

온톨로지 저장소를 이용한 설계 지식의 저장과 회수

지경환*, 양정진 (가톨릭대 컴퓨터정보공학부)

Storing and Querying of Design Knowledge Using Ontology Repository

KyengWhan Jee*, JungJin Yang (Catholic Univ. of Korea. Dept., CSIE)

ABSTRACT

The requirement to reuse a design knowledge have been enlarged with the automation of a design system. A design knowledge gives logical and technical meanings to design data of a problem area. The representation of the knowledge is distributed and developed independently. For this reason, we need a general methodology with a semantic interoperability of design knowledge. In this paper, we accept previous requirements by using semantic query system with ontology repository.

Key Words : Design Repository, Ontology Repository, Semantic Query

1. 서론

설계 시스템들의 자동화와 더불어 설계 지식의 재사용성의 요구가 증대되고 있다. 설계 자동화 시스템은 주어진 문제에 따라 독립적으로 발전하고 있기 때문에 설계 지식의 재사용을 위해선 설계 데이터의 의미적 상호운용성을 보장할 필요가 있다. DBMS 시스템은 재사용 가능한 데이터와 어플리케이션의 독립을 이루었지만 데이터의 의미 자체는 여전히 DB Schema에 Ad-Hoc되어 있기 때문에 설계 지식은 점점 더 분산되고 독립적으로 발전하여 잠재적인 가치 회수에 제약을 가진다. 본고에서는 설계 시스템의 의미적 상호운용성을 보장하기 위한 온톨로지 기반의 설계 지식 표현과 응용 계층에서 지식을 회수하기 하기 위해 의미적 질의문을 통하여 설계 지식을 회수하는 방법론을 제시한다.

2. 관련연구

W3C에서는 데이터를 위한 의미(Meta-data)를 기술하기 위한 표준으로 XML을 권고한다. XML은 기계가 읽을 수 있는 구조로 정의 되었고 XML Schema와 더불어 성분(Element)과 속성(Attribute)으로 메타 데이터 어휘를 기술하고 개체 수 제약(Cardinality Restriction)을 이용하여 구문을 기술한다. RDF[1]는 XML을 기반으로 유, 무형의 자원을 주어 술어 목적어 기반의 트리플 형태로 선언하며 URI와 더불어 유일한 메타데이터를 기술하기 위한 기반구조로 사용된다. RDF Schema는 RDF의 용어, 역할의 개념을 정의하는 경량의 온톨로지를 정의하기 위한 메타언어로 사용되며 응용계층은 각 개념들이 선언된 온톨로

지를 명시하여 의미적 상호운용성을 보장 받는다. OWL[2]은 W3C에서 권고하는 온톨로지 언어이며 서술 논리(Description Logic)를 기술하여 논리적인 표현력과 합리적인 시스템을 위한 자료구조로 사용될 수 있다.

온톨로지의 메모리 상주 모델 처리 방식은 방대한 설계 지식을 수용하기에 부적절하기 때문에 온톨로지 저장소의 사용을 요구하고 논리적으로 표현된 지식의 일관성과 추론을 위해 추론엔진의 사용이 요구 된다. 온톨로지 저장소의 기능적 요구사항과 성능적인 요구사항을 충족시키기 위해 KAON, DLDB 등이 연구 되었으며 본고에서는 두 요구사항을 모두 수용하는 Minerva[3] 온톨로지 저장소를 이용하여 표현된 설계 지식을 저장하고 회수 한다.

3. 온톨로지 기반의 설계 지식 표현

설계 지식을 온톨로지 표현하기 위해선 서술논리로 표현할 수 있는 논리의 범위와 일관성을 분별하고 온톨로지 저장능력을 가진 온톨로지 전문가와 설계 지식을 모델링하고 표현된 논리의 합리성을 분별하기 위한 해당분야의 전문가의 공동 작업이 필요하다. Fig. 1은 설계 지식의 온톨로지 표현의 예로 개념간의 일반화 관계를 명시 하고 있다. 가장 세분화된 개념인 Fan_Gate개념(Concept)의 개체(Individual)는 Gate, Mold_Part, Object 개념의 개체가 될 수 있다. 만약 Object 개념이 hasStructure 술어의 정의역(Domain)이라면 Object 용어와 개념과 Object 개념의 지식 개념들도 hasStructure 역할의 정의역이 된다. 온톨로지는 서술 논리를 이용하여 이 지식을 명시

하고 있으며 추론엔진을 통해 논리적 지식을 기반으로 모든 지식 표현들을 확장한다.

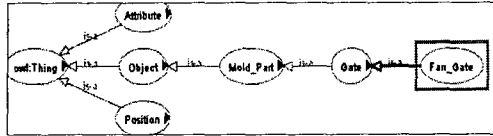


Fig. 1 Design concepts

4. 설계 지식의 저장과 회수

온톨로지로 작성된 설계 지식을 온톨로지 저장소에 저장하기 위해 Minerva 온톨로지 저장소와 제공된 API를 사용하여 저장한다. 논리적 설계 지식의 확장과 일관성을 검사하기 위해 온톨로지의 적재, 수정시점과 의미적 질의시점에 추론을 수행할 수 있으며 Minerva는 적재, 수정 시에 추론을 수행하므로 저장 공간과 추론시간을 교환하는 정책을 사용한다.

온톨로지를 저장하기 위한 DBMS로 IBM DB2를 사용하고 표현된 설계 지식을 관계형 데이터베이스에 저장하기 위해서 Minerva 온톨로지 저장소에서 제공하는 DB Schema와 적재 모듈을 사용하였다. Fig. 2는 제공된 API를 이용해 온톨로지를 적재하고 추론을 수행하는 코드이다.

```

public class Demo1
//load owl documents into ontology repository
public static void loadOWLDocuments() {
//generate an ontology store manager
OntologyStoreManager minerva = new OntologyStoreManager();
//delete an ontology store
minerva.deleteOntologyStore("DesignOntology");
//new an ontology store
minerva.newOntologyStore("DesignOntology");
//select an ontology store
minerva.selectOntologyStore("DesignOntology");
//add an owl documents into the selected ontology store
minerva.addOWLDocument("C:/workspace/minerva/ontology.owl", false);
//do inference
minerva.commit();
}
public static void main(String[] args) {
//configure log4j
BasicConfigurator.configure();
//configure minerva
Config.setConfigFile("C:/workspace/minerva/minerva-05.cfg");
Config.install();
loadODMModel();
}
}

```

Fig. 2 Load and inference of Design Repository

온톨로지 저장소로부터 설계 지식을 회수하기 위해 의미적 질의를 수행한다. 의미적 질의 언어는 질의문 자체에 필요로 하는 데이터의 의미를 부여해 질의 할 수 있는 언어이며 본고에서는 RDF 트리플 형태로 의미를 부여하는 SPARQL[4]을 사용하였다. Minerva 온톨로지 저장소는 의미적 질의 언어를 파싱하고 응답할 수 있는 질의응답 엔진을 가지고 있으며 온톨로지 적재 시 추론 되지 않았던 부분을 추론하며 질의에 응답한다. Table 1의 Q.1은 Fan_Gate의 일반화된 개념을 질의하는 질의문이고 Q. 2는 FanGate 개념의 FanGate_개체가 속한 개체들을 질의하는 질의문이다. Fig. 3은 질의 결과이며 Table 1의 Q. 1과 Q. 2가 같은 결과를 반환하는 것을 알 수 있다.

Q. 1 select * where
(Design:Fan_Gate rdfs:subClassOf ?y)
Q. 2 select * where
(Design:Fan_Gate_ rdf:type ?y)

Table 1 Query using SPARQL

```

Console X
<terminated> QueryDemo2 [Java Application] C:\jdk1.4.2_10\bin\javaw.exe
D [main] INFO com.ibm.cri.minerva.db.DAO - Get connection
y
http://www.w3.org/2002/07/owl#Thing
http://www.owl-ontologies.com/unnamed.owl#Mold_Part
http://www.owl-ontologies.com/unnamed.owl#Object
http://www.owl-ontologies.com/unnamed.owl#Gate
http://www.owl-ontologies.com/unnamed.owl#Fan_Gate
0

```

Fig. 3 Query Result : general concepts of FanGate concepts, concepts (Types) of FanGate individual

설계 지식 온톨로지는 설계 지식을 논리적인 개념 수준으로 표현하고 응용 계층에 의미적인 상호운용성을 보장하며 합리적인 설계 시스템을 위한 기반 구조가 된다.

5. 결론

설계 시스템간의 설계 지식의 재사용성을 높이기 위해서는 의미적 상호운용성을 보장할 수 있어야한다. 본고에서는 응용 계층에서 사용될 설계 지식의 의미적 상호운용성을 보장하기 위해 온톨로지를 사용하였고 온톨로지 저장소와 응용계층과의 일반화된 인터페이스의 사용을 보였다. 응용계층은 온톨로지 저장소를 이용하여 일반화된 방법론을 통해 설계 지식의 Know-Where의 제약을 넘어 Know-How를 질의할 수 있는 기반 구조를 가지게 된다.

후기

본 연구는 산업자원부 핵심연구개발사업 "미세형상 설계 지능화 기술 개발" 과제 수행의 일환으로 이루어진 것임을 밝히며, 지원 기관에 감사드립니다.

참고문헌

1. W3C, RDF Primer W3C Recommendation 10 February, <http://www.w3.org/TR/2004/REC-rdf-primer-20040210>, 2004.
2. W3C, OWL Web Ontology Language Overview W3C Recommendation 10 February, <http://www.w3.org/TR/2004/REC-owl-features-20040210>, 2004.
3. IBM, IBM Integrated Ontology Development Toolkit March 15, <http://www.alphaworks.ibm.com/tech/semanticstk>, 2006.
4. W3C, SPARQL Query Language for RDF 20 February, <http://www.w3.org/TR/rdf-sparql-query>, 2005.