

## OWL-DL 기반의 대용량 ABox 추론 기법

서은석<sup>o</sup> 최용준 박영택

송실대학교 컴퓨터학과

erin214@ailab.ssu.ac.kr<sup>o</sup>, ai@ailab.ssu.ac.kr, park@computing.ssu.ac.kr

### A Method for Supporting Description Logic SHIQ(D) Reasoning over Large ABox

Eun-Seok Seo<sup>o</sup> Yong-Joon Choi Young-Tack Park

Department of Computing, Soongsil University

#### 요약

현존하는 추론 엔진들은 대부분 Tableau 알고리즘 기반의 TBox의 최적화를 위한 연구를 진행하였다. 하지만 현실에서 대용량의 ABox를 추론하기 위한 유효한 시간 내에 결정 가능성을 보장하지 못한다. 따라서 실용성 있는 추론 엔진 효율을 위해서는 대용량 데이터를 가지는 ABox를 위한 최적화된 추론 기법이 필요하다.

본 논문에서는 OWL-DL 기반의 온톨로지(Ontology)를 데이터로그(Datalog)와 같은 규칙(Rule) 형태로 변환하여 관계형 데이터베이스와 같은 저장 시스템과 연동하기 위한 방법을 이용한다. 최종적으로 실세계의 환경에서의 데이터타입 속성(Datatype Property)이 포함된 SHIQ(D) 구성의 실용적인 추론 시스템을 수행하고자 한다. 따라서 OWL이 가지는 공리(Axiom)를 이용하여 데이터타입 속성이 포함된 규칙을 적용한 추론 방법에 대해서 제안하였다.

#### 1. 서론

현존하는 대부분의 추론엔진들은 OWL-DL기반의 온톨로지(Ontology)를 이용한 추론을 수행한다. OWL-DL은 W3C표준의 표현력이 서로 다른 세 가지의 하위 언어들 중 하나이며 서술논리(Description Logic)[1]를 기반으로 한다. 서술논리의 구조는 개념(Concept)을 표현하기 위하여 온톨로지의 스키마 구조를 가지는 TBox와 개념에 포함되는 직접적인 실세계의 데이터 개체(Individual)들을 표현하고 있는 ABox로 구성된다. 이러한 구조를 바탕으로 하여 추론엔진은 명시적(explicit)으로 표현된 지식으로부터 암묵적(Implicit)인 지식을 도출할 수 있다. 추론엔진의 추론 처리 과정에서 발생하는 시간 비용을 줄이기 위하여 다양한 최적화 기법이 연구 및 제시되었으며 추론엔진들은 TBox를 통한 최적화 기법에 초점을 두었다. 하지만 최근에 시멘틱 웹(Semantic Web)에 대한 연구가 진행되어 오면서 실용성 있는 대용량의 ABox 처리를 위하여 기존 추론엔진들이 수행해 오던 Tableau 알고리즘 기반의 TBox 최적화를 통한 추론 수행만으로는 유효한 시간 이내에 지식을 도출하는데 한계가 있다. 따라서 기존 추론 엔진과는 다른 추론 방안이 필요하며 궁극적으로 대용량 ABox를 위한 최적화 기법을 통한 실세계의 응용 어플리케이션에서 효율성 있는 연구가 필요하다.

이러한 문제점을 해결하기 위하여 Tableau 방식이 아닌 서술 논리 프로그램(Description Logic Program) 방식을 사용할 수 있다. 서술 논리 프로그램 방식은 기존에 잘 알려진 사실(Fact)과 규칙의 집합을 다루는 논리 프로그램과 서술 논리를 결합한 형태를 말한다. 이러한 접근 방법은 OWL-DL 기반의 온톨로지를 선언적 데이터 로그(Disjunctive Datalog)와 같은 규칙 언어와 결합하기

위한 방법이다. 데이터로그는 관계형 데이터베이스(RDBMS)에서 추론 규칙을 직접 표현할 수 있으며 규칙에 만족하는 결과에 대하여 빠른 시간 내에 찾을 수 있도록 한다. 최종적으로 온톨로지의 스키마 구조인 TBox를 규칙 형태로 변환하여 비교흡수(Resolution)방식을 수행하며 실질적 데이터인 ABox를 관계형 데이터베이스와 같은 영구적인 저장소에 분리하여 대용량 데이터에 대한 실용적인 추론 수행이 이루어질 수 있도록 하기 위한 방법이다. 하지만 이러한 규칙 형태로의 변환과정에서 수치적 연산을 위한 데이터타입 속성이 포함된 SHIQ(D) 구성의 온톨로지는 비교흡수기반의 정리 증명기(Theorem Prover)를 위하여 상당히 복잡하며 정교한 기술이 요구된다. 예를 들어 수치형 제약(Number Restriction)을 위한 처리를 위해서는 동치성(Equilty)에 대한 추론이 요구되며 복잡한 계산이 필요로 하다. 따라서 데이터타입 속성이 포함된 온톨로지를 처리하여 최종적으로 추론을 수행 위하여 새로운 접근 방식을 필요로 한다.

#### 2. 관련연구

##### 2.1 Tableau 알고리즘 기반의 접근 방식

Tableau 알고리즘 기반의 대표적 추론기인 RACER(Renamed ABox and Concept Expression Reasoner)는 SHIQ(D) 범위의 서술논리 문법의 추론을 수행하기 위하여 개발된 추론 엔진이다. RACER의 장점으로는 Tableau 알고리즘 기반의 TBox 최적화를 통하여 현존하는 대부분의 추론 엔진보다 TBox 추론에서는 뛰어난 성능을 나타내고 있다. 또한 각 개체가 가지는 수치적 형태의 추론을 위한 데이터타입 속성에 대하여

제한적으로 추론을 수행 할 수 있다. 하지만 RACER가 가지는 단점으로는 대용량 ABox에서 Tableau 알고리즘의 한계를 나타내고 있다. 초기 시스템에 대한 설계 당시 ABox에 대한 실용적인 추론 수행을 목표로 두었지만 Tableau 알고리즘의 추론 수행방식으로 인하여 대용량의 ABox 추론에서는 KAON2와 같은 추론 엔진에 비해 효율성이 크게 떨어지는 것을 알 수 있다.

### 2.2 선언적 데이터로그 접근 방식

KAON2는 OWL-DL로 표현된 대용량의 ABox에 대하여 기존의 추론엔진이 사용하고 있는 Tableau 알고리즘 방식이 아닌 서술논리 및 선언적 데이터로그를 통한 데이터베이스에서의 추론을 모두 처리할 수 있도록 설계된 하이브리드(Hybrid) 방식의 추론 엔진이다. KAON2의 주요 장점으로는 작은 사이즈의 TBox와 대용량의 ABox가 서술논리로 표현 되었을 경우 매우 효율적인 추론을 수행하는 것을 나타내었다. 하지만 KAON2 단점으로 데이터베이스의 처리를 위하여 OWL-DL 기반의 FOL 구문에 대하여 Horn-Clause 형태로의 변환 시에 Function Symbols의 제거로 인한 잘 정의된 의미를 지원하지 않는다. 또한 OWL의 데이터타입 속성과 같은 구체화된 도메인에 대한 질의 수행 시 결과를 도출할 수 없는 단점이 있다.

### 3. 대용량 ABox에서 수치형 데이터타입 속성 지원 방안

본 논문에서는 OWL-DL 기반의 온톨로지를 선언적 데이터로그와 같은 규칙 언어로의 변환 시 데이터타입 속성에 대한 지원 방법을 제안하였다. 데이터로그를 사용하여 추론 수행을 하기 위해서는 OWL-DL 기반의 온톨로지 대하여 혼 로직(Horn-logic)방식에서의 변환과정이 필요하다. 데이터로그는 관계형 데이터베이스(RDBMS)에서 규칙을 적용한 질의를 수행하기 위하여 프롤로그와 같이 혼 로직 스타일의 규칙의 적용을 가능하게 하여 일반적으로 사용되는 데이터베이스에서 처리할 수 없는 트랜지티브(Transitive) 형태에 대한 검색을 가능하도록 한다. OWL-DL로 표현되는 지식정보에 대하여 데이터베이스로 표현하기 위한 혼 로직 형태로의 변환과정에서 각 개념에 포함될 수 있는 개체의 연관 정보가 손실 될 수 있다. 예를 들어 'SportsMan이면 BaseballPlayer 또는 SoccerPlayer이다'라고 OWL-DL 형태로 표현 되어있는 지식의 경우 혼 로직의 표현이 불가능 하다. 따라서 혼 로직에서 처리 가능한 형태로 의미적 정보를 유지하도록 변환 하여야 한다.

데이터타입이란 개념이 가지는 구체화된 도메인(Concrete domain)을 의미한다. OWL-DL에서 데이터타입은 XML 스키마로 표현 될 수 있다. 이러한 속성은 String, Integer 등으로 각 개체가 가지는 속성 값을 정의 할 수 있다. 예를 들어 [표 1]에서는 각 개체인 Tom, John, Michie에 대하여 이름 및 나이에 대한 속성을 표현하기 위해서 String 데이터타입으로 이름을 정의 하였으며 Int 데이터타입으로는 개체의 hasAge를 나타내

도록 정의하였다.

표 1 데이터타입 속성의 예

```
<BaseballPlayer rdf:ID="Tom">
  <name rdf:datatype="http://www.w3.org/2001/XMLSchema#string">
    Tom</name>
  <hasAge rdf:datatype="http://www.w3.org/2001/XMLSchema#int">33</age>
</BaseballPlayer>
<BaseballPlayer rdf:ID="John">
  <hasAge rdf:datatype="http://www.w3.org/2001/XMLSchema#int">22</age>
  <name rdf:datatype="http://www.w3.org/2001/XMLSchema#string">
    John</name>
</BaseballPlayer>
<SoccerPlayer rdf:ID="Michel">
  <name rdf:datatype="http://www.w3.org/2001/XMLSchema#string">
    Michel</name>
  <hasAge rdf:datatype="http://www.w3.org/2001/XMLSchema#int">22</age>
</SoccerPlayer>
```

하지만 이러한 규칙 형태로의 변환과정에서 수치적 연산을 위한 데이터타입이 포함된 SHIQ(D) 구성의 온톨로지는 비교출수기반의 정리 증명기를 위하여 상당히 복잡하며 정교한 기술이 요구된다. 예를 들어 수치형 제약(Number Restriction)을 위한 처리를 위해서는 동치성에 대한 추론이 요구되며 복잡한 계산이 필요로 하다. 'BaseballPlayer또는 SoccerPlayer에서 age가 20이상인 개체를 추출하라'와 같은 데이터타입 추론을 지원하기 위하여 본 논문에서는 데이터로그 언어로의 변환과정에서 OWL syntax의 rdf:datatype이 포함되는 공리에 대한 추출 과정을 수행한다.

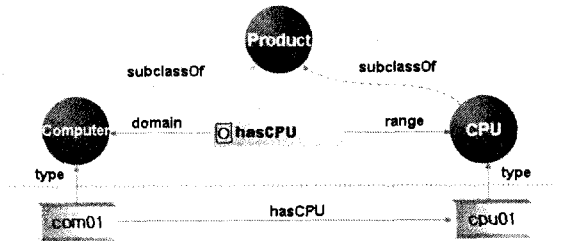


그림 1 온톨로지 모델링

OWL-DL 문법으로 구성된 지식 베이스에 대하여 데이터로그 규칙 형태로의 변환을 위하여 [그림 1]에서는 개념과 개념간의 관계 및 개념에 포함되는 개체에 대한 관계를 모델링하였다. Product, Computer, CPU 세 개의 클래스를 정의하고 Computer와 CPU는 Product의 subClass로서 정의하였다. 또한 Computer는 hasCPU라는 속성(Property)의 도메인에 해당되며 CPU는 같은 속성의 레인지에 해당된다. 실제 개체를 묘사하기 위해서 'com01'라는 개체는 타입이 Computer 클래스에 포함되며 'com02'라는 개체는 CPU 클래스 타입을 가지고 있다. 따라서 'com01'과 'com02'는 서로 hasCPU 속성을 가진 관계로 나타낼 수 있다. 각 클래스들에 대한 계층 관계는 rdfs:subClassOf라는 OWL 공리를 통하여 표현 하였으며 클래스와 클래스 간에 관계를 정의한 속성은

owl:ObjectProperty와 rdfs:domain, rdfs:range로 표현하였다. 또한 실제 개체에 대해서는 rdf:ID로서 표현될 수 있다. OWL-DL 문법으로 작성된 온톨로지에 대해서 공리가 가지는 의미적 정보를 혼절 형태로 나타낼 수 있다.

#### 4. 시스템 구조

본 논문에서 제안하는 방식에 대하여 [그림 2]에서 시스템 구조를 묘사하였다.

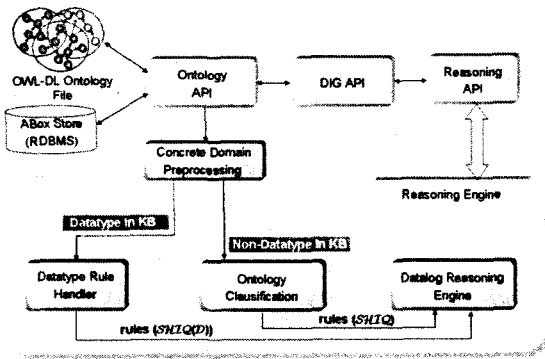


그림 2 시스템 구조

Protege와 같은 온톨로지 모델링 툴을 이용하여 온톨로지를 모델링 하면 모델링된 온톨로지 파일을 Concrete Domain Preprocessing 모듈에 전달한다. Concrete Domain Preprocessing은 데이터타입이 포함되는 지식베이스에 대한 분리작업을 수행한다. 이러한 분리작업을 수행하는 이유는 최종적으로 데이터타입이 포함되어진 온톨로지의 추론이 수행되는지의 여부의 실행을 위하여 SHIQ 구성의 지식베이스는 기존 KAON2 알고리즘을 이용하여 데이터타입이 포함되는 지식베이스는 본 논문에서 제시하는 데이터타입 처리(Datatype Handling) 알고리즘을 이용한 처리를 위해서이다. Ontology Classification 모듈에서는 SHIQ 구성의 OWL-DL의 스키마 구조를 가지는 TBox에 대하여 기존 KAON2 알고리즘을 이용하여 FOL 문법으로 변환하며 최종적으로 데이터 로그 엔진에서 사용되는 규칙 처리를 위한 혼절(Horn Clauses) 형태로 변환된다.

본 시스템에서의 데이터타입에 대한 처리 지원 방안을 위하여 데이터타입 규칙 처리기(Datatype Rule Handler)는 데이터타입이 포함되는 TBox에 대해서 데이터 로그 엔진에 사용되는 규칙 언어 형태로 변환된다. 최종적으로 Ontology Classification 모듈과 데이터타입 규칙 처리기 모듈에서 추출되는 지식베이스는 데이터타입이 포함되는 SHIQ(D)의 구성을 가지게 된다.

본 논문에서는 트리플로서 표현된 OWL의 의미적 정보를 이용한다. 데이터타입이 가지게 되는 트리플 정보에 포함된 owl 공리의 구성을 이용하여 궁극적으로 데이터타입이 포함된 OWL 온톨로지에 대하여 규칙 언어 형태

로써 변환할 수 있도록 하였다.

[그림 3]에서는 데이터 타입 지원을 위한 트리플 구조에 대하여 설명하고 있다. 그림의 예에서와 같이 'HighEnd 클래스가 가지는 속성인 hasPrice는 2000 이상이다.' 라는 정의에 대하여 서술논리로 HighEnd드크 hasPrice.≥2000 과 같이 표현될 수 있다. OWL-DL로서 표현할 경우 rdf:ID를 이용하여 HighEnd 클래스를 정의한다. HighEnd 클래스가 가지는 hasPrice라는 속성 정보에 대하여 owl:Restriction 공리를 통하여 hasPrice는 owl:DatatypeProperty이며 owl:minCardinality 공리로서 최소 2000 이상으로 정의될 수 있다.

모델링된 온톨로지를 데이터타입 규칙처리기로 전달하기 위하여 전처리 통하여 트리플 구조로 변환한다. 트리플 구조로 변환된 온톨로지는 [그림 3]과 같이 클래스와 속성에 대한 타입과 임의의 클래스가 가지는 서브클래스 정보들을 트리플 구조로서 알 수 있다.

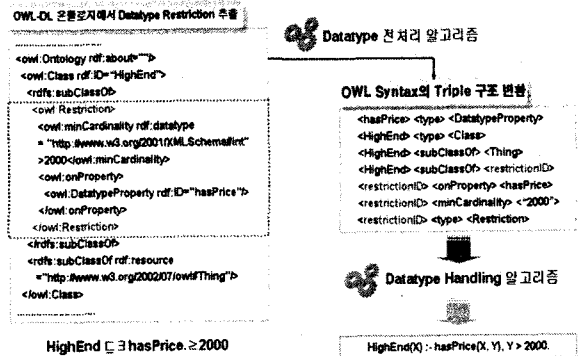


그림 3 OWL 트리플 구조 적용

예를 들어 <HighEnd><type><Class>는 'HighEnd 클래스의 타입은 클래스이다.'라는 사실을 의미하며 <HighPrice><type> <DatatypeProperty>는 'hasPrice의 타입은 데이터타입 속성이다.'라는 사실을 의미한다. 임의의 클래스에 제약 정보를 부여할 경우 그림에서와 같이 restrictionID가 트리플 구조에서 생성될 수 있다. 예를 들어 'HighEnd 클래스는 hasPrice로서 최소 2000이상의 값을 갖는다' 라는 제약 정보를 부여할 경우 HighEnd 클래스는 익명의 클래스(restrictionID)에 하위클래스이고 이러한 <restrictionID>의 값을 갖는 익명 클래스는 <onProperty><minCardinality> <type>등의 OWL이 가지는 공리 정보를 통하여 임의의 클래스에 대한 제약 정보를 알 수 있다.

본 시스템에서는 데이터타입 전처리 모듈에서 트리플 형식으로 추출된 지식베이스를 데이터로그 규칙 엔진에서 사용하도록 하기 위한 규칙 형태로의 변환과정을 수행한다. 이러한 작업 수행을 위하여 변환된 트리플 구조에서 데이터타입이 가지는 제약 아이디(Restriction ID)를 추출할 수 있다. 본 논문에서는 이러한 제약 아이디가 가지는 관계를 통하여 데이터로그 형태의 룰을 구성하였다. [그림 4]에서는 지식베이스에서 데이터타입을 위한 규칙의 구성을 위한 알고리즘의 예를 나타내고 있다. 데

이더타입 규칙 처리기에서 처음으로 수행하는 작업은 데이터타입 전처리 모듈로부터 전달 받은 트리플 형태의 지식베이스에서 데이터타입과 관련된 정보가 있는지를 검사한다. 이러한 검사수행을 위해서 최소치 값이나 최대치 값에 대한 데이터타입 처리를 위하여 트리플 지식베이스 상에 이러한 정보를 나타내고 있는 <minCardinality> 또는 <maxCardinality>가 존재하는지에 대하여 체크한다. 예를 들어 최소치 값을 위한 <minCardinality>가 트리플 지식 베이스 내에 존재한다면 <minCardinality>를 가지는 제약 아이디인 <restrictionID>를 추출하며 추출된 <restrictionID>를 통하여 임의의 클래스가 가지는 모든 조건이 만족되면 제약 아이디를 기준으로 서브클래스 정보는 데이터로그 규칙의 Head 부분으로 나머지 제약 정보에 대해서는 규칙의 Body 부분으로 구성되어 지게 된다.

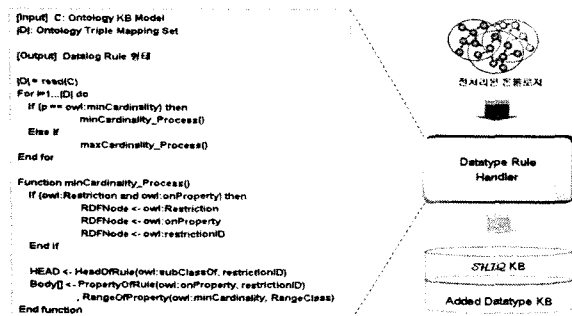


그림 4 SHIQ(D) 구성의 데이터로그 규칙 형태로 변환

최종적으로 변환되어진 데이터로그 규칙은 데이터타입이 포함되지 않은 지식베이스와 결합함으로써 완전한 SHIQ(D) 구성의 데이터로그 규칙으로서 구성된다.

5. 실험

데이터타입 문법이 포함되어져 있는 컴퓨터 도메인의 동일한 온톨로지에 대하여 본 시스템과 KAON2에서 처리해 보았다. 실험을 위하여 시스템의 각 단계별 과정을 나타낼 수 있는 시뮬레이터를 구현하였다.

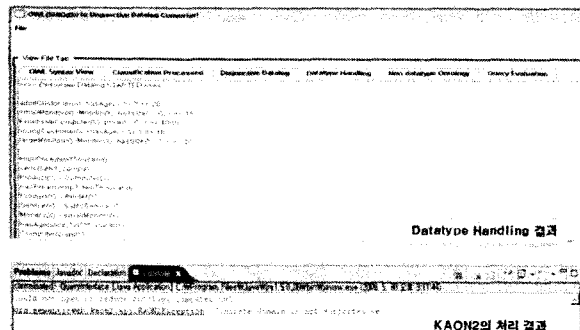


그림 5 SHIQ(D) 구성의 데이터로그 규칙 형태로 변환

[그림 5]에서와 같이 본 시스템에서 제안한 처리 결과로 데이터타입 속성이 포함된 규칙과 사실에 대하여 나타낼 수 있었지만 KAON2에서는 수행 할 수 없었다.

데이터타입 속성이 포함되어진 온톨로지를 데이터로그의 규칙 형태로 변환하여 올바른 질의 여부가 수행되는지를 평가하기 위하여 다음과 같은 질의를 수행하여 보았다. 아래의 데이터타입 규칙과 사실이 있는 경우 smallMonitor 클래스가 가지는 모든 개체를 검사하고자 하였다.

```

데이터타입 규칙 1: smallMonitor(Monitor):-
    monitor(Monitor),

    hasScreenSize(Monitor, Size),
    Size =<15.
    
```

```

데이터타입 규칙 2: largeMonitor(Monitor):-
    monitor(Monitor),
    hasScreenSize(Monitor, Size),
    Size =<20.
    
```

```

사실 : monitor(lg001), monitor(lg001),
    monitor(lg001), monitor(lg001),
    hasScreenSize(lg001, 14),
    hasScreenSize(lg001, 12)
    hasScreenSize(samsung001, 29),
    hasScreenSize(samsung001, 20)
    
```

질의 : smallMonitor(X).

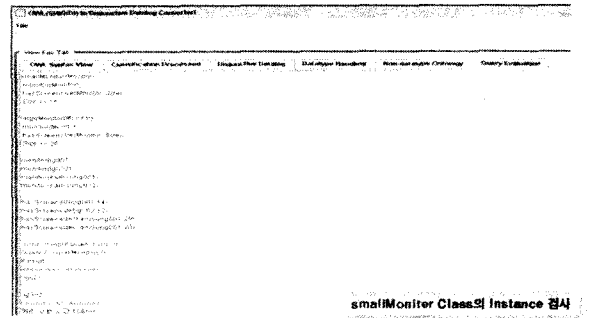


그림 6 질의 평가

[그림 6]의 질의 예에서 smallMonitor 클래스가 가지는 개체에 대한 검사 시에 규칙1을 통하여 monitor 클래스의 개체들 중 lg001과 lg002가 검색되는 것을 확인할 수 있었다.

6. 결론 및 향후 연구

본 논문에서는 수치형 연산을 위한 데이터타입 속성과 같은 구체화된 도메인이 포함되는 SHIQ(D) 구성의 대용량 ABox에 대한 처리 방안에 대하여 제안하였다. 논문에서 제안한 방법을 통하여 데이터타입이 포함되는 OWL-DL 문법의 온톨로지를 선언적 데이터로그의 규칙

형태로 변환하여 실용성 있는 추론이 가능하도록 하였다. 향후 연구로서는 좀 더 확장된 범위에 대한 데이터 타입 처리 방법에 대한 연구를 진행 하고자 한다.

#### 참고문헌

- [1]. F. Baader et al., 2003. "The Description Logic Handbook: Theory, Implementation and Applications." Cambridge University Press.
- [2]. B. Motik and U. Sattler. "Practical DL Reasoning over Large ABoxes with KAON2. Submitted for publication", 2005
- [3]. U. Hustadt and B. Motik and U. Sattler. "Reasoning for Description Logics around SHIQ in a Resolution Framework." 2005
- [4]. Ian Horrocks and Ulrike Sattler. A tableaux decision procedure for SHOIQ. In Proc. of the 19th Int. Joint Conf. on Artificial Intelligence (IJCAI 2005), 2005.
- [5]. I. Horrocks and U. Sattler. Optimised reasoning for SHIQ. In Proc. of the 15th Eur. Conf. on Artificial Intelligence (ECAI 2002), pp. 277-281, 2002.
- [6]. V. Haarslev and R. Moller. RACER User's Guide and Reference Manual Version1.9.
- [7] <http://www.mindswap.org/2003/pellet/>
- [8]. Dmitry Tsarkov and Ian Horrocks. Optimised classification for taxonomic knowledge bases. In Proc. of the 2005 Description Logic Workshop
- [9]. G. Antoniou and F. van Harmelen, 2004 "A Semantic Web Primer"
- [10]. Horrocks and U. Sattler. "Ontology reasoning in the SHOQ(D) description logic". In B. Nebel, editor, Proc. of IJCAI-01, pages 199-204. Morgan Kaufmann, 2001.
- [11] <http://www.w3.org/TR/owl-semantic/>
- [12] <http://www.w3.org/TR/owl-xmlsyntax/>

#### Acknowledgement

본 논문은 정통부 및 정보통신연구진흥원의 정보통신선  
도기반기술개발사업의 연구결과로 수행되었습니다.