

# OWL Ontology 저작도구의 설계 및 구현

안병규\*, 김동혁\*, 장창복\*, 고병오\*\*, 조현규\*\*\*, 송병열\*\*\*, 최의인\*

\*한남대학교 컴퓨터공학과

\*\*공주교육대학교 컴퓨터교육과

\*\*\*한국전자통신연구원

e-mail: bgahn@dbl.hannam.ac.kr

## Design and Implementation of OWL Ontology Authoring Tool

Byung-Gyu Ahn\*, Dong-Hyuk Kim\*, Chang-Bok Jang\*,

Byung-Oh Goh\*\*, Hyun-Kyu Cho\*\*\*, Byoung-Youl Song\*\*\*, Eui-In Choi\*

\*Dept of Computer Engineering, Hannam University

\*\*Dept. of Computer Education, Gongju National University of Education

\*\*\*Electronics and Telecommunications Research Institute

### 요 약

현재의 웹 검색은 단순히 키워드 매칭만을 수행하기 때문에 검색 결과가 사용자가 원하는 정보와는 의미적으로 상이한 결과들을 많이 포함하고 있다. 사용자가 원하는 정보와 의미적으로 정확히 일치하는 정보들을 추출하기 위해서는 웹 자원에 대한 정확한 의미 부여와 함께, 이를 파악하고 선택적으로 획득, 통합, 가공할 수 있는 온톨로지와 같은 시맨틱 웹 환경요소들이 필요하다. 웹 기술 표준화 단체인 W3C에서는 이와 같은 웹 자원에 대한 의미표현 기술로 OWL(Web Ontology Language)이라는 웹 온톨로지 표현 언어를 발표하였으나 아직 이를 효과적으로 저작, 편집할 수 있는 도구는 전무한 실정이다. 따라서 본 논문은 OWL을 기반으로 하여 온톨로지의 생성 및 편집을 효과적으로 제공할 수 있는 통합된 저작도구를 설계 및 구현하였다.

### 1. 서론

팀 버너스 리가 1990년에 제안한 웹은 정보의 양적인 면과 질적 수준에서 지난 10여 년간 급속하게 성장하여 왔다. 웹의 사용 환경이 급속도로 성장함에 따라 웹 사이트와 사용자의 수도 기하급수적으로 증가하였다. 정보의 양이 증가하고 웹 이용자의 수준과 요구사항이 다양하고 복잡해짐에 따라 정확하고 적절한 정보를 제공하는 것이 중요한 문제가 되고 있다. 필요한 정보만을 효율적으로 추출하기 위해서 웹 정보들 간의 관계와 정보 자원들 사이의 의미적 연관성에 대한 메타정보와 컴퓨터가 이해할 수 있는 지식 표현 수단이 필요하다.

팀 버너스 리는 1998년에 기계가 정보를 이해하  
본 연구는 한국전자통신연구원의 "시맨틱 비즈니스 문서 편집기 개발"의 연구 결과임

고 처리하기 위하여 기존 웹을 확장하는 시맨틱 웹을 제안하였다. 시맨틱 웹에서는 웹 자원에 유일한 식별자(URI)를 부여하고, 온톨로지를 통하여 표준화된 방식으로 표현된 지식을 이용함으로써, 정보를 공유할 수 있도록 하고 있다. 기계가 스스로 정보의 의미를 인식함으로써 사용자가 원하는 정보와 의미적으로 일치하는 정보를 검색할 수 있으며, 검색된 정보에서 지식을 유추하는 기능도 제공할 수 있다. 즉, 시맨틱 웹은 사람이 웹 정보에 대하여 의미를 파악하고 의미에 따라 필요한 정보를 선택하듯이 컴퓨터가 웹 정보의 의미를 이해하고 의미에 따라 선택적으로 정보를 획득, 통합, 가공할 수 있는 웹 환경을 제공하는 것이다[1]. 시맨틱 웹에서 이러한 기능 및 수단을 제공할 수 있도록 하기 위한 기술 요소들로는 메타데이터, XML, RDF, 온톨로지 등이 있다[2, 3].

온톨로지는 시맨틱 웹의 핵심 기술로 에이전트가 지식을 표현하고 절차적 추론을 처리하기 위해 사용된다. 이에 따라 XOL, SHOW, OML, DAML+OIL 등 여러 온톨로지 표현 언어와 Protégé-2000, OntoEdit, OilEd 등의 온톨로지 저작 및 활용 도구가 다양하게 연구되고 있다.

하지만 기존의 온톨로지 저작도구는 각기 다른 온톨로지 언어를 기반으로 하고 있으며, 각기 다른 방식으로 개발되었기 때문에 온톨로지를 저작하고 탐색함에 있어 서로 연동하기가 어렵다는 문제점이 있다. 이러한 이유로 W3C를 중심으로 온톨로지 표현 언어에 대한 표준화 작업이 진행되고 있다. 그 결과로 새로운 온톨로지 표현 언어인 OWL을 발표하였다[5, 6, 7, 8]. 본 논문에서는 W3C의 온톨로지 언어인 OWL을 기반으로 온톨로지를 저작 및 편집할 수 있는 저작도구를 설계하고 구현한 결과를 제시한다.

## 2. 관련연구

온톨로지와 관련된 연구는 크게 온톨로지 표현 언어에 관한 연구와 이를 활용할 수 있는 도구에 관한 연구로 분류할 수 있다. 여러 가지 온톨로지 저작 도구 중 가장 대표적인 것으로는 Protégé-2000, OntoEdit, OilEd, DUET 등이 있다.

가장 대표적인 온톨로지 저작 도구로 손꼽히는 Protégé-2000은 국립의료도서관(National Library of Medicine), NSF(National Science Foundation), DARPA(Defense Advanced Research Projects Agency)의 후원을 받아 스탠포드 의과대학의 의료정보학과(Stanford University School of Medicine, Stanford Medical Informatics)에서 지식 기반의 구조를 작성하기 위한 도구로 15년간의 연구 기간을 거쳐 개발되었다[9]. 오랜 개발 기간이 소요된 만큼 가장 완성도가 뛰어나며, 가장 널리 이용되는 저작 도구이다.

맨체스터 대학에서 개발된 OilEd는 DAML+OIL을 이용하여 온톨로지를 구축하는 온톨로지 저작 도구로써, 현재 버전에서는 대규모 온톨로지 구축, 온톨로지 통합 및 migration 등의 완전한 온톨로지 구축 환경을 제공하지는 않지만, 이용자가 온톨로지를 구축해보고 FaCT 추론기를 이용하는 방법 등에 대해 시현해볼 수 있는 기능을 제공한다[10].

OntoEdit은 독일의 ontoprise® GmbH사에서 개발하였으며, 그래픽을 이용하여 온톨로지의 개발과 유지보수를 제공하는 온톨로지 공학 환경(Ontology

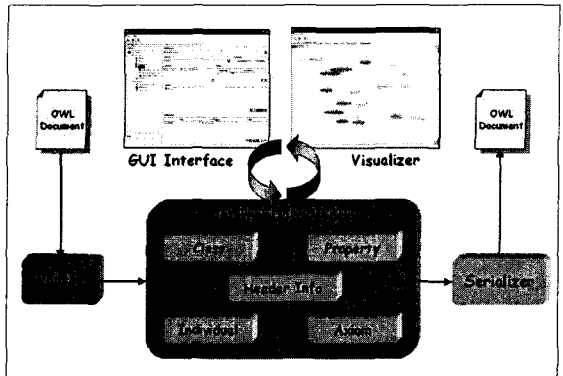
Engineering Environment)이다[11]. OntoEdit은 강력한 내부 온톨로지 모델을 가지고 있으며 이 모델은 개념(concept), 관계(relation), 원칙(axioms)이 가능한 중립적 모델링 표현 언어를 제공한다. 이 틀은 직관적인 인터페이스를 통해서 온톨로지에 대한 시각화 및 탐색을 지원한다.

DUET은 AT&T에서 개발되었고, DAML을 위해 UML(Unified Modeling Language) 시각화와 저작 환경을 제공한다. 이 틀의 특징은 유효성 검증을 통해 UML 다이어그램을 DAML+OIL로 생성시켜준다.

## 3. OWL 저작도구

### 3.1 시스템 구조

<그림 1>은 본 연구에서 설계한 OWL 저작도구의 전체 시스템 구조로서, Parser, Internal Data Model, GUI Interface, Visualizer, Serializer로 구성되어 있다.



<그림 1> 전체 시스템 구조

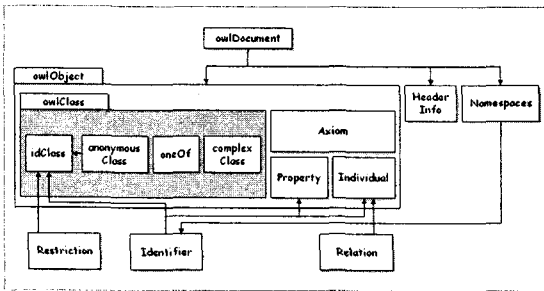
### 3.2 Parser

Parser는 텍스트 형태의 OWL 문서를 입력받아 OWL 문서에서 요소들을 추출하여 내부 데이터 모델의 객체로 변환한다. Parsing은 두 단계로 수행되며, 먼저 pre-parsing 단계에서 URI 식별자를 갖는 Class, Property, Individual 들을 추출하여 내부 데이터 모델에 Dummy 객체를 생성한다. post-parsing 단계에서는 이들 간의 참조관계에 따라 객체들의 링크를 연결하고, 객체의 실제 Contents를 생성된 객체에 반영시킨다.

### 3.3 Internal Data Model

Parsing 단계에서 추출된 요소들은 내부 데이터

모델로 변환되어 메모리에 저장된다. 내부 데이터 모델의 설계는 OWL 스펙에서 요구하는 의미 표현을 충실하게 제공할 수 있도록 owlDocument, owlObject, owlClass, idClass, anonymousClass, oneOf, complexClass, Axiom, Property, Individual, HeaderInfo, Namespace dataValue, Identifier, Restriction, Relation의 16개의 클래스로 정의하였다. <그림 2>는 정의한 Class들의 관계를 나타낸 Diagram이고 <그림 3>은 Property 정보를 표현하기 위한 Java 소스이다. OWL 요소는 내부 모델의 객체로 1:1 매칭되며, 모든 요소들은 owlDocument 객체에 연결된다. 또한 GUI 인터페이스에서 반복적으로 사용되는 Class, Property의 Tree-Model도 owlDocument 객체에서 통합 관리된다.



<그림 2> Class Diagram

```
public class Property implements Comparable, owlObject {
    public static final int DATA_PROPERTY = 1;
    public static final int OBJECT_PROPERTY = 2;
    public static final int DEPRECATED_PROPERTY = 3;
    public static final int IMPORTED_PROPERTY = 4;

    public Identifier ID;
    public owlDocument doc = null;

    public int type=OBJECT_PROPERTY;

    public String comment = new String();
    public Vector vLabel = new Vector();

    public Vector subPropertyOf = new Vector();
    public Vector inverseOf = new Vector();
    public Property equivalentProperty;
    public Vector domain = new Vector();
    public Vector range = new Vector();

    public boolean Functional;
    public boolean inverseFunctional;
    public boolean Symmetric;
    public boolean Transitive;

    public Vector vTreeNodes = new Vector();
    public TreeSet tsTreeChild = new TreeSet();
    public Vector vReferencedObject = new Vector();
    .....(중략)
}
```

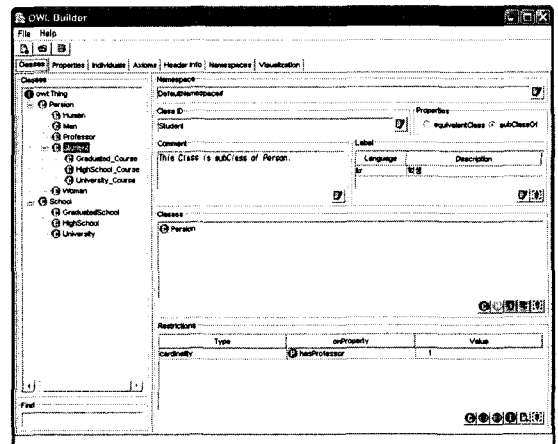
<그림 3> Property의 내부 데이터 모델

### 3.4 GUI Interface

OWL은 온톨로지 저작을 위한 풍부한 표현력을 제공하고 있기 때문에, 본 연구에서는 OWL 명세서 분석을 통해 이를 효과적으로 편집할 수 있도록 Class, Property, Individual, Axiom, Header, Namespace 6개의 모듈로 나누어 설계 및 구현하였다. Class, Property, Individual 모듈은 기본적인 OWL 요소를 정의하며, Axiom 모듈에서는 하나의 Class나 Individual 상에서 정의할 수 없는 Disjoint-With, AllDifferent를 정의하도록 하였다. Header에서는 OWL 온톨로지의 Header 정보인 versionInfo, priorVersion, backCompatibleWith, inCompatibleWith, import를 정의한다. Namespace에서는 온톨로지 전체에서 사용하는 Namespace들을 편집할 수 있도록 하였다. <그림 4>는 각 인터페이스별 기능을 나타내며, <그림 5>는 Class 편집을 위한 화면구성을 보여준다.

Annotation 정의 - ID, Label, Comment subClassOf equivalentClass Restriction 정의 - Cardinality - Property Restriction Class Combination - intersectionOf - unionOf - complementOf	Annotation 정의 - ID, Label, Comment subPropertyOf inverseProperty Domain Range Property 특성 정의 - Functional - inverseFunctional - Symmetric - Transitive	Annotation 정의 - ID, Label, Comment Individual간 관계 - sameIndividualAs - differentFrom Relation 정의
AllDifferent disjointWith	versionInfo priorVersion backwardCompatibleWith inCompatibleWith import	Namespace 추가/삭제 Default Namespace 설정

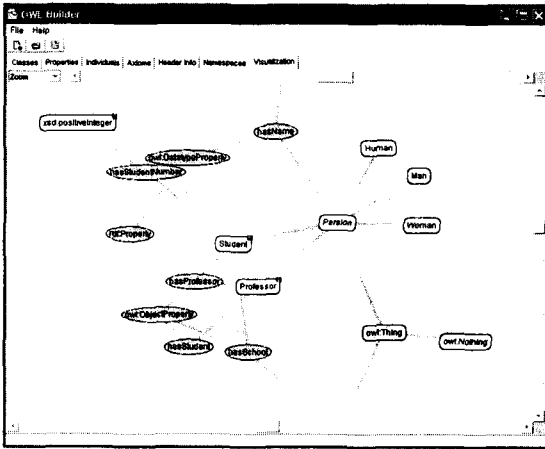
<그림 4> 인터페이스 모듈별 기능



<그림 5> Class 편집 화면

### 3.5 Visualizer

Visualizer 모듈은 온톨로지에 표현된 정보를 Node & Edge Diagram 형태로 시각화하여 표시하고, 노드를 선택하면서 정보를 탐색할 수 있도록 하는 시각화 및 탐색 모듈이다. Class, Property, Individual에 따라 Node의 모양이 서로 다르게 표현되도록 하였으며 Edge에 노드와 노드간의 Relation을 표시할 수 있도록 하였다. <그림 6>은 Visualizer의 탐색 화면을 보여준다.



<그림 6> Visualizer 탐색 화면

### 3.6 Serializer

Serializer는 편집된 내부 데이터 모델을 다시 텍스트 형태의 OWL 문서로 변환한다. Serializer에는 내부 모델의 각 요소들을 XML DOM 모델로 변환하기 위한 API를 제공하며, 모든 요소들이 DOM 모델로 변환되면 최종적으로 텍스트 형식의 OWL 문서를 반환한다.

## 4. 결 론

현재 국외에서 온톨로지를 처리하기 위해 많은 노력들이 진행 중에 있으며, 시맨틱 웹을 위한 많은 프로젝트와 제품을 개발 중이다. 하지만 W3C에서 온톨로지 표현 언어로 제정하고 있는 OWL 기반의 온톨로지를 효율적으로 작성하고 편집할 수 있는 저작도구는 아직까지 전무한 실정이다. 이에 본 연구에서는 OWL 기반의 온톨로지 문서를 시각적인 GUI 인터페이스를 통해서 효율적으로 생성 및 편집할 수 있고 온톨로지의 정보를 시각적으로 표시하여 탐색할 수 있는 저작 도구를 설계 및 구현하였다.

한편 여러 자원으로 분산 저장되어 있는 OWL 온톨로지들을 하나로 통합할 수 있는 Merge 기능의 구현과, 다른 언어로 작성된 온톨로지를 OWL로 변환하여 통합할 수 있는 방안에 대한 연구가 필요하다.

### 참고문헌

- [1] Berners-Lee, T., Hendler, J. and Lassila, O., "The Semantic Web", Scientific American, 2001.
- [2] W3C, Extensible Markup Language (XML), <http://www.w3.org/XML/>.
- [3] W3C, Resource Description Framework (RDF), <http://www.w3.org/RDF/>.
- [4] W3C, RDF Vocabulary Description Language (RDFS), <http://www.w3.org/TR/rdf-schema/>.
- [5] Deborah L. McGuinness and Frank van Harmelen, "OWL Web Ontology Language Overview," 2003, <http://www.w3.org/TR/owl-features/>.
- [6] Michael K. Smith, et al., "OWL Web Ontology Language Guide," 2003, <http://www.w3.org/TR/owl-guide/>.
- [7] Mike Dean and Guus Schreiber, "OWL Web Ontology Language Reference," 2003, <http://www.w3.org/TR/owl-ref/>.
- [8] Peter F. Patel-Schneider, et al., "OWL Web Ontology Language Semantics and Abstract Syntax," 2003, <http://www.w3.org/TR/owl-semantics/>.
- [9] Protégé-2000, <http://protege.stanford.edu/index.html/>.
- [10] OilEd, <http://oiled.man.ac.uk/>.
- [11] OntoEdit, [http://www.ontoprise.de/home\\_en/](http://www.ontoprise.de/home_en/).
- [12] Frank van Harmelen, et al., "Reference description of the DAML+OIL ontology markup language," 2001, <http://www.daml.org/2001/03/reference>.