인 'P1(마크 트웨인) Transliteration N5(馬克吐溫)'이 완성된다. 이와 같이 특정인물 P1(주어)에 대해 17종의 객체속성(술어) 어휘를 사용하여 N1 ... Nn과 같은 다양한 이름형식(목적어)을 연결하는 과정을 반복하여 온톨로지를 기술한다.

이러한 과정을 반복해서 생성된 SPO트리플들이 쌓여 특정작가 마크트웨인에 대한 인명접근점제어 온톨로지가 기술된다.

〈그림 13〉은 위 과정의 결과로 생성된 온톨로지를 디스플레이하기 위해 Protégé에서 제공되는 플러그인인 OntoGraf를 이용해 나타낸 것이다.



〈그림 12〉 SPO 술어기반에 의한 인스턴스 접근점 간 연결



〈그림 13〉 OntoGraf 플러그인을 이용한 인명접근점제어 온톨로지(마크 트웨인) 표현

5. 결 론

지금까지 전거표목 및 이형 표목, 제어형 및 비제어형접근점의 개념을 배제한 대등형접근점의 개념을 적용한 웹 온톨로지 기반의 인명접근점제어 온톨로지를 기술하였다. 이 과정에서 인물과 이름 간의 다양한 관계와 특질을 표현하기 위해 RDF/OWL의 공리를 정의하였는데 인물·이름클래스 간 서로 소 공리정의, 객체속성의 정의역·치역 공리 정의, 인물·이름클래스의 객체속성값 출현횟수 제약 공리정의, 인물과 이름 개체 간 역함수 관계 공리정의, 인스턴스와 리터럴 간 데이터속성 공리정의 등을 적용하였다. 정의된 공리를 바탕으로 온톨로지 저작도구인 Protégé를 사용하여 미국 작가 마크 트웨인을 대상으로 온톨로지를 구축하는 기술하는 과정과 그 결과를 디스플레이하였다.

본 연구를 통해 온톨로지 기반의 공리정의를

통해 기존의 접근점 제어기능을 수용하고 보다 강화된 표현력을 지닌 온톨로지를 기술할 수 있었다. 또한 온톨로지 환경에서 전거와 표목의 개념과 더불어 제어형접근점과 비제어형접근점 개념을 배제하고 언어, 문자, 표현형식 등에 상관없이 모든 형식을 대등한 접근점으로 적용할 수 있는 소위 '대등형접근점'이라는 개념을 제시할 수 있었다.

또한 본 연구에서 구축한 마크 트웨인의 접근점제어 온톨로지는 동일한 이름공간 내에서 다른 인물들의 모든 표현형식의 접근점을 무한히 추가하여 기술할 수 있다. 이는 유일성, 상호운용성, 공유성, 재사용성, 통합과 확장의 용이성 및 유지관리의 편의성이라는 온톨로지의 기본적인 특성으로 볼 때, 인명접근점제어 온톨로지는 기존 MARC 기반의 전거레코드가 할 수 없었던 접근점 제어기능의 국제적인 공유와 통합을 가능하게 중요한 메커니즘으로 작용할 것이다.

참 고 문 헌

- [1] 김성혁, 박영택, 추윤미. 2009. 『온톨로지 개발자를 위한 시맨틱 웹: W3C RDF·RDFS·OWL 기반 온톨로지 모델링』. 서울: 사이텍미디어.
- [2] 김태수. 2008. 『목록의 이해』. 서울: 한국도서관협회.
- [3] 김태수, 심경. 2005. 『RDF 입문서』. 서울: 문헌정보처리연구회.
- [4] 황석영, 양해술. 2008. 『시맨틱 웹을 위한 RDF/OWL 입문』. 서울: 홍릉과학출판사.
- [5] Antoniou, Grigoris, & van Harmelen, Frank. 2008. *A Semantic Web Primer*. 2nd ed. Cambridge, MA: The MIT Press.
- [6] Bock, Conrad. 2009. *OWL 2 Web Ontology Language Structural Specification and Functional-Style Syntax*. [online]. [cited 2012.4.16].
 <<http://www.w3.org/TR/2009/REC-owl2-syntax-20091027/>>.

- [7] Gruber, Thomas R. 1993. "Toward principles for the design of ontologies used for knowledge sharing." *International Journal Human-Computer Studies*, 43(5-6): 907-928.
- [8] Joint Steering Committee for Development of RDA. 2008. RDA Full Draft. [online]. [cited 2012.4.13]. <<http://www.rdaonline.org/constituencyreview>>.
- [9] Kanzaki, Masahide. 2005. *Semantic Web No Tame No RDF/OWL Nyumon*. Tokyo: UNI Agency.
- [10] Mungall, Haase. 2009. *A Practical Guide To Building OWL Ontologies Using Protege 4 and CO-ODE Tools : Edition 1.2*. Manchester: The University Of Manchester.
- [11] Paskin, Norman. 2006. "Identifier interoperability." *D-Lib Magazine*, 12(4). [online]. [cited 2012.4.12]. <<http://www.dlib.org/dlib/april06/paskin/04paskin.html>>.

• 국문 참고자료의 영어 표기

(English translation / romanization of references originally written in Korean)

- [1] Kim, Sung-Hyuck, Park, Yung-Tack, & Choo, Yoon-Mi. 2009. *Semantic Web for the Ontology Engineering Developer: Ontology Modeling Based W3C RDF·RDFS·OWL*. Seoul: Sytechmedia.
- [2] Kim, Tae-Soo. 2008. *The Understanding of Catalog*. Seoul: Korean Library Association.
- [3] Kim, Tae-Soo & Shim, Kyung. 2005. *The Primer of RDF*. Seoul: The Studies of Library and Information Processing.
- [4] Hwang, Suck-Young & Yang, Hae-Sool. 2008. *The Introducing of RDF/OWL for Semantic Web*. Seoul: HongNeung Publication.