

클래스 속성 중심 메타데이터 레지스트리 시스템 구축에 관한 연구*

- 한국정보통신기술협회 메타데이터 레지스트리 시스템을 중심으로 -

A Study on Building a Metadata Registry System Centered on Class and Property Elements: Focused on the MDR of the Korea Information and Communication Technology Association

박진호 (Jin Ho Park)**

목 차

- | | |
|------------------|-------------------|
| 1. 서론 | 4. 선행연구 |
| 2. 연구 대상과 범위 | 5. 단일 요소중심 MDR 설계 |
| 3. ISO/IEC 11179 | 6. 결론 |

초 록

이 연구는 ISO/IEC 11179 표준을 준수하는 메타데이터 레지스트리(MDR) 시스템의 재설계와 구현에 관한 것이다. 재설계의 핵심은 MDR 시스템을 유지하면서 이용자들에게 실제로 필요한 핵심이라 할 수 있는 클래스(class), 속성(property), 값 영역(value domain)의 코드값을 중심으로 시스템을 재편하는 것에 있다. 이를 구현하기 위해서는 표준을 준수하는 운영 시스템이 필요하여, 그 대상을 한국정보통신기술협회의 MDR 시스템으로 하였다. 그 결과 재설계한 시스템은 표준에서 제시하고 있는 개념영역, 메타데이터 요소 개념, 메타데이터 요소, 값영역에 대한 이해 없이도 쉽게 특정 메타데이터의 검색과 활용이 가능함을 확인하였다.

ABSTRACT

This study proposed a method to redesign the search system by extracting the class, property, and code values of value domain from the key elements of the MDR system that conforms to the ISO/IEC 11179 standard. To implement this, an operating system conforming to the standard is required. Therefore, this study targeted the MDR system of the Korea Information and Communication Technology Association. As a result, it was confirmed that the redesigned MDR system can easily search and utilize specific metadata without understanding the conceptual domain, metadata element concept, metadata element, and value domain proposed in the standard.

키워드: 메타데이터 레지스트리, ISO/IEC 11179, 클라우드 상호운용성, 메타데이터 상호운용성, 메타데이터 레지스트리 재설계
MDR, Metadata Registry, ISO/IEC 11179, Cloud Interoperability, Metadata Interoperability, Redesigned Metadata Registry

* 본 연구는 2019년 한국정보통신기술협회의 '클라우드 데이터 상호운용성 지원시스템 기능 개선' 사업의 결과 중 일부를 수정·보완한 것임.

** 주식회사 리스트 사업개발본부장(jinhopark.lis@gmail.com / ISNI 0000 0004 7641 0372)

논문접수일자: 2020년 7월 23일 최초심사일자: 2020년 7월 31일 게재확정일자: 2020년 8월 12일
한국문헌정보학회지, 54(3): 263-283, 2020. <http://dx.doi.org/10.4275/KSLIS.2020.54.3.263>

1. 서론

메타데이터의 상호운용성(Metadata Interoperability) 확보 필요성은 정보화 환경의 변화와 무관한 지속적인 주제 중 하나이다. 지식정보자원의 형태가 무엇이든 이를 관리하고 서비스하기 위해서는 해당 자원에 적합한 메타데이터의 선택은 필수적이라고 할 수 있다. 이는 지식정보자원의 교환과 공동활용을 위한 상호운용성 문제와 직결된다. 상호운용성 문제는 지식정보자원의 공동활용은 물론 기기종 하드웨어, 운영체제, 응용 프로그램, 데이터베이스까지 정보화 환경 전반에 걸친 공통 문제이다.

최근 정보통신 자원(IT Resource)을 인터넷을 통한 온디맨드(on-demand)로 제공하고 사용한 만큼 지불하는 방식의 클라우드 컴퓨팅(Cloud Computing)이 빠르게 확산하고 있다. 이 과정에서 다양한 클라우드 간의 상호운용성은 물론이고, 클라우드 내의 응용시스템, 데이터 등의 상호운용성 확보의 필요성이 증가하고 있다. 초기 상호운용성 문제는 클라우드 시장 확대에 따른 것으로 이용자가 특정 업체의 클라우드 서비스에 종속되는 현상을 막기 위해 출발했다고 볼 수 있다. 즉, 클라우드 서비스 제공업체가 제공하는 가상머신이미지(virtual machine image), 데이터 추출방식, 파일 형식, API, 프로토콜이 상이하여 클라우드 간의 이전이 불가능한 상황이 원인이다.

해외의 경우 산업계 표준화 단체인 DMTF(Distributed Management Task Force)를 중심으로 문제해결을 위한 노력이 이어지고 있다. DMTF는 가상화 플랫폼 간 가상머신의 배포와 이동성을 보장하기 위한 표준형식을 배포하고

있다. 대표적인 표준이 OVF(Open Virtualization Format)이다.

반면 국내의 경우 민간 산업계의 표준화 노력이 활성화되기 전에 국가 주도의 상호운용성 해결을 위한 노력을 시작했다고 볼 수 있다. '클라우드컴퓨팅 발전 및 이용자 보호에 관한 법률 [시행 2017. 7. 26.] [법률 제14839호, 2017. 7. 26., 타법개정]'이 그것으로 동법 22조에 상호운용성 확보를 위해 클라우드컴퓨팅서비스 제공자의 협력 체계 구축 권고를 과학기술정보통신부장관이 수행할 수 있도록 명문화하였다. 이 법은 클라우드 초기 시장부터 상호운용성 논의를 시작하여 서비스 연동의 어려움, 소수 상위 기업의 독점적 구조 해소를 위한 것이다. 한국정보통신기술협회(이하, TTA)는 동법을 근거로 상호운용성 문제해결을 위해 '상호운용성 협의체'를 발족하고 클라우드 컴퓨팅 기술과 관련된 다양한 상호운용성 문제를 발굴하고 해결하기 위한 과제를 수행하고 있다. 데이터의 의미적 상호운용성 확보는 이 과제 중의 하나로 클라우드 애플리케이션의 메타데이터 표준화를 위한 메타데이터 레지스트리(Metadata Registry, MDR) 시스템 구축과 운영을 목표로 한다.

MDR을 대표하는 표준은 ISO/IEC 11179 Metadata Registry로 데이터 공유와 교환을 근본적으로 해결하기 위한 목적을 갖고 있다. 이 표준은 MDR 구축의 기본지침으로 호주의 METeOR(Metadata Online Registry), 국내의 TTA MDR 등이 활용하고 있다. 표준은 총 7개의 파트로 구성되어 있는데 MDR에 대한 개념적인 이해부터 등록지침과 실제 운영조직의 역할까지 상세히 기술하고 있다. 실제 운영조직과 시스템을 갖추고자하는 경우에는 유용

할 수 있으나 동 표준이 제시하는 개념모델의 학습과 이용을 위한 이해에는 상당한 어려움이 따른다. 특히 기본 개념이라고 할 수 있는 개념 영역(conceptual domain), 데이터 요소 개념(data element concept), 데이터 요소(data element), 값영역(value domain)에 대한 이해가 없을 경우 실제 시스템 이용에 어려움이 따를 수 있다. 반면 최근의 표준 용어집 서비스는 Schema.org (<http://schema.org>) 서비스처럼 클래스(class), 속성(property), 유형의 간단하고 상호간의 연결을 중심으로 하는 경우가 많다.

본 연구에서는 ISO/IEC 11179 표준을 준수하는 MDR 시스템을 유지하면서 이용자들에게 실제로 필요한 핵심이라 할 수 있는 클래스, 속성, 값 영역을 중심으로 검색 시스템을 재설계하여 방안을 제시하였다. 클래스와 속성은 ISO/IEC 11179 표준에서 데이터 요소 개념의 하위에 속하는 요소들이며, 값 영역은 실제 클래스와 속성으로 구성된 하나의 요소를 어떻게 표현할지를 다룬다. 즉, MDR에서 메타데이터 요소를 찾거나 활용하고자 하는 이용자 입장에서는 실제 응용 가능한 핵심적인 요소라 할 수 있다. 본 연구는 이 핵심 요소를 MDR에서 연계하여 검색시스템을 구성함으로써 이용과 응용이 쉬운 MDR을 구성하여 제시하였다.

2. 연구 대상과 범위

본 연구 목적 달성을 위해서는 ISO/IEC 11179 표준을 준수하는 MDR 정보시스템이 필요하여, TTA에서 운영하고 있는 MDR 시스템을 대상으로 하였다. 물론 세계적으로 가장 잘 알려진

시스템인 호주의 METeOR가 존재하기는 하지만, 이 시스템의 경우 온라인으로 메타데이터 검색과 활용은 가능하지만 본 연구의 목적 달성을 위해 필요한 원천 데이터베이스에 대한 접근과 자유로운 활용이 불가능하다. 반면 TTA MDR은 ISO/IEC 11179 표준을 준수하는 오픈 소스 Aristotle MDR (<https://www.aristotlemetadata.com/>)을 기반으로 하고 있어 원천 데이터베이스에 대한 구조의 이해와 활용이 용이하다. 부가적으로 이미 메타데이터가 입력된 상태로 변환 구조 및 시스템 개발 후의 모습까지 직접 확인할 수 있는 장점이 있다.

TTA MDR이 채용하고 있는 시스템은 ISO 표준을 그대로 준수하고 있어 그 구조가 방대하다. 때문에 본 연구에서는 전체 시스템 변경을 다루지 않고 ISO/IEC 11179에서 핵심적인 메타데이터 입력 개념 중 메타데이터 요소 개념의 하위 요소인 클래스, 속성 추출에 중점을 두고 해당 요소를 표현하기 위한 값 영역까지를 포괄하여 구성하는 것으로 범위를 한정하였다. 즉 간결한 메타데이터 표현으로 검색결과에 대한 이해의 어려움을 감소시키기 위함이다. 여기서 간결한 표현은 ISO 11179를 준수할 경우 '클래스명, 속성명, 값표현방식'의 조합형 표현방식을 '클래스명', '속성명'으로 간결하게 보여주는 것을 의미한다. 이 후 논문에서는 이를 '단일요소'라는 명칭을 사용하여 제시하였다.

3. ISO/IEC 11179

본 연구는 ISO/IEC 11179 표준을 근간으로 한다. 이 표준은 ISO/IEC JTC1/SC32에서 제

정 및 운영하고 있다. SC32는 데이터 관리와 교환(Data management and interchange)에 관한 표준을 다루는 곳이다. ISO/IEC 11179 작업반 2(WG 2)가 책임을 갖고 운영하며, 이 작업반은 메타데이터를 주 대상으로 한다. ISO 11179 표준은 총 7개의 부분으로 구성되어 있으며 <표 1>과 같이 정리된다.

ISO 표준은 지속적인 갱신 작업이 이루어지고 필요에 따라 새로운 보충 표준이나 신규 표준이 포함된다. ISO/IEC 11179의 경우 'ISO/IEC 11179-3:2013/AMD 1:2020'이 Part3의 수정안으로 별도로 공표되었다. 또 'ISO/IEC TS 11179-30:2019'은 기술문서로 11179-3에 적합하지 않은 상황에서 사용할 수 있는 별도의 사양을 제시하고 있다. Part 7의 경우는 최근에 추가한 표준으로 데이터 셋(data set) 등록을 위한 표준이다.

ISO/IEC 11179 표준 중 본 연구와 가장 관련이 깊은 부분은 Part3로 실제 시스템 구현에 필요한 모든 사항을 상세하게 담고 있다. 특히 시스

템 구현이 필요한 모든 요소 간의 관계를 UML 형식으로 제공하고 있어 데이터베이스 구성과 개발에 직접 활용 가능하다(ISO 2013). TTA MDR이 사용하고 있는 Aristotle MDR 역시 Part3가 제시하고 있는 요소를 중심으로 만들어진 시스템이다.

4. 선행연구

본 연구 관련 선행연구는 메타데이터의 상호 운용성과 클라우드 상호운용성 두 가지로 볼 수 있다. 클라우드 상호운용성의 경우 초기에는 클라우드 서비스 간의 상호운용성과 표준화의 필요성에 초점을 맞춘 연구들이 중심을 이루고 있었다. Bernstein et al.(2009), Lewis (2013)의 연구가 대표적으로 클라우드 서비스 간의 프로토콜 표준화와 가상화 기술 표준의 필요성과 중요성을 지적한 바 있다. 이 후 클라우드 서비스가 확대되면서 특정 분야에 국한한 상

<표 1> ISO/IEC 11179 표준의 구성과 주요내용

표준명	최근발행년도	주요내용
Part 1: Framework	2015	ISO/IEC 11179의 각 부분들을 연관하여 이해할 수 있도록 메타데이터와 메타데이터 레지스트리에 대한 개념적 이해를 제공
Part 2: Classification	2019	분류 체계 등록 방법을 설명
Part 3: Registry metamodel and basic attributes	2013	개념적 데이터 모델 형식으로 MDR의 구조 제시
Part 4: Formulation of data definitions	2004	데이터와 메타데이터에 대한 정의를 구성하기 위한 요구 사항과 권장 사항 안내
Part 5: Naming principles	2015	Part 3에서 정의한 주요 항목의 명명 지침 제공
Part 6: Registration	2015	메타데이터 등록 절차 안내
Part 7: Metamodel for data set registration	2019	Part 3에 명시한 MDR 확장 사양으로, 데이터셋 등록을 위한 방법을 제시함

출처: ISO n.d.

호운용성과 전체 체계 유지를 위한 규범의 필요성 연구가 지속적으로 이어지고 있다. Mahmud, Koch and Buyya(2018)의 경우 보건의료분야에서 IoT 솔루션들 간의 상호운영성 확보를 위한 Cloud-Fog 서비스 통합 방안을 제시한 바 있다. Song(2017)의 경우는 상호운영성 확보에 영향을 주는 유럽의 법률을 조사하여 이를 저해하거나 충돌이 일어날 수 있는 부분을 조사분석하여 제시하고 지적재산권, 표준화를 위한 노력의 필요성을 언급하였다. Yongsiriwit, Sellami and Gaaloul(2016)은 이기종 클라우드 간 상호운용을 위한 의미적 프레임워크(semantic framework)를 제안하여 서로 다른 표준을 준수하는 클라우드 간의 리소스를 이해하고 활용할 수 있는 방안을 제시하기도 하였다.

메타데이터의 상호운용성의 경우 문헌정보학분야를 중심으로 활발한 연구가 지속되어 왔다. 도서관 MARC의 상호운영성 확보를 위한 연구(심경 2003)부터 이기종 데이터베이스의 통합운영을 위한 연구(고영만, 배경재 2011), 마스터 데이터를 활용한 메타데이터 간 상호운영성 확보에 관한 연구(이승민 2013)와 같이 전통적인 데이터베이스를 기반으로 한 연구가 대표적이라 할 수 있다. 보다 근본적인 해결을 위한 ISO/IEC 11179를 소개하고 확장하기 위한 연구(정혜진, 백두권, 정동원 2009; 남영광, 서태설, 황상원 2005)로 이어지던 연구는 MDR 자체를 시맨틱 웹 기반으로 확장하여 활용(오삼균 2004; 2005)하기 위한 연구까지 이어졌다.

지금까지의 연구는 클라우드와 메타데이터 상호운용성 부분과 메타데이터 상호운용성 부분이 큰 연관없이 각자의 분야에서 지속성을 갖고 이어지고 있는 양상을 보인다고 할 수 있다. 반면

본 연구는 클라우드 애플리케이션 상호운용성, 특히 메타데이터의 상호운용성이라는 구체적인 목표를 갖는 국제표준 기반 시스템을 기반으로 하고 있다는 점과 국제표준을 확장하거나 시스템을 응용하기보다 이용자 측면에서 간결하게 메타데이터 탐색과 응용이 가능한 시스템을 설계한다는 점에서 의의가 있다 할 수 있다.

5. 단일 요소중심 MDR 설계

5.1 TTA MDR

TTA MDR은 2017년 시스템 구축과 메타데이터 입력을 시작하였다. 이 후 메타데이터를 보강하여 2020년 ‘http://mdr.tta.or.kr’라는 도메인으로 서비스를 시작했다. 이 시스템은 오픈소스 MDR인 Aristotle MDR을 활용하고 있으며 상세 운영 현황은 <표 2>와 같다.

<표 2> TTA MDR 설치 현황

구분	내용
개발자	Aristotle Metadata Community
버전	v2.1(2018-07-10)
개발환경	Python, Django
준수표준	ISO/IEC 11179
라이선스	BSD
운영환경	AWS m4.large(vCPU 2, Memory 8GB, 대역폭 450Mbps)
운영체제	Ubuntu 16

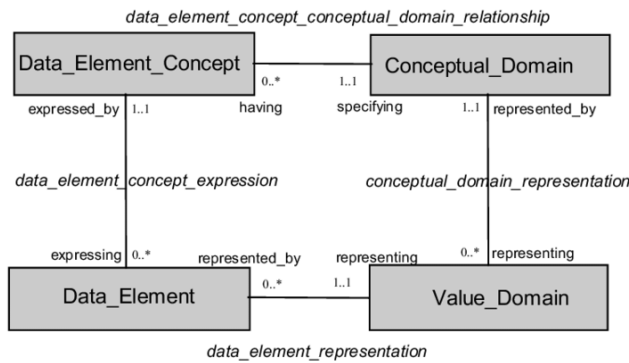
ISO/IEC 11179 표준을 준수하는 시스템을 구축하기 위해서는 특히 Part3에서 기술하고 있는 모델 이해가 필수적이다. Part3는 레지스

트리 메타모델과 기본 속성에 대해 기술하고 있으며, 시스템 구축 시 가장 핵심으로 활용해야 한다. Aristotle MDR의 경우도 이 표준의 요건을 만족시키고 있다. 이 표준이 제시하고 있는 메타 모델은 <그림 1>과 같다.

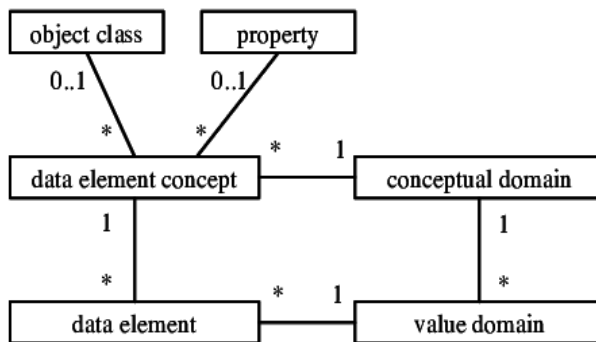
개념 영역은 값 영역을 이루는 집합들을 확장한 개념이다. 데이터 요소 개념은 데이터 요소형태로 표현되는 개념으로 개념 영역의 추상적인 개념을 구성하는 단위 요소개념이다. 데이터 요소는 이 개념을 구체적인 표현으로 나타낼 수 있는 단위이며, 값 영역은 데이터 요소에 대해 허용

가능한 값들의 집합이다. 고영만(2005)은 각 구성요소의 이해를 위한 예로 국가라는 개념 영역, 국가 이름이라는 데이터 요소 개념, 국제표준 국가명코드라는 데이터 요소, ISO 3166-3-Alpha Code라는 값 영역의 예를 제시한 바 있다. 이 표준을 준수하는 MDR은 위 4가지 구성요소에 대한 관리와 서비스 체계를 갖추고 있어야 한다.

사실 <그림 1>의 데이터 요소 개념은 객체 클래스와 속성을 포함하고 있는데 이를 반영하여 개체(entity) 관점에서 모델을 간결하게 표현하면 <그림 2>와 같다.



<그림 1> ISO/IEC 11179 Part3 메타모델
출처: Stausberg et al. 2009



<그림 2> ISO/IEC 11179 기본 개체(entity) 모형
(출처: Davies et al. 2008)

즉 데이터 요소 개념을 표현하기 위해서는 객체 클래스와 속성을 포함하여 표현하고 기술해야함을 알 수 있다.

실제 이 모델을 기반으로 한 TTA MDR의 입

력시스템은 관리기능에서 확인할 수 있는데 <그림 3>, <그림 4>와 같다. <그림 3>에서 보는 것처럼 TTA MDR은 ISO/IEC 11179 기본 개체모형에서 제시하는 모든 요소의 입력이 가능함을

Aristotle Metadata Registry

Create new	Help	Basic description	
Conceptual Domain	?	개념적 영역은 유효한 인스턴스의 의미나 설명을 표현하는 영역임.	+ Create
Data Element	?	재사용 가능한 메타데이터 구성 요소를 사용해서 일련의 객체 측정 가능한 특성을 MDR 시스템에 기록하는 방법을 정의	+ Create
Data Element Concept	?	관찰 대상, 아이디어 또는 이벤트의 측정 가능한 속성을 정의하기 위한 객체 클래스와 속성의 유효한 쌍	+ Create
Data Element Derivation	?	하나 이상의 입력 데이터 요소(data element)에 기반하여 규칙을 적용하여 출력 데이터 요소가 파생되는 경우.	+ Create
Data Type	?	데이터 유형(data type)은 해당 값의 속성과 해당 값을 연산하는데 필요한 정보로 구분할 수 있는 고유값을 의미함.	+ Create
Object Class	?	모든 측정 가능한 속성을 공유하는 실제 객체, 아이디어, 이벤트 집합을 분류하기 위한 정의.	+ Create
Property	?	객체 클래스에 의해서 정된 집합의 모든 구성요소에 공통된 속성.	+ Create
Unit Of Measure	?	값 영역(value domain)에서 값이 측정되는 단위.	+ Create
Value Domain	?	값이 코딩되는 목록 혹은 가능한 값에 대한 설명을 사용해서 특정 속성을 측정 값으로 기록하는 방법에 대한 정의.	+ Create

<그림 3> TTA MDR 관리시스템의 입력 항목
출처: TTA MDR(<http://mdr.tta.or.kr>) 관리시스템

Create conceptual domain - Step 2 of 2 - Select or create

일지하는 항목이 발견되지 않았으므로 이 마법사는 새 항목을 만듭니다.

Definition Names & References Components Tips and help

이 필드는 항목을 식별하고 정의하는 데 도움이 됩니다. 이름과 정의가 필요하지만 버전을 추가하는 것은 선택 사항입니다.

Name: Metadata Version:

The primary name used for human identification purposes.

Workgroup: The workgroup responsible for maintaining this metadata item.

Definition: Representation of a concept by a descriptive statement which serves to differentiate it from related concepts. (3.2.39)

Rich text editor with toolbar (bold, italic, link, etc.)

< Back 저장

<그림 4> TTA MDR 관리시스템의 입력 상세 화면
출처: TTA MDR(<http://mdr.tta.or.kr>) 관리시스템

알 수 있다.

〈그림 4〉는 개념적 영역의 입력화면으로 나머지 요소의 입력화면에서도 공통적으로 확인 가능하다. 표준에 따라 각 요소별로 입력 요소는 유사하다. 〈그림 4〉에서처럼 입력 요소 구분은 정의, 이름과 참고사항, 구성요소(components)로 나누어진다. 정의에서는 요소 이름, 버전, 정의를 입력해야한다. 이름과 참고사항에는 참고정보,

원 URI(origin URI), 부연설명(commnets), 요소 제출기관과 책임기관 이름을 입력하도록 구성되어 있다. 구성요소는 선택사항이다.

이렇게 입력한 결과는 메타데이터 요소 검색 결과 화면에서 확인할 수 있으며, 메타모델을 기반으로 제공하고 있음을 알 수 있는데 이는 〈그림 5〉, 〈그림 6〉과 같다.

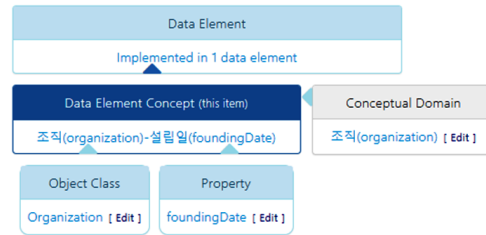
〈그림 5〉는 메타데이터 요소 개념 결과화면

조직(organization)-설립일(foundingDate) (Data Element Concept)

Definition

특정 조직이 설립된 날짜를 표현할 때 사용하기 위한 데이터 요소 개념.

Components



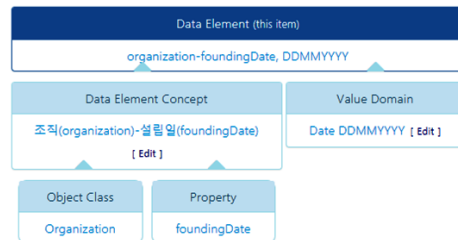
〈그림 5〉 TTA MDR 검색결과 화면 중 메타데이터 요소 개념 결과화면
출처: TTA MDR(<http://mdr.tta.or.kr>)

organization-foundingDate, DDMMYYYY (Data Element)

Definition

특정 조직이 설립된 날짜를 표현할 때 사용 데이터 요소로 ISO 8601에 따라 간략하게 DDMMYY(날짜 월 연도)순으로 표기함.

Components



〈그림 6〉 TTA MDR 검색결과 화면 중 메타데이터 요소 결과화면
출처: TTA MDR(<http://mdr.tta.or.kr>)

으로 이용자는 개념영역, 데이터 요소개념, 데이터 요소, 클래스, 속성에 대한 관계를 이해해야만 활용 가능하다.

〈그림 6〉은 메타데이터 요소 결과화면으로 〈그림 5〉처럼 개념영역, 데이터 요소개념, 데이터 요소, 클래스, 속성에 대한 관계를 이해해야만 활용할 수 있다. 즉 〈그림 5〉, 〈그림 6〉처럼 이용자는 메타데이터 요소를 검색하면 ISO 11179의 개념모델을 반영한 결과를 확인할 수 있다. 실제 응용에 필요한 하나의 메타데이터 요소가 아니라 관련된 모든 개념의 이해가 있어야 활용 가능하다고 할 수 있다.

〈그림 3〉, 〈그림 4〉, 〈그림 5〉, 〈그림 6〉에서 확인한 것처럼 TTA MDR은 표준을 엄격하게 준수하고 있다. 이 서비스 시스템의 관점을 이용자로 돌려보면 이용자의 경우 표준에 대한 이

해, 특히 MDR의 메타모델에 이해가 없으면 어떤 요소를 활용해야하는지에 대한 어려움이 발생한다. 요소(element)를 찾고자 하는 경우를 상정한다면 데이터 요소, 데이터 요소개념에서 표현하고 있는 것들이 실제 이용자가 찾는 개념을 정확하게 반영한다고 볼 수 있지만 실제 이 요소 값들은 클래스와 속성의 조합으로 이루어져 있음을 알 수 있다. 특정 하나의 메타데이터 요소를 찾고자하는 경우에 표준에서 제시한 메타 모델에 대한 이해가 없으면 활용이 어려울 수 있음을 확인할 수 있다.

또 MDR의 특징 중 하나는 값 영역에 입력해야하는 허용 가능한 값(value)들을 직접 입력할 수 있는 구조를 갖고 있다는 점이다. 〈그림 7〉은 대한민국 전화번호 표기의 지역 구분 값에 대한 값 영역 입력결과 화면이다.

지역번호 (Value Domain)

Definition
대한민국 전화번호 표기 시에 참조할 수 있는 값으로, 각 지역을 구분하는 지역번호 값 집합임.

Representation

Data Type	String
Maximum character length	3

Permissible Values

Permissible Values [edit]	Value	Meaning	Start Date	End Date
	02	서울	-	-
	031	경기도	-	-
	032	인천광역시	-	-
	033	강원도	-	-
	041	충청남도	-	-
	042	대전광역시	-	-
	043	충청북도	-	-
	044	세종특별자치시	-	-
	051	부산광역시	-	-
	052	울산광역시	-	-
	053	대구광역시	-	-
	054	경상북도	-	-
	055	경상남도	-	-
	061	전라남도	-	-
	062	광주광역시	-	-
	063	전라북도	-	-
	064	제주특별자치도	-	-

〈그림 7〉 TTA MDR 값 영역 검색 결과 화면 중 일부
출처: TTA MDR(<http://mdr.tta.or.kr>)

표준 메타데이터를 검색하고 직접 활용하고자하는 입장에서 본다면 클래스, 속성 검색뿐만 아니라 <그림 7>처럼 직접 데이터베이스에 입력하여 활용할 수 있는 코드 값들이 갖는 의미가 더 크다고 할 수 있다. 정리하면 단일 요소 기반으로 MDR을 재편하기 위해서는 클래스, 속성, 값 영역이 갖고 있는 코드형의 데이터를 중심으로 새로운 데이터베이스 구성이 필요하다고 할 수 있다.

5.2 TTA MDR 데이터베이스

TTA MDR을 유지하면서 클래스, 속성, 코드 값 중심으로 새로운 시스템을 구축하기 위해서는 TTA MDR의 데이터베이스 구조를 파악하는 것이 필요하다. 이 데이터베이스는 총 57개의 테이블로 구성되어 있다. 전체 테이블 목록은 [부록 1]에 첨부하였으며, 이 중 개념모델과 관련된 요소 등록 테이블을 정리하면 <표 3>과 같다.

<표 3>에서 정리한 내용은 'ISO/IEC 11179-3: Registry Metamodel and Basic Attributes'에서 제시하고 있는 개념 요소, 데이터 요소 개념,

데이터 요소, 값 영역을 표현하기 위한 요소를 포함한 테이블이다. 테이블 구분번호로 개념영역의 각 요소들을 구분하면 개념 요소는 1번, 데이터 요소 개념은 3번, 5번, 7번, 데이터 요소는 2번, 값 영역은 4번, 6번, 8번, 9번과 관련된다. 즉, 요소중심의 새로운 데이터베이스를 구성하기 위해서는 TTA MDR의 전체 테이블 중 위 9개 테이블에 대한 이해가 필요하다. 특히, 객체, 속성, 값 영역을 직접적으로 입력받는 5번, 7번, 8번, 9번 테이블은 직접 연동하여 활용해야한다. 이 외 실제 시스템 운영을 위한 조직관리, 인적자원 관리, 권한관리 등의 요소들은 제외하였다. 비록 본 고에서는 제외하였지만 이런 요소들은 표준에도 기술되어 있으며, 실제 운영을 위해서는 반드시 필요한 요소임을 밝혀둔다.

5.3 신규 데이터베이스 설계

본 연구에서 설계 및 구축하고자하는 새로운 MDR은 기존의 TTA MDR을 새로운 시스템으로 개편하기 위한 것은 아니다. TTA MDR은 국제표준을 준수하고 있는 의미있는 시스템

<표 3> TTA MDR 데이터베이스 테이블 중 개념모델 관련 테이블 목록

구분	테이블명	내용
1	aristotle_mdr_conceptualdomain	개념적 영역(Conceptual domain)에 해당하는 메타데이터 테이블
2	aristotle_mdr_dataelement	데이터 요소(Dataelement)에 해당하는 메타데이터 테이블
3	aristotle_mdr_dataelementconcept	데이터 요소(Dataelement concept)에 해당하는 메타데이터 테이블
4	aristotle_mdr_datatype	데이터 유형(data type)에 해당하는 메타데이터 테이블
5	aristotle_mdr_objectclass	객체 클래스(object class)에 해당하는 정보가 있는 테이블
6	aristotle_mdr_permissiblevalue	승인된 값에 대한 정보가 있는 테이블
7	aristotle_mdr_property	속성(property)에 해당하는 정보가 있는 테이블
8	aristotle_mdr_valuedomain	값 영역(value domain)에 대한 정보가 있는 테이블
9	aristotle_mdr_valuemeaning	값(value)에 대한 정보가 있는 테이블

으로 그 자체로 서비스를 유지해야한다. 즉 신규 시스템이 활용할 원천데이터를 지속적으로 생산해야한다. 즉 신규시스템은 기능적인 면에서 보면 TTA MDR로부터 원천데이터를 반입하고, 반입한 클래스, 속성, 코드값을 편집 및 수정할 수 있는 기능과 이용자에게 단일 요소로 서비스하는 것이 핵심기능이라 할 수 있다. 기능적인 측면에서 신규 데이터베이스 구성을 위해 핵심적인 요소들을 도출하면 <표 4>와 같이 정리할 수 있다.

우선 신규 시스템의 원천데이터는 TTA MDR로 이 데이터를 반입하기 위한 기능이 필요하다. 반입한 데이터는 신규 시스템에서 새로운 메타데이터 요소로 등록해야하므로 요소등록 기능 역시 필요하며, 각 요소 간의 관계 설정과 새로운 설명 등을 추가하기 위한 수정 기능이

필요하다. 마지막 태깅은 필수적인 요소로 보기는 힘들지만 TTA MDR의 주요 대상이 국제표준 용어를 대상으로 하고 있고 영어로 표기되는 것이 대부분으로 한글을 포함한 다양한 검색 접근점을 확보하기 위해 구성하였다.

<표 4>의 신규 시스템의 기능 요소를 충족시키기 위해 필요한 테이블을 정리하면 <표 5>와 같다.

<표 5>의 테이블은 기존 TTA MDR 테이블 중 클래스, 속성, 코드와 관련된 요소를 중심으로 설계하였다. 데이터 반입 기능을 위한 테이블은 1, 2번, 요소등록과 수정을 위한 용어 관리 기능을 위한 테이블은 3, 4, 6, 7번이다. 마지막으로 검색 지원을 위한 태그는 5번이다. 6, 7번은 클래스, 속성, 코드 간의 관계설정을 위한 것으로 요소등록, 수정과 직접적으로 연관된다. 이를

<표 4> 신규 시스템 데이터베이스 구성을 위한 기능 요소

구분	기능	세부 필요 기능
1	데이터 반입	• TTA MDR로부터 원천데이터 반입
2	요소등록	• TTA MDR에서 반입한 데이터를 신규 시스템에 등록
3	요소수정	• 등록된 요소의 상세설명 등 정보 수정 • 클래스, 속성, 값영역별 간의 관계 설정
4	태깅	• 요소에 대한 다양한 접근을 위한 태깅 정보 입력 및 수정

<표 5> 신규 시스템 구성을 위한 테이블 목록

구분	테이블명	내용	관련기능
1	waitingCode	원천데이터에서 입수한 코드 데이터(값 영역)	데이터 반입
2	waitingTerm	원천데이터에서 입수한 요소 데이터(클래스, 속성)	데이터 반입
3	code	메타데이터 Code 정보가 있는 테이블	요소등록, 요소수정(값영역의 코드)
4	term	메타데이터 요소에 대한 정보가 있는 테이블	요소등록, 요소수정(클래스, 속성 입력)
5	tag	메타데이터 요소에 대한 태그 정보가 있는 테이블	태깅
6	relatedCode	term테이블과 code 테이블간의 관계 정보 테이블	요소등록, 요소수정
7	relatedTag	term 테이블과 tag 테이블 간의 관계 정보 테이블	요소등록, 요소수정
8	relatedTerm	term 테이블의 내부 관계 정보가 있는 테이블	요소등록, 요소수정

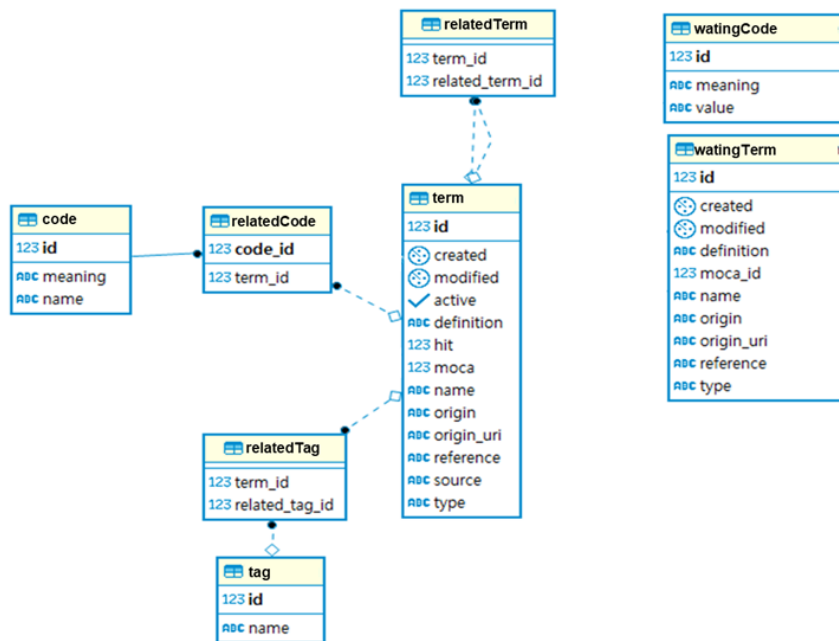
기반으로 테이블 간의 관계를 ERD로 표현하면 <그림 8>과 같다. 각 테이블의 컬럼 구성을 포함한 테이블 정의서는 [부록 2]에 첨부하였으며 각 컬럼은 원천 데이터인 TTA MDR 테이블이 갖고 있는 컬럼명과 유형을 준수하였다.

<그림 8>에서 보듯이 신규 데이터베이스는 핵심은 'term 테이블'이다. 'term 테이블'의 'type' 컬럼에서 단일요소가 클래스, 속성, 코드 중 어디에 속하는지를 선택할 수 있도록 하였다. 또한 'Moca' 컬럼은 TTA MDR에서 가지고 있는 ID 값으로 향후 신규 시스템과 기존 시스템을 연계하고자할 때 사용할 수 있도록 하였다. 'waitingCode', 'waitingTerm' 테이블은 TTA MDR에서 입력한 요소를 임시로 저장하여 신규 시스템에 반입하기 전 임시 보관용이다.

5.4 이용자 서비스

신규 데이터베이스 구성하여 시스템을 구축한 이용자 서비스는 TTA MDR 시스템과 다르게 클래스, 속성명, 코드로 단순화시킬 수 있다. 검색결과화면으로 이를 비교해보면 <그림 9>는 TTA MDR에서 'Person'이라는 클래스를 검색했을 경우이며, <그림 10>은 신규 시스템의 Person 검색결과 화면이다.

<그림 9>에서 보는 것처럼 TTA MDR의 경우 특정 클래스 검색 시 클래스와 관련된 데이터 요소 개념 검색까지 확인이 가능하다. 즉, ISO/IEC 11179 표준을 준수하는 경우는 특정 메타데이터 요소 개념에 집중하여 관련 메타데이터 요소, 클래스, 속성 등으로 연결하여 결과를 보여준다. 클래스를 검색하더라도 메타데이



<그림 8> 신규 데이터베이스 구성을 위한 ERD

Person (Object Class)

Definition
 사람을 표현하기 위한 클래스로 살아있는 사람, 죽은 사람, 허구/가상의 사람 모두를 포함할 수 있음.

References
 Person은 [schema.org](#)에서 정의하고 있는 용어집에서 채택한 것입니다.
 Schema.org는 Google, Microsoft, Yahoo!가 웹의 데이터를 설명하는 공통 용어를 만들어 일을 개선하기 위해 시작한 공동 프로젝트입니다. HTML 페이지에 schema.org 마크업을 추가하면 다양한 검색엔진, 검색 서비스를 비롯한 여러 기업 및 제품에서 사이트의 데이터를 이해할 수 있게 됩니다. schema.org에 대해 더 자세히 알아보려면 [schema.org FAQ](#)를 참조하시기 바랍니다.
 Schema.org는 웹 데이터에 대한 공통 용어로 Person의 Origin URI를 선택하면, Microdata, RDFa, JSON-LD로 표현한 예제정보를 함께 검토할 수 있습니다.

Submitting Organisation
 한국정보통신기술협회(TTA, Telecommunication Technology Association)

Responsible Organisation
 한국정보통신기술협회(TTA, Telecommunication Technology Association)

Additional Attributes

Slot name	타입	Value	Similar
short_name		Person	1

🔍 Actions

📄 Download

Person

Type: Object Class

Identifiers: UUID - be91910-3b68-11e7-83f9-055de2af637c

Origin URI: <http://schema.org/Person>

Endorsed by (Change):

- TTA_MDR_Initiative_RK : Standard on 2017년 8월 22일

View registration history

Last updated: 2년, 10개월 전 By 박진호

Created: 2017-05-18

Supersedes (Add):

- None

Superseded by (Add):

- None

Workgroup: TTA_MDR_Initiative

해소포자: View history

Discussions: Start a new discussion about this item

Related content

Data Element Concepts implementing this Object Class

- 사람(person)-일하는 곳(worksFor)
 Status: (TTA_MDR_Initiative_RK: Standard)
 사람이 일하는 조직을 표현하기 위한 데이터 요소 개념.
- 사람(person)-배우자(spouse)
 Status: (TTA_MDR_Initiative_RK: Standard)
 사람의 배우자, 반려자를 표현하기 위한 데이터 요소 개념.
- 사람(person)-후원자(funder)
 Status: (TTA_MDR_Initiative_RK: Standard)
 재정적인 지원을 통해서 무언가를 후원하는 사람을 표현하기 위한 데이터 요소 개념.
- 사람(person)-후원자(sponsor)
 Status: (TTA_MDR_Initiative_RK: Standard)
 사람 중 후원자를 표현하기 위한 데이터 요소 개념.
- 사람(person)-직장연락처(workLocation)
 Status: (TTA_MDR_Initiative_RK: Standard)
 사람의 직장 연락처를 표현하기 위한 데이터 요소 개념.
- 사람(person)-형제자매(sibling)
 Status: (TTA_MDR_Initiative_RK: Standard)
 사람의 형제 자매 관계를 표현하기 위한 데이터 요소 개념.
- 사람(person)-가족관계(relatedTo)
 Status: (TTA_MDR_Initiative_RK: Standard)
 사람의 일반적인 가족 관계를 표현하기 위한 데이터 요소 개념.
- 사람(person)-팩스번호(faxNumber)
 Status: (TTA_MDR_Initiative_RK: Standard)
 특정 사람과 연결될 수 있는 팩스 번호를 표현하기 위한 데이터 요소 개념.
- 사람(person)-출연한 이벤트(performerIn)
 Status: (TTA_MDR_Initiative_RK: Standard)
 특정 사람이 출연한 이벤트를 표현하기 위한 데이터 요소 개념.

<그림 9> TTA MDR에서 'Person' 클래스 상세검색 결과화면

Person	
유형	Class
활용빈도	30
출처	Schema.org
Origin URI	http://schema.org/Person
정의	사람을 표현하기 위한 클래스로 살아있는 사람, 죽은 사람, 허구/가상의 사람 모두를 포함할 수 있음.
참조정보	Person은 schema.org 에서 정의하고 있는 용어집에서 채택한 것입니다. Schema.org는 Google, Microsoft, Yahoo!가 웹의 데이터를 설명하는 공통 용어를 만들어 일을 개선하기 위해 시작한 공동 프로젝트입니다. HTML 페이지에 schema.org 마크업을 추가하면 다양한 검색엔진, 검색 서비스를 비롯한 여러 기업 및 제품에서 사이트의 데이터를 이해할 수 있게 됩니다. schema.org에 대해 더 자세히 알아보려면 schema.org FAQ 를 참조하시기 바랍니다. Schema.org는 웹 데이터에 대한 공통 용어로 Person의 Origin URI를 선택하면, Microdata, RDFa, JSON-LD로 표현한 예제정보를 함께 검토할 수 있습니다.
관련클래스	
관련속성	worksFor spouse funder sponsor workLocation sibling relatedTo faxNumber performerIn parent owns makesOffer memberOf knows jobTitle colleague children brand homeLocation givenName follows familyName email birthPlace birthDate address additionalName nationality gender affiliation alumniOf award contactPoint deathDate deathPlace hasOfferCatalog hasPOS height honorificPrefix honorificSuffix isicV4 memberOf owns seeks telephone weight
관련코드	
기술방식	

<그림 10> 신규 시스템에서 'Person' 클래스 상세검색 결과화면

터 요소, 혹은 메타데이터 요소 개념을 확인하면 '클래스, 속성, 값영역의 데이터 유형'을 하나의 값으로 제시하여 활용하고자하는 이용자 입장에서 어려움이 존재한다.

반면 <그림 10>에서 보는 것처럼 클래스와 속성을 중심으로 설계한 시스템은 이용자가 직접 시스템 설계와 구현에 적용해야하는 단일 요소를 중심으로 검색결과를 제시한다. 또한 클래스와 클래스, 속성과 속성 간의 포함관계와 상·하위 관계, 연결 정보 확인이 가능하다. 관계 정보는 기존 MDR에서는 구현이 어려웠다. 결과적으로 신규 시스템은 ISO/IEC 11179 표준을 준수하는 시스템을 그대로 유지하면서 표준에 대한 이해가 없더라도 쉽게 필요한 요소 검색과 활용이 가능하다고 할 수 있다. 즉 표준이 갖는 가치를 유지하면서 보다 이용자 친화적인 시스템 구성을 했다고 할 수 있다.

6. 결론

본 연구에서는 ISO/IEC 11179 MDR 표준을 준수하는 시스템의 데이터베이스를 재설계하여 클래스, 속성 값영역의 코드값을 중심으로 하는 시스템 재설계하고 간단하게 검색결과 화면을 확인했다. 지식정보를 획득하고자하는 사람들에게 메타데이터는 찾고자하는 유형의 지식정보탐색은 물론 관련 정보로의 이동을 용이하게 하는 중요한 역할을 한다. 이런 서비스를 제공해주는 기관이나 서비스의 입장에서 보면 점점 개방화와 다른 정보자원의 연결로 보다 풍부한 탐색결과를 제공해 줄 수 있는 중요한 도구이다. 즉, 메타데이터의 표준화는 이용

자, 관리자입장에서 모두 양적, 질적으로 좋은 결과 도출을 가능하게 해준다고 할 수 있다.

이런 메타데이터는 정보화 환경이 변화하면서 다양한 유형의 지식정보자원을 관리, 서비스해야하는 필요성으로 중요성이 더 증가했다고 볼 수 있다. 또 문서형태의 정보서비스 외에 데이터 개방에 대한 요구 증가와 함께 개방 데이터를 조합하여 새로운 서비스를 만들어내는 시대에 그 중요성이 더해지고 있다고 할 수 있다. 자연스럽게 메타데이터 자체도 중요한 데이터 정보자원으로 관리와 서비스가 필요하다고 할 수 있다. ISO/IEC 11179 표준은 단순히 표준 메타데이터를 관리 서비스하기 위한 틀을 제공하기보다 개념적으로 메타데이터라는 것을 어떻게 바라봐야하는지에 대한 지침인 동시에 실제 구축, 운영에 필요한 관리적, 물리적, 시스템적, 운영적 요건을 제공해주고 있다. 반면, 이런 정교하고 복잡한 구조는 실제 시스템 구축 시 해당 표준에 대한 이해를 필요로 한다는 점에서 일부 문제점을 노출할 수도 있다.

본 논문에서는 ISO/IEC 11179 표준을 준수하는 TTA의 MDR 시스템의 데이터베이스 구조를 클래스, 속성, 코드를 중심으로 재설계하였다. 이 과정에서 기존 시스템의 운영과 유지는 물론 데이터베이스 구성까지 유지하면서 좀 더 쉽게 메타데이터 탐색과 이용이 용이하도록 연계가능한 구조의 데이터베이스를 만들었다. 결과적으로 표준을 이해하고 있는 사람과 그렇지 못한 사람 모두를 포괄할 수 있는 접근점과 서비스를 만들었고, 이용자들은 좀 더 쉽게 직접 활용 가능한 단일 요소로의 접근과 관련 요소 탐색으로 이어질 수 있게 되었다.

사실 표준을 준수하는 시스템은 다양한 형태

로 존재할 수 있다. 표준을 준수하고 같은 형태의 데이터베이스를 유지하더라도 이용자에게 보여주는 서비스의 구성 역시 달라질 수 있다. 본 논문의 결과는 표준이 갖는 사상적 가치를 인정하면서 정보화 시대의 변화에 대응할 수 있는 또 다른 시스템 구성의 가능성을 보여주었다. 그렇지만 본 연구의 대상이 TTA MDR 하나의

사례라는 점은 이 연구가 갖는 한계로 볼 수 있다. 향후 ISO 11179 표준을 준수하는 또 다른 시스템들이 등장한다면 본 연구가 비교연구의 대상이 될 수 있을 것으로 보인다. 시스템의 가치 평가는 이용자에게 의해서만 가능한 영역으로 향후 두 개 시스템의 활성화 정도에 따라 다양한 연구가 이어질 수 있기를 기대한다.

참 고 문 헌

- [1] 고영만. 2005. ISO/IEC 11179 표준 패밀리. [online] [cited 2020. 8. 3.]
<<http://home.skku.edu/~ymko/proceedings/11179.pdf>>
- [2] 고영만, 배경재. 2011. 상이한 데이터 구조의 데이터베이스간 통합 운영방안 연구. 『한국문헌정보학회지』, 45(3): 69-85.
- [3] 남영광, 서태설, 황상원. 2005. SO/IEC 11179에 따른 산업기술정보 메타데이터 표준화. 『Journal of Information Science Theory and Practice』, 36(1): 57-75.
- [4] 남태우, 이승민. 2010. 메타데이터의 의미론적 확장에 관한 연구. 『한국문헌정보학회지』, 44(4): 373-393.
- [5] 심경. 2003. 메타데이터 통합 방안. 『한국도서관·정보학회지』, 34(3): 169-192.
- [6] 오삼균. 2004. 국가지식정보자원관리를 위한 시맨틱웹 설계 및 정책 방향에 관한 연구. 『한국비블리아학회지』, 15(1): 43-67.
- [7] 오삼균. 2005. 시맨틱웹 기반 메타데이터 레지스트리 설계에 관한 연구. 『한국도서관·정보학회지』, 36(3): 109-136.
- [8] 이승민. 2013. 마스터 데이터를 활용한 메타데이터 통합 프레임워크 구축. 『한국도서관·정보학회지』, 44(1): 201-225.
- [9] 정혜진, 백두권, 정동원. 2009. ISO/IEC 11179 기반의 온톨로지 확장 모델. 『정보과학회논문지: 데이터베이스』, 36(5): 386-398.
- [10] Bernstein, D., Ludvigson, E., Sankar, K., Diamond, S. and Morrow, M. 2009. "Blueprint for the intercloud-protocols and formats for cloud computing interoperability." In *2009 fourth international conference on Internet and web applications and services*, 24-28 May 2009. 328-336. Venice/Mestre, Italy.

- [11] Davies, J., Harris, S., Crichton, C., Shukla, A. and Gibbons, J. 2008. "Metadata standards for semantic interoperability in electronic government." In *Proceedings of the 2nd international conference on Theory and practice of electronic governance*, December 2008, 67-75. Cairo, Egypt.
- [12] ISO. n.d. STANDARDS BY ISO/IEC JTC 1/SC 32. [online] [cited 2020. 8. 3.]
<<https://www.iso.org/committee/45342/x/catalogue/p/1/u/0/w/0/d/0>>
- [13] International Organization for Standardization. 2013. Information technology - Metadata registries (MDR) - Part 3: Registry metamodel and basic attributes. Switzerland: International Organization for Standardization.
- [14] Lewis, G. A. 2013. "Role of standards in cloud-computing interoperability." In *2013 46th Hawaii international conference on system sciences*, 7-10 January 2013, 1652-1661. Wailea, Maui, HI USA.
- [15] Mahmud, R., Koch, F. L. and Buyya, R. 2018. "Cloud-fog interoperability in IoT-enabled healthcare solutions." In *Proceedings of the 19th international conference on distributed computing and networking*, January, 2018, 1-10. Varanasi, India.
- [16] Song, S. 2017. "Competition law and interoperability in cloud computing." *Computer law & security review*, 33(5): 659-671.
- [17] Stausberg, J., Löbe, M., Verplancke, P., Drepper, J., Herre, H. and Löffler, M. 2009. "Foundations of a metadata repository for databases of registers and trials." *Studies in Health Technology and Informatics*, 150: 409-413.
- [18] Yongsiriwit, K., Sellami, M. and Gaaloul, W. 2016. "A semantic framework supporting cloud resource descriptions interoperability." In *2016 IEEE 9th International Conference on Cloud Computing (CLOUD)*, Jun 27 - Jul 2 2016, 585-592. San Francisco, USA.

• 국문 참고자료의 영어 표기

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

- [1] Ko, Young Man. 2005. "ISO/IEC 11179 Standard Family." [online] [cited 2020. 8. 3.]
<<http://home.skku.edu/~ymko/proceedings/11179.pdf>>
- [2] Ko, Young Man and Bae, Kyoung Jae. 2005. "A Study of the Integrated Operation for Databases with Different Data Structures." *Journal of The Korean Society for Library and Information Science*, 45(3): 69-85.
- [3] Nam, Young Kwang, Seo, Tae Sul and Hwang, Sang Won. 2009. "Standardization of Industrial

- Information Metadata Based on ISO/IEC 11179.” *Journal of Information Science Theory and Practice*, 36(1): 57-75.
- [4] Nam, Tae Woo and Lee, Seung Min. 2010. “Study on the Semantic Extension of the Concept of Metadata.” *Journal of Korean Society for Library and Information Science*, 44(4): 373-393.
- [5] Sim, Kyoung. 2003. “Integration of Different Metadata Formats.” *Journal of Korean Library and Information Science Society*, 34(3): 169-192.
- [6] Oh, Sam Gyun. 2004. “A Study of Designing Semantic Web and Policy Directions for National Knowledge and Information Management.” *Journal of the Korean Biblia Society for Library and Information Science*, 15(1): 43-67.
- [7] Oh, Sam Gyun. 2005. “Designing a Metadata Registry Using SemanticWeb Technology.” *Journal of Korean Library and Information Science Society*, 36(3): 109-136.
- [8] Lee, Seung Min. 2013. “Construction of Framework for Metadata Integration Using Master Data Approach.” *Journal of Korean Library and Information Science Society*, 44(1): 201-225.
- [9] Jeong, Hye jin, Baik, Doo Kwon and Jeong, Dong Won. 2009. “An Ontology Population Model based on ISO/IEC 11179.” *Journal of KISS: Databases*, 36(5): 386-398.

[부록 1] TTA MDR 데이터베이스 테이블 목록

구분	테이블명	내용
1	aristotle_mdr__concept	메타데이터 정보가 있는 테이블
2	aristotle_mdr_concepthelp	Help페이지의 개념정보가 있는 테이블
3	aristotle_mdr_conceptualdomain	개념적 영역(Conceptual domain)에 해당하는 메타데이터 id가 있는 테이블
4	aristotle_mdr_dataelement	데이터 요소(Dataelement)에 해당하는 메타데이터 id가 있는 테이블
5	aristotle_mdr_dataelementconcept	데이터 요소(Dataelement concept)에 해당하는 메타데이터 id가 있는 테이블
6	aristotle_mdr_dataelementderivation	데이터 요소 파생(Dataelement derivation) 규칙 정보가 있는 테이블
7	aristotle_mdr_dataelementderivation_derives	데이터 요소 파생(Dataelement derivation)에 해당하는 정보가 있는 테이블
8	aristotle_mdr_dataelementderivation_inputs	데이터 요소 파생(Dataelement derivation)에 해당하는 정보가 있는 테이블
9	aristotle_mdr_datatype	데이터 유형(data type)에 해당하는 메타데이터 id가 있는 테이블
10	aristotle_mdr_discussioncomment	토론(Discussion)의 comment 정보가 있는 테이블
11	aristotle_mdr_discussionpost	토론(Discussion)의 요청 정보가 있는 테이블
12	aristotle_mdr_discussionpost_relatedItems	토론(Discussion)의 요청과 관련된 메타데이터 id가 있는 테이블
13	aristotle_mdr_help_concepthelp	Help페이지의 개념 정보가 있는 테이블
14	aristotle_mdr_help_helpbase	모든 help 페이지 정보가 있는 테이블
15	aristotle_mdr_help_helppage	기본 help 페이지 id가 있는 테이블
16	aristotle_mdr_helpbase	모든 help 페이지 정보가 있는 테이블(not use)
17	aristotle_mdr_helppage	기본 help 페이지 id가 있는 테이블(not use)
18	aristotle_mdr_identifiers_namespace	Identifier에 대한 정보가 있는 테이블
19	aristotle_mdr_identifiers_scopedidentifier	Identifier에 대한 정보가 있는 테이블
20	aristotle_mdr_measure	Measure에 해당하는 정보가 있는 테이블
21	aristotle_mdr_objectclass	객체 클래스(object class)에 해당하는 정보가 있는 테이블
22	aristotle_mdr_organization	등록 기관에 해당하는 정보가 있는 테이블
23	aristotle_mdr_organization_managers	등록 기관에 속해있는 유저에 대한 id가 있는 테이블
24	aristotle_mdr_permmissiblevalue	승인된 값에 대한 정보가 있는 테이블
25	aristotle_mdr_possumprofile	활성화된 유저 정보가 있는 테이블
26	aristotle_mdr_possumprofile_favourites	즐거찾기 정보가 있는 테이블
27	aristotle_mdr_property	속성(property)에 해당하는 정보가 있는 테이블
28	aristotle_mdr_registrationauthority	등록 권한에 대한 정보가 있는 테이블
29	aristotle_mdr_registrationauthority_registrars	등록기관에 속해 있는 관리자 id가 있는 테이블

구분	테이블명	내용
30	aristotle_mdr_reviewrequest	검토 요청에 대한 정보가 있는 테이블
31	aristotle_mdr_reviewrequest_concepts	검토 요청에 대한 메타데이터 id 가 있는 테이블
32	aristotle_mdr_slots_slot	슬롯에 해당하는 정보가 있는 테이블
33	aristotle_mdr_status	메타데이터 상태에 대한 테이블
34	aristotle_mdr_supplementaryvalue	셋에 대한 파일 상세 정보가 저장 되는 테이블
35	aristotle_mdr_unitofmeasure	Unit of measure에 속해 있는 메타데이터 id가 있는 테이블
36	aristotle_mdr_valuedomain	값 영역(value domain)에 대한 정보가 있는 테이블
37	aristotle_mdr_valuemeaning	값(value)에 대한 정보가 있는 테이블
38	aristotle_mdr_workgroup	속정 정보가 있는 테이블
39	aristotle_mdr_workgroup_managers	데이터 셋에 대한 마스터 정보가 있는 테이블
40	aristotle_mdr_workgroup_stewards	작업 그룹에 속해 있는 대표자(steward) 정보가 있는 테이블
41	aristotle_mdr_workgroup_submitters	작업 그룹에 속해 있는 제출자(submitter) 정보가 있는 테이블
42	aristotle_mdr_viewers	작업 그룹에 속해 있는 열람자(viewer) 정보가 있는 테이블
43	auth_group	그룹 정보가 있는 테이블
44	auth_group_permissions	그룹의 권한 정보가 있는 테이블
45	auth_permission	권한에 대한 정보가 있는 테이블
46	auth_user	유저 정보가 있는 테이블
47	auth_user_groups	유저의 그룹 정보가 있는 테이블
48	auth_user_user_permissions	유저의 권한 정보가 있는 테이블
49	djagno_admin_log	유저의 활동 이력 정보에 대한 테이블
50	django_content_type	App과 model이 정보에 대한 테이블
51	django_migrations	Migration에 대한 정보가 있는 테이블
52	django_session	Django에서 관리하는 세션 정보가 있는 테이블
53	notifications_notification	알림(notification) 정보가 있는 테이블
54	reversion_revision	개정 이력에 대한 정보가 있는 테이블
55	reversion_version	메타데이터 version 정보가 있는 테이블
56	search_keyword	검색 이력에 대한 정보가 있는 테이블
57	static_precompiler_dependency	Precompiler에 대한 정보가 있는 테이블(not use)

[부록 2] 테이블 정의서

1. code

컬럼명	comment	데이터 타입	데이터사이즈	Null 컬럼	PK	FK
id	code 일련번호	Integer		N	Y	
Meaning	의미	Varchar	255	Y		
name	Code의 값	varchar	255	Y		

2. tag

컬럼명	comment	데이터 타입	데이터사이즈	Null 컬럼	PK	FK
Id	태그 일련번호	Integer		N	Y	
Name	태그 이름	varchar	255	Y		

3. term

컬럼명	comment	데이터 타입	데이터사이즈	Null 컬럼	PK	FK
id	메타데이터 일련번호	Integer		N	Y	
Created	생성일	Timestamp with time zone	Y			
Modified	수정일	Timestamp with time zone	Y			
Active	등록 여부	Bool		N		
Definition	설명	Text		Y		
Hit	조회수	Integer		Y		
Moca	내부데이터 일 경우 Moca ID 값	Integer		Y		
Name	메타데이터 이름	Varchar	255	N		
Origin	출처	Varchar	255	Y		
origin_uri	출처 uri	Varchar	255	Y		
Reference	참조 정보	Text		Y		
Source	반입 방법	Varchar	255	N		
type	유형(class,property,code)	varchar	255	N		

4. waitingCode

컬럼명	comment	데이터 타입	데이터사이즈	Null 컬럼	PK	FK
Id	대기 코드 일련번호	Integer		N	Y	
meaning	코드 설명	Varchar	255	Y		
value	코드 값	Varchar	255	Y		

5. waitingTerm

컬럼명	comment	데이터 타입	데이터사이즈	Null 컬럼	PK	FK
id	대기 메타데이터 일련번호	Integer		N	Y	
Created	생성일	Timestamp with time zone	Y			
Modified	수정일	Timestamp with time zone	Y			
Definition	설명	Text		Y		
Moca_id	내부데이터 일 경우 Moca ID 값	Integer		Y		
Name	메타데이터 이름	Varchar	255	N		
Origin	출처	Varchar	255	Y		
origin_uri	출처 uri	Varchar	255	Y		
Reference	참조 정보	Text		Y		
type	유형(class,property,code)	varchar	255	N		

6. relatedCode

컬럼명	comment	데이터 타입	데이터사이즈	Null 컬럼	PK	FK
Term_id	메타데이터 요소 아이디	Integer		N		Y
Code_id	코드 아이디	integer		N		Y

7. relatedTag

컬럼명	comment	데이터 타입	데이터사이즈	Null 컬럼	PK	FK
term_id	메타데이터 요소 아이디	Integer		N		Y
Related_tag_id	태그 아이디	integer		N		Y

8. relatedTerm

컬럼명	comment	데이터 타입	데이터사이즈	Null 컬럼	PK	FK
term_id	메타데이터 요소 아이디	Integer		N		Y
Related_term_id	메타데이터 요소 아이디	integer		N		Y