

# 효과적인 시맨틱 웹의 구현을 위한 마크업 언어

유명환<sup>0</sup>, 정희준<sup>\*</sup>, 이강찬<sup>\*\*</sup>, 김성환<sup>\*\*</sup>, 민재홍<sup>\*\*</sup>, 정인정<sup>\*</sup>

<sup>\*</sup>고려대학교 전산학과 {myong<sup>0</sup>, joonny96, chung}@korea.ac.kr

<sup>\*\*</sup> 한국전자통신연구원 표준연구센터 {chan, sh-kim, jhmin}@etri.re.kr

## The Markup Language for Implementation of the Effective Semantic Web

Myong-Hwan Yoo<sup>0</sup>, Hee-Joon Chung<sup>\*</sup>, Kang-Chan Lee<sup>\*\*</sup>,  
Sung-Han Kim<sup>\*\*</sup>, Jae-Hong Min<sup>\*\*</sup>, In-Jeong Chung<sup>\*</sup>

<sup>\*</sup> Dept. of Computer Science, Korea University

<sup>\*\*</sup> Protocol Engineering Center, Electronics and Telecommunications Research Institute

### 요 약

컴퓨터 통신기술이 급격히 발달하고 인터넷이 광범위하게 사용됨에 따라서 웹은 다양하며 무한한 용량의 데이터 원천으로 부각되었다. 그러나 웹의 사용이 지속적으로 증가함에 따라 웹 상에서 원하는 유용한 데이터를 찾고 구성하며 통합하는 일들이 매우 어렵게 되었다. 이러한 문제점을 극복하기 위해 1990년 말에 시맨틱 웹 기술이 소개되었다. 시맨틱 웹은 정보를 온톨로지로 구성함으로써 정보의 재사용성을 높여 주고 컴퓨터가 정보를 이해할 수 있도록 하여 이기종 간의 상호운용성을 보장한다. 더 나아가서는 에이전트에서 온톨로지를 판단, 재조합 함으로써 사용자가 원하는 서비스의 자동적인 실행과 추론을 할 수 있는 환경을 제공한다. 이와 같은 기능을 수행하는 시맨틱 웹을 위해 RDF(S), OIL, DAML, SHOE와 같은 마크업 언어가 제안되었다. 그러나 이 언어들은 지식표현을 위한 프레임 시스템과 기술 로직 등에 기반을 두고 있기 때문에 몇 가지 문제점을 지니고 있다. 본 논문에서는 기존의 시맨틱 웹을 위한 마크업 언어의 문제점을 알아보고 효과적인 시맨틱 웹의 구현을 위해 필요한 마크업 언어의 요건을 연구·제안한다.

## 1. 서 론

1990년대 중반에 이르러 인터넷이 광범위하게 사용됨에 따라 웹은 다양한 정보원으로써 무한한 저장 능력을 갖춘 정보의 바다로 부각되었다. 그러나 정보가 급속하게 증가함에 따라 유용한 정보를 효율적으로 찾는 것이 점점 어렵게 되었다. 이와 같은 문제점은 현재의 웹 마크업 언어들이 표현을 위한 정보만을 표시하여 이기종 간의 정보시스템과의 통합이 어렵고, 단일 기종 내에서도 디자인의 변화에 따라 정보도 함께 수정해야하며 정보의 의미 전달이 취약하기 때문이다. 이러한 문제점을 해결하기 위해 시맨틱 웹이 1990년대 말에 제안되었다.

시맨틱 웹은 현재의 웹 환경의 문제점을 해결하고 자동화된 웹 서비스를 제공하며 컴퓨터의 지능적인 정보 처리가 가능토록 웹 문서 내에 지식표현을 위한 온톨로지를 삽입하고 지식간의 관계를 설정하며 추론규칙을 포함시킨다. 이를 통해서 이기종 간의 상호운용성을 보장하고 사용자가 원하는 웹서비스의 발견, 자동적인 웹 서비스의 실행과 동시에 웹 서비스들의 통합과 상호작용을 하여 원하는 정보를 찾고 추론이 가능토록 한다[1].

온톨로지는 공유되는 데이터들의 개념화한 형식적이고 명백한 규정[1,2]으로써, 특정 도메인에서 사용되는 표준 어휘들의 모음이다. 온톨로지는 도메인 내의 지식을 표현하고 의사소통을 위한 어휘를 제공함으로써 이기종 간의 상호운용성 및 지식의 통일화 등의 장점을 가졌다. 그리고 어휘 사전의 역할이외에, 지식을 효과적으로 표현하기 위해 정보의 의미를 부여하고 또 정보들간의 관계를 설정해 준다. 따라서 컴퓨터는 온톨로지를 이용

하여 자동적인 실행과 추론을 하기 때문에 온톨로지의 중요성이 부각되었다.

RDF(S)[3], OIL[4], DAML[5], SHOE[6]와 같은 시맨틱 웹을 위한 마크업 언어는 웹 문서에서 온톨로지를 표현함으로써 웹 문서에 나타난 정보의 의미와 정보들 간의 관계를 설정해 주어 시맨틱 웹의 기능을 수행할 수 있도록 한다.

본 논문에서는 차세대 웹 환경인 시맨틱 웹을 위한 마크업 언어에 대한 연구로써 현재까지 개발되어 있는 RDF(S), OIL, DAML, SHOE의 언어에 대한 분석을 통하여 위 언어들의 문제점을 살펴보고 문제점을 해결할 수 있는 마크업 언어의 환경을 제시한다.

본 논문의 구성은 다음과 같다. 2장에서는 현재의 시맨틱 웹을 위한 마크업 언어를 분석하고 문제점을 알아보고 3장에서는 객체지향의 마크업 언어를 사용하여 문제점을 해결하고 그 이점을 살펴본다. 마지막으로 결론과 향후 과제를 언급한다.

## 2. 현재의 마크업 언어와 문제점

### 2.1. 현재의 마크업 언어

현재 시맨틱 웹을 위한 마크업 언어로써는 RDF(S), OIL, DAML, SHOE 등이 있다. RDF(S)는 시맨틱 웹의 가장 기본이 되는 언어로써 XML의 임의적 정보 표현과 정보의 의미표현 부족 단점을 개선하기 위해 제안된 언어이다. RDF(S)는 시맨틱 네트워크에 기반으로 하여 정보를 객체, 특성, 값의 쌍으로 표현하고 웹 문서에 포함된 자원들을 URI로 나타냄으로써 자원들 간의 충돌을

피하고 각 자원들 간의 구분 역할을 한다[3].

OIL과 DAML은 RDF를 사용하여 좀더 효과적인 온톨로지 구성을 위해 개발된 언어이다. OIL과 DAML은 인공지능 기반의 마크업 언어로써 온톨로지에 추론을 위한 규칙과 복잡한 정보의 의미를 효과적으로 표현해준다. RDF가 정보의 의미를 표현하는 반면, OIL과 DAML은 지식표현 분야인 프레임 시스템과 기술 로직을 기반으로 하였으며 이의 장점을 활용하여 정보의 복잡한 분류나 논리적인 관계를 설정하여 온톨로지를 표현해 줄 수 있다[4,5].

RDF(S), OIL, DAML이 서로 밀접한 연관성이 있으나 SHOE는 이와 다르게 독자적으로 개발된 언어로써 HTML의 확장된 형태로 HTML 태그와 함께 사용할 수 있다. SHOE는 ISA 계층구조와 FOL(First-Order Logic)과 혼 클로즈 로직을 기반으로 개발되었다. 이는 계층구조를 통한 분류체계를 제공하고 로직에 기반한 IF-THEN형태의 추론 규칙을 정의해주고 줌으로써 온톨로지를 표현한다[6].

## 2.2. 문제점

RDF는 시맨틱 네트워크를 기반으로 정보의 의미를 정확하게 설정하고 URI를 사용하여 자원의 정확한 기술을 통하여 컴퓨터가 정보를 이해할 수 있도록 한다. 그러나 정보의 의미만을 표현해줄 뿐 정보간의 관계나 동치성 등과 같은 연산·표현능력이 미약하며 추론을 위한 규칙 표현을 해줄 수가 없다. 이와 같은 문제점을 해결하고자 인공지능 기반의 마크업 언어인 OIL, DAML, SHOE 등과 같은 언어가 제안되었으나 이는 프레임 시스템이나 기술 로직에서 발생하는 문제점을 지니고 있다.

첫째, 직관적이지 못하다. 온톨로지 설계자에게 있어서 이는 중요한 문제이다. 프레임 시스템과 기술 로직 등과 같은 지식 표현 방식이 정형화되고 일관성 있는 환경을 제공하지만 클래스와 슬롯 등의 나열로 인하여 복잡한 구조를 지니게 된다. 이는 뒤에서 언급하는 나오는 문제점들과 연관성이 있다. 둘째, 프레임 시스템과 기술 로직에서 사용되는 클래스와 슬롯 등과 같은 구성요소들은 온톨로지 전반에 걸쳐 영향(global scope)[7,8]을 미친다. 이로 인해 정보의 확장, 유지 보수 등을 위해서는 하위 온톨로지 역시 상위 온톨로지의 모든 구성요소와 그 구조를 명확히 알고 있어야 하고 하위 온톨로지에서도 사용되는 클래스와 슬롯들과 상위 온톨로지와 충돌을 일으키는 문제점이 있다. 또한 시스템간의 온톨로지 교환 시에도 사용중인 다른 온톨로지와의 충돌하게 된다. 온톨로지 설계자 입장에서는 병렬적인 온톨로지 설계가 불가능하여 설계의 효율성을 감소시킨다[7]. 셋째, 온톨로지에서도 다른 온톨로지를 참조하는 효과적인 메커니즘을 제공하지 못하기 때문에 온톨로지의 재사용성이 떨어진다. 다시 말해, 온톨로지가 다른 온톨로지를 참조할 경우 필요에 따라 집합, 리스트 등과 같은 여러 형태의 자료구조를 제공하지 못한다. 넷째, 온톨로지의 분류체계에 있어서 기술 로직에 근간하기 때문에 상속에 의한 일반화(generalization) 및 상세화(specialization) 등과 같은 논리적인 분류체계는 우수하나[4,5,6] 의미적 연관성에 의

한 의미적 분류체계는 미흡하다. 이는 정보의 통합적인 관리와 연관성이 깊은 온톨로지간의 이용을 어렵게 한다. 마지막으로 온톨로지의 수정 및 확장에 의한 문제이다. 현재의 마크업 언어들은 온톨로지의 수정 및 확장 내용이 있을 경우 수처에 의한 버전 정보를 사용하여 온톨로지의 수정 및 확장을 나타낸다[4,5,6]. 이 버전 정보는 온톨로지를 판별하는 중요한 정보이지만 온톨로지의 변경사항에 버전 정보 충돌로 인하여 온톨로지의 수정을 어렵게 한다.

## 3. 시맨틱 웹을 위한 객체지향의 마크업 언어

### 3.1. 기존 마크업 언어의 문제점 해결방안

본 장에서는 위에서 언급한 문제점을 해결하기 위해 객체지향 패러다임을 사용하여 시맨틱 웹을 위한 마크업 언어의 요건을 제안한다.

객체 지향 패러다임은 추상 클래스, 상속, 정보은닉에 의해서 도메인의 개념을 단순화시키고 유연성과 온톨로지의 재사용성을 제고해 줄 수 있다.

첫째 객체지향 패러다임을 사용한 마크업 언어는 UML(Unified Modeling Language)[9]을 사용할 수 있다. UML은 소프트웨어공학 분야에서 소프트웨어의 설계·디자인 부분에서 사용되는 모델링 언어로써 그래픽 표기방법을 사용하여 클래스 다이어그램과 객체 다이어그램을 제공함으로써 온톨로지의 정적·동적 구조를 효과적으로 표현할 수 있기 때문에 직관적이다. 둘째, 프레임 시스템과 기술 로직과 달리 온톨로지를 각각의 객체로 나타내고 클래스 내의 슬롯들을 객체 내부의 범위로 한정시킴(local scope)[7,8]으로 온톨로지 구조를 단순화시키고 온톨로지 구조의 수정 및 확장을 용이하게 한다. 이와 같이 클래스와 슬롯을 객체 내부의 범위로 한정시킴으로써 다른 시스템간의 온톨로지 교환을 효과적으로 해줄 수 있을 뿐만 아니라 온톨로지 설계자들은 설계자들 간의 간략한 규칙에 의해 온톨로지의 병렬적 설계가 가능하게 한다. 셋째, 온톨로지 참조를 위한 효과적인 메커니즘을 제공한다. 마크업 언어를 프로그램언어와 같이 생각하여 배열 등과 같은 자료구조를 사용하도록 한다. 이는 도메인 영역의 개념을 다양한 형태의 온톨로지 구성이 가능하여 현실세계에 좀더 적합한 온톨로지를 만들어 줄 수 있다. 또한 URI를 통한 객체 참조로써 기존에 제작된 온톨로지의 재사용성을 높일 수 있다. 넷째 현실세계에 근접한 개념의 표현이 가능하다. 프레임 시스템이나 기술 로직 기반의 마크업 언어는 일반화 및 상세화를 통하여 계층적 분류를 한다. 이는 논리적 연관성에만 초점을 두어 분류한 방법으로써 다양한 기준으로 의미적 분류가 필요한 현실세계에는 적합하지 않다. 이에 자바에서 사용되는 패키지[9]와 유사한 개념을 사용하여 의미적 연관성이 높은 온톨로지를 분류한다. 이는 패키지 내의 온톨로지간이 접근을 용이하게 하고 패키지 외의 온톨로지와의 구분을 통하여 온톨로지의 통합적 관리와 온톨로지간의 이용을 쉽게 할 수 있다. 마지막으로 버전 정보를 URI와 함께 명시함으로써 같은 버전 넘버의 차이가 있는 온톨로지의 배포를 막을 수 있다.

객체지향 패러다임을 마크업 언어에 이용함으로써 그

외에도 다음과 같은 문제점을 해결해 줄 수 있다. 기존의 마크업 언어에서 상속을 받은 하위 클래스에서는 상속받음을 나타냄과 동시에 대표성을 띠는 상위 클래스를 나타내기 위해서 하위 클래스들을 나열해 주어야 한다. 그러나 객체지향 패러다임에서의 추상 상위 클래스를 사용함으로써 하위 클래스에서 상속받음을 표기해주고 별도로 상위 클래스를 위한 리스트를 만들어 줄 필요가 없다. 상속받는 하위 클래스는 상위 클래스로부터 필요에 따라 상속받아 중복되는 값의 불필요한 입력을 피할 수 있다.

### 3.2. 객체지향의 마크업 언어의 이점

시맨틱 웹의 효과적인 구현을 위해 UML 등과 같은 정형도구를 제공하는 객체지향 개념을 마크업 언어에서 사용함으로써 얻을 수 있는 이점은 다음과 같다.

첫째, 산업계 전반적인 파급효과가 크다. 현재의 마크업 언어들은 인공지능의 지식표현분야가 주로 연구되고 있다. 이는 산업계 전반에 걸친 이해가 없기에 때문에 이해의 병목현상이 일어날 수 있다. 그러나 객체지향의 마크업 언어는 현재 많이 배포되어 있는 C++, 자바와 같은 프로그램 언어에 친숙한 사용자에게 사용이 쉬울 뿐만 아니라 산업계에 널리 사용되는 UML을 사용하여 직관적이고 효율적인 온톨로지의 생성관리가 가능하다.

둘째, 온톨로지 설계 측면에서의 효율성이다. 프레임 시스템이나 기술 로직과는 달리 객체지향 방식은 정보는 닉이나 추상화를 사용하여 병렬적 설계 작업을 용이하게 하며 프레임 시스템이나 기술 로직과는 달리 하위 클래스들의 생성이 자유롭다. 또한 객체지향의 정보는닉이나 추상화는 온톨로지의 확장성, 유연성을 증대시켜 정보의 재사용성을 높여줄 뿐만 아니라 온톨로지 전반의 계산 복잡도를 줄일 수 있다. 또한 프레임 시스템과 기술 로직의 설계 방식이 하향(top-down)방식인데 비해 객체지향 방식은 이용된 마크업 언어는 상향(bottom-up)-하향(top-down)의 설계가 모두 가능하다.

셋째, 정보의 재사용성의 향상이다. 추상 상위 클래스의 사용은 설계측면에서의 효율성 증대뿐만 아니라 여러 하위 온톨로지의 대표성을 띠기 때문에 다른 온톨로지를 대신할 수 있을 뿐만 아니라 온톨로지의 교체가 용이하다. 또한 하위 클래스들의 특수성을 부각시킴으로써 좀 더 현실세계에 부합하는 온톨로지의 구성이 가능하다.

마지막으로 다양한 에이전트 및 컴퓨팅 환경에 적합하다. 시맨틱 웹 환경은 정보의 추출·제조합·추론·자동실행 등과 같은 기능을 수행하기 위해 에이전트와 함께 연계되어야 한다. 특히 멀티 에이전트의 경우 문제 해결을 위해 적합한 에이전트들에게 질의를 하기 때문에 다양하게 표현이 가능한 온톨로지가 필요하다. 프레임 시스템이나 기술 로직의 지식표현은 견고하고 일관성 있는 환경을 제공하지만 다양한 기법의 에이전트에게는 적합하지 않다. 이에 객체 지향의 마크업 언어는 각 에이전트에게 적합한 온톨로지의 표현이 가능하고 컴퓨터, PDA 등과 같은 계산능력이 다양한 컴퓨팅 환경에 적합한 온톨로지를 통한 효과적인 정보 표현이 가능하다.

### 4. 결론 및 향후 과제

시맨틱 웹은 컴퓨터가 이해할 수 있는 정보를 웹 문서에 삽입하여 자동적인 정보의 추출·제조합·자동실행이 가능하도록 하는 차세대 웹 환경이다. 시맨틱 웹의 이상적 기능을 수행하기 위해서는 가장 먼저 웹 문서에 온톨로지를 삽입할 수 있는 마크업 언어에 대한 연구가 필요하다. 본 논문에서 현재 개발 중인 혹은 개발된 시맨틱 웹을 위한 RDF(S), OIL, DAML, SHOE와 같은 마크업 언어를 소개하고 이에 대한 문제점을 분석하였다. 그리고 효과적인 시맨틱 웹을 구현하기 위하여 현존 마크업 언어의 문제점에 대한 방안으로 객체지향의 마크업 언어의 요건을 제시하였다.

시맨틱 웹은 독립된 새로운 웹 환경이 아닌 기존의 웹 환경을 수용하고 자동화되고 지능적인 웹 환경을 제공하는데 그 목적이 있다. 이를 위해서는 기존의 마크업 언어들을 수용할 수 있어야 한다. 따라서 독자적인 마크업 언어가 아닌 기존의 마크업 언어를 기반으로 하여 발전해 나가야 할 것이다. 그 외에 마크업 언어로 표현된 지식을 사용하여 효과적인 웹 서비스를 제공하도록 에이전트의 연구와 함께 병행되어야 하며 객체의 의미 표현, 재사용성, 확장성 등의 연구가 오랫동안 진행해 온 소프트웨어 공학측면에서의 연구도 필요하다.

### 참고문헌

- [1] McIlraith, S.A., Son, T.C., Honglei Zeng. Semantic Web services. IEEE Intelligent Systems. Vol. 16 Issue 2, p.46-53 March-April, 2001.
- [2] J. Hendler. Agents and the Semantic Web. IEEE Intelligent Systems. Vol.16 Issue 2, p.30-37 March-April 2001.
- [3] S. Decker, P. Mitra, S. Melnik. Framework for the semantic Web: an RDF tutorial. IEEE Internet Computing, Vol. 4 Issue 6 p.68-73, Nov.-Dec. 2000
- [4] D. Fensel, I. Horrocks, F. Van Harmelen, S. Decker, M. Erdmann, M. Klein. OIL in a nutshell. Proceedings of the 12th European Workshop on Knowledge Acquisition, Modeling, and Management. p.1-16, 2000.
- [5] About DAML. <http://www.daml.org/about.html>
- [6] The SHOE Specification <http://www.cs.umd.edu/projects/plus/SHOE/spec.html>
- [7] S. Cranefield, S. Haustein, M. Purvis UML-Based Ontology Modelling for Software Agents Proceedings of the Workshop on Ontologies in Agent Systems, 5th International Conference on Autonomous Agents. 2001
- [8] Alexander Felfernig, Gerhard Friedrich, Dietmar Jannach, Markus Stumptner, Markus Zanker. Transforming UML domain descriptions into Configuration Knowledge Bases for the Semantic Web. Proceedings of the Workshop on Knowledge Transformation for the Semantic Web p. 11-18. 2002
- [9] Jesse Liberty. Beginning Object-Oriented Analysis and Design with C++. WROX 1998