

시멘틱 웹 기반 컴포넌트 저장소 관리 시스템 설계

김양훈[†], 장준식^{**}, 김국보^{***}

요 약

정보기술의 발달과 웹 기술의 발전으로 인하여 정보의 양이 증가함에 따라 기존의 웹은 많은 문제점을 노출하고 있다. 소프트웨어 에이전트를 이용한 웹 서비스의 이용으로 그 한계점을 극복하고자 하였으나, 기존의 소프트웨어 에이전트는 사용자 요구사항에 의한 지식 제공이라는 패러다임에 있어서 만족스러운 결과를 제공하지 못하였다. 또한 최근의 소프트웨어 개발 형태는 CBD(Component Based Development)를 기조로 한다. 그러나 CBD를 이용하여 새로운 컴포넌트를 구축하기 위해서는 많은 비용으로 인해, 저비용으로 빠르고 정확하게 컴포넌트의 정보를 웹상에서 획득할 수 있는 모델이 요구되고 있다. 본 논문에서는 시멘틱 웹상에서 컴포넌트를 획득하고 관리하기 위한 저장소의 관리 시스템을 설계하고 기존 컴포넌트 저장소 관리 시스템들과의 비교 분석한 결과를 제시하였다. 그리고 기존 컴포넌트 저장소 관리 시스템의 한계점인 낮은 적중률과 검색어의 제한, 단어에 의한 높은 의존도, 제한된 화면상의 정보를 극복하고 컴포넌트를 위한 지식검색을 할 수 있는 방안을 제안하였다.

The Design of Component Repository Management System for Semantic Web

Yang-Hoon Kim[†], Joon-Sik Jang^{**}, Guk-Boh Kim^{***}

ABSTRACT

According to the development of information & web technology and the amount of information increases, there have been numerous problems exposed. Although, software engineers try to overcome the limitations by using software agent and web service, there hasn't been satisfactory result in the paradigm of current software agent being a provider to the user's demand. Moreover, the latest configuration of software development is based on CBD (Component Based Development). However, to construct new component using CBD costs great deal of expenses and therefore, a new model which can acquire component information on the web promptly and accurately with low expenses is required. In this paper, the repository management system which acquires and manages on the Semantic web is designed and compare them to the existing component repository management and present its analysis result. In addition, to overcome the limitations of existing component repository system; low accuracy of search result, restrictive search vocabulary and faulty information, the specific plan is presented to perform a knowledge search for the component.

Key words: Semantic Web(시멘틱 웹), Component(컴포넌트), Component Repository(컴포넌트 저장소)

※ 교신저자(Corresponding Author) : 김국보, 주소 : 경기도 의정부시 신곡동 684 동아아파트 104동401호, 전화 : (031) 539-1961, FAX : (031) 539-1960, E-mail : kgb@deafen.ac.kr
접수일 : 2007년 9월 18일, 완료일 : 2007년 12월 18일
[†] 준회원, 대전대학교 컴퓨터공학과 박사과정
(E-mail : kimyh7902@daejin.ac.kr)

^{**} 준회원, 대전대학교 컴퓨터공학과 박사과정
(E-mail : joonsikjang@nate.com)
^{***} 종신회원, 대전대학교 컴퓨터공학과 교수
※ 본 연구는 2007년도 대전대학교 교내연구비에 의해 연구되었음.

1. 서 론

1989년 Tim Berners-Lee에 의해서 제안된 웹(World Wide Web : WWW)은 사람들 간의 정보 공유에 매우 큰 영향을 미쳤다. 이에 따라 수많은 정보가 인터넷상에 발표되고 유통되어서 사람들은 정보의 바다에서 자신이 원하는 정보를 유용하게 찾을 수 있고 이를 활용할 수 있게 되었다[1,2].

웹이 발달될수록 웹상에 수많은 정보가 범람함에 따라 원하는 정보를 발견하는 작업이 점점 더 어려워지는 현상이 발생하였고 이에 따라 소프트웨어 에이전트라는 방안을 제안하였다. 하지만 소프트웨어 에이전트는 정보를 찾고자 하는 사람에게 주요 키워드에 따라 검색하는 기능을 제공할 뿐, 실제 검색 프로그램을 이용하면 자신이 원하는 정보만 나오는 것이 아니라, 수천 또는 수만의 결과가 나오고 다시 사람이 이 중에서 자신이 원하는 정보를 찾아야 하는 수고를 해야 했다.

정보의 양이 많아지고 사람들은 보다 빠르게 정보를 얻고자 하는 욕구가 증가함에 따라 기존의 웹은 많은 문제점을 노출하기 시작했다. 또한, 소프트웨어 에이전트를 이용한 웹 서비스의 이용으로 자동화된 서비스를 얻고자 하지만, 기존의 소프트웨어 에이전트는 단순 작업만이 가능하기 때문에 이와 같은 요구 사항을 만족시킬 수 없었다.

이와 같이 소프트웨어 에이전트가 제대로 동작하지 못하는 이유는 웹의 구조적인 문제에서 유래한다. 즉 웹에 있는 모든 정보는 사람을 위한 정보로 사람은 이를 이해할 수 있으나 소프트웨어 에이전트는 웹에 있는 정보를 이해할 수 없기 때문이다. 이와 같은 문제를 극복하기 위해서 웹을 주장하였던 Tim Berners-Lee는 1999년에 W3C를 중심으로 차세대 웹 기술인 시멘틱 웹(Semantic Web)을 제안했다[3].

소프트웨어 개발의 생산성과 소프트웨어 시스템의 안정성은 소프트웨어 공학계 뿐만 아니라, 업계에서도 중요한 연구 과제 중의 하나가 되었다. 작은 단위의 소프트웨어 부품을 조립하여 더 큰 소프트웨어 시스템을 구축하는 조립 생산 방식이 소프트웨어의 재사용성을 보장해 줄 것으로 생각되었던 객체지향 언어나 방법론들이 우수한 개념과 연구 성과에도 불구하고, 소프트웨어 재사용 측면에서는 주목할 만한 결과를 보이지 못 하면서, 학계와 업계는 컴포넌

트에 관심을 돌리기 시작했다[4-6].

기존의 개발자들은 컴포넌트 저장소에 컴포넌트를 저장하고 키워드로 설명해 놓은 후 검색하여 획득하였다. 개발자들이 저장소에 컴포넌트들을 많이 축적해 갈수록 기존의 웹상에서 보여주던 한계점이 나타나게 되었다. 게다가, 저장소에 축적된 컴포넌트들의 명세, 검색, 버전관리 등에 못지않게 협업에 의한 연구를 잘 하는 것도 매우 중요해졌고, 이에 원격 개발 팀들이 협의할 수 있는 설비와, 웹을 통해 원격 저장소에 질의할 수 있는 기능이 필요하게 되었다.

본 연구는 시멘틱 웹상에서 컴포넌트를 획득하기 위한 저장소 시스템에 관하여 연구하고 기존 컴포넌트 저장소와의 비교, 분석하였다. 기존 컴포넌트 저장소의 한계점인 낮은 적중률과 검색어의 제한, 단어에 의한 높은 의존도, 제한된 화면상의 정보를 극복하고 컴포넌트를 위한 지식검색을 할 수 있도록 하였다.

2. 관련연구

2.1 기존 컴포넌트 명세 기법[7-9]

컴포넌트의 기본 원리는 데이터와 업무 기능의 통합, 캡슐화, 식별성의 객체 기술을 근간으로 하는 기본 원칙을 따른다. 컴포넌트는 인터페이스라고 불리는 명시적인 명세상의 의존관계를 통해 오브젝트 명세를 정확히 기술함으로써 객체 기술 원칙을 확장하고 있다.

2.1.1 Catalysis에 의한 명세

Catalysis는 UML을 컴포넌트 기반 개발에 적용한 최초의 방법론 중 하나이며 아키텍처, 패턴, 프레임워크 등 오늘날 컴포넌트 기반 개발을 위해 필수요소라고 생각하고 있는 수많은 아이디어를 제시하였다.

Catalysis의 주목할 만한 특징은 다이어그램 같은 컴포넌트의 구문적인 측면과 더불어 OCL과 같은 제약어를 사용하여 사전/사후조건에 대한 컴포넌트의 의미적인 측면을 함께 기술할 수 있는 방안을 제시했다.

2.1.2 Select Perspective에 의한 명세

Select Perspective는 앞에서 설명한 Catalysis와 비슷한 시기에 만들어진 컴포넌트 기반 개발 방법론이다.

이 방법론은 개발 초기에 비즈니스 프로세스 모델링의 중요성을 강조하고 있으며 비즈니스 프로세스를 구현 모델로 하여 단계적으로 전환하기 위한 명확한 프로세스를 정의하고 있다.

Select Perspective는 앞서 언급한 RUP와 유사하기도 하지만 매우 독특한 구조를 가지고 있다. 이 방법론을 적용한 프로젝트는 시간의 흐름이 따라 정렬(Align), 설계(Architect), 조립(Assembly)의 세 가지 단계로 진행되며 이것은 다시 하나 이상의 증분(Increment)으로 나뉘어 작업을 수행한다.

Select Perspective의 가장 큰 약점은 비즈니스 오브젝트를 설명하고 있는 모델의 구분이 다소 모호하다는 것이다.

2.1.3 UML(Unified Modeling Language)에 의한 명세

UML은 객체지향 분석과 설계를 위한 모델링 언어로, 표기하려는 대상을 다이어그램을 사용하여 나타내고, 그 대상에 의미를 부여한다. 방법론이 아닌 방법론에 따른 설계에서 사용되는 언어이며, 공식적인 표기법으로 사용된다.

UML에 의한 컴포넌트 명세는 기존 모델링 기술에 스테레오 타입, 인터페이스 클래스, 실현관계를 추가하고 이를 이용하여 컴포넌트를 기술하게 된다.

그림 1은 사용자 주문에 따른 스테레오 타입의 컴포넌트 타입을 나타낸 것으로, 사용자와 주문은 컴포넌트 타입이고, 각각 하나의 인터페이스를 가지고 있으며 점선은 이들 간의 의존성을 나타내고 있다.

2.1.4 RUP에 의한 명세

RUP는 90년대 초반에 등장하였던 다양한 객체지향 방법론들을 통합하여 발전시킨 프로세스 프레임워크이다. 가장 큰 특징은 성공적인 경험을 바탕으로

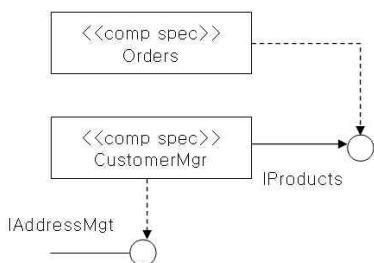


그림 1. UML 컴포넌트 명세

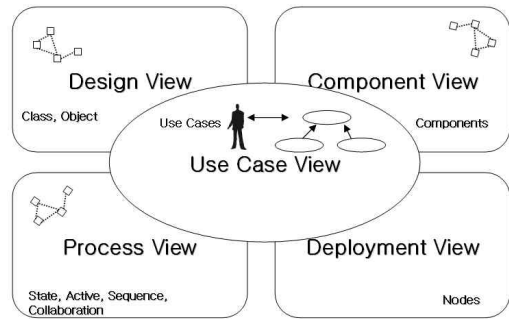


그림 2. RUP의 4+1 view

6가지의 지침을 강조하였으며, 개발 프로세스를 자동화 시켜주는 다양한 도구를 제공하였다.

RUP는 최근 소프트웨어 공학의 거의 모든 경험적 요소를 포함하고 있지만 이들 사이에서 필연적으로 발생하게 되는 충돌을 어떻게 해결할 지에 대한 구체적인 지침이 부족하다. 예를 들어 RUP는 ‘유스케이스 중심적(UseCase Driven:기능 지향적)’임과 동시에 ‘아키텍처 중심적 (Architecture Centric : 객체 지향적)’이지만 이들이 상충되었을 때 어느 것을 우선시 해야 할지에 대해서는 명확하게 설명하고 있지 않다. 그림 2는 RUP의 4+1 view에 대하여 보여주고 있다.

2.2 시멘틱웹

반 세기동안 수많은 학자들에 의해 하이퍼링크가 연구되었고, 이렇게 축적된 기반이 마침내 팀 버너스 리의 웹으로 1차 완성되었다.

지난 10년간 우리가 접한 웹은 단지 선형적 구조에서 탈피한 하이퍼링크가 네트워크로 구현된 정도에 불과하다.

초기 웹의 특징은 단순링크라는 점이다. 선형적 구조를 탈피한 초기의 웹은 다음과 같이 세 가지의 문제점을 지니고 있었다.

1. 쓰레기 정보의 증가와 처리 문제
2. 정보의 생산, 관리, 재활용 문제
3. 정보 검색, 배포, 수집의 문제

시멘틱웹은 초기 웹의 이런 문제를 해결하기 위해서 제안된 것이다. 시멘틱웹은 웹의 창시자인 팀 버너스 리에 의해 1998년에 제안된 차세대 웹의 이름으

로 각종 회의와 연구를 통해 지속적으로 관련 규격과 기술들이 개발되고 있다. 시멘틱웹은 웹 자체가 지식 기반으로 자동화될 수 있도록 거대한 정보 구조가 짜인 웹이다. 이를 위해 컴퓨터가 정보를 구성하는 자원의 뜻을 이해하고 추론할 수 있는 지능형 웹을 목표로 한다. 팀 버너스리는 ‘시멘틱웹은 현재 웹의 확장으로 잘 정의된 의미를 제공함으로써 사람과 컴퓨터가 좀 더 협력을 잘 할 수 있는 웹’이라고 설명하였다.

시멘틱웹은 컴퓨터와 컴퓨터 사이에서 정보를 주고받을 때 잘 정리된 좀 더 많은 정보를 추가로 제공해 정보해석력을 높이고, 이를 통해 자동화처리를 향상시킨 웹이다. 이때 컴퓨터들 사이에서 주고받는 추가적인 정보는 사용자에게는 가시적으로 제공되지 않는 숨은 자료 즉, 메타데이터(meta data)를 활용하게 된다. 그리하여, 사용자나 관리자가 관여하지 않고 컴퓨터와 컴퓨터 사이에 스스로 일을 처리하는 지능화된 웹을 지향한다. 컴퓨터 사이에 자동으로 일을 처리하기 위해서는 컴퓨터들이 대화를 나누고 대화 내용을 이해할 수 있는 장치가 필요하다. 이를 위해 컴퓨터가 이해할 수 있는 언어로 웹을 구성해야 하는데, RDF, 온톨로지(Ontology), OWL(Ontology Web Language) 등이 이를 위해 논의되고 있는 기술이다. 시멘틱 웹의 구조는 팀 버너스 리의 제안에 따라 다음 그림 3과 같다[10].

2.3 온톨로지

정보기술의 발달과 웹 기술의 발전으로 인하여 엄청난 양의 정보가 축적되고 있으며 이에 대한 손쉬운 접근이 가능하게 되었다. 지식사회에서는 자신이 필요로 하는 지식을 얻고 활용하는 것이 개인뿐만 아니

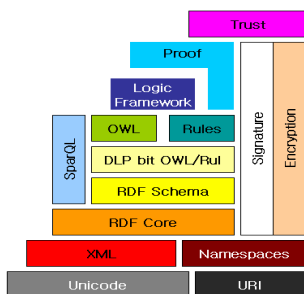


그림 3. 시멘틱 웹의 구조

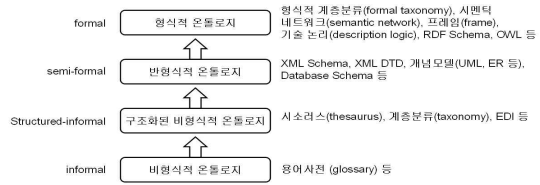


그림 4. 형식성과 추론 메커니즘에 따른 온톨로지

라 기업에게도 매우 중요한 일이다. 하지만 정보의 홍수 속에서 필요로 하는 지식을 찾기가 쉬운 일이 아니다. 이러한 문제를 해결하기 위해 인터넷 상의 검색엔진, 기업의 지식관리시스템(KMS : Knowledge Management Systems) 등이 구축되었다. 그러나 지금까지의 정보기술은 주로 단순히 정보를 저장하고 처리하는 데 중점을 두었기 때문에 필요한 정보의 지능적 검색이나 지적인 수행 능력은 부족하였다. 정보의 홍수 속에서 정보를 지식화하고 활용하는 일은 여전히 개인의 일로 남아있다.

질적으로 향상된 정보서비스를 위해서는 축적된 정보를 단순히 공유하는 것 외에도 지식을 기술하고 표현하며 공유할 수 있도록 하는 체계적이며 통제된 메커니즘이 필요하다. 온톨로지는 정보기술 분야에서 지식을 표현하고 활용하기 위해 사용되는 핵심 기술이다.

온톨로지는 인간뿐만 아니라 소프트웨어 에이전트 간의 커뮤니케이션 시 특정한 단어가 나타내는 개념의 의미를 이해하는 데 사용된다. 또 온톨로지는 추론규칙을 포함할 수 있기 때문에 새로운 사실을 자동으로 추출하거나 제약 조건에 맞지 않는 오류를 찾아낼 수 있다. 따라서 온톨로지는 지식을 수집하고 표현하는 데 사용될 뿐만 아니라 추론 기능을 지원함으로써 지식 모델링 및 지식 검색 시 유용한 수단으로 이용될 수 있다. 온톨로지는 지식 명세로서 다양한 분야에서 응용될 수 있기 때문에 각각의 적용 분야에 따라 여러 온톨로지가 존재할 수 있다. 그림 4는 형식성과 추론 메커니즘에 따른 온톨로지 분류를 보여주고 있다.

온톨로지의 사용 목적은 크게 시멘틱 상호운용성, 표준화, 커뮤니케이션, 지식관리 및 검색의 네 가지로 요약할 수 있다. 이러한 목적들이 지식의 공유와 재사용의 관점에서 보면 서로 밀접한 관련성을 가지고 있다[11].

3. 시멘틱 웹기반 컴포넌트 저장소 관리 시스템(CMSS) 설계

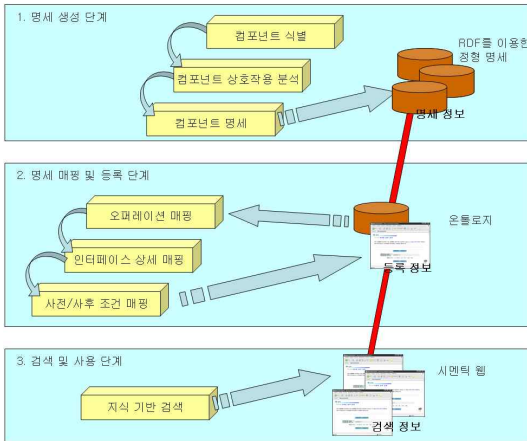


그림 5. 시멘틱 웹 기반 컴포넌트 명세 생성을 위한 구조

시멘틱 웹 환경에서 사용자의 요구사항 분석에 따른 컴포넌트의 개발과 온톨로지 저장, 그리고 검색 및 획득을 위한 그 구조는 다음 그림 5와 같다.

3.1 시멘틱 웹기반 컴포넌트 명세 생성을 위한 개발 단계

컴포넌트 아키텍처는 컴포넌트 명세, 컴포넌트 구현 그리고 컴포넌트 오브젝트 중 초점을 맞추어 정의될 수 있다. 컴포넌트 명세 아키텍처 다이어그램은 컴포넌트 명세와 인터페이스만을 포함하고 있다. 다이어그램 상에서 모든 의존성은 인터페이스의 연태나 인터페이스에 의존하고 있는 컴포넌트 명세의 형태로 나타난다.

의존성은 컴포넌트 명세를 구현하는데 제약조건

으로 작용되고 어떤 식으로 구현하든 반드시 이 의존성이 구현에 반영되어야 한다. 기존의 컴포넌트 명세 아키텍처 상의 의존성은 상호작용 다이어그램 같은 명세의 상세한 부분으로부터 도출된다.

이에 따라 컴포넌트 명세 생성을 위해 크게 3단계로 나누어 질 수 있다.

3.1.1 컴포넌트 식별 단계

컴포넌트 식별 단계는 사용자의 형태를 파악하여 요구사항을 정의하고 초기 컴포넌트 아키텍처를 생성한다.

3.1.2 컴포넌트 상호작용 분석 단계

컴포넌트 식별 단계를 거쳐 초기 컴포넌트 아키텍처에 따라 컴포넌트 상호작용 분석 단계는 필요한 오퍼레이션을 발견하고 인터페이스의 책임 영역을 할당한다.

컴포넌트들의 재사용을 위하여 공용 오퍼레이션을 도출하고 사용 패턴을 추출한다.

인터페이스의 영역 할당을 위해 컴포넌트들 간의 의존관계를 최소화하고 참조 무결성을 고려한다.

3.1.3 컴포넌트 명세 단계

컴포넌트 명세 단계는 오퍼레이션, 인터페이스, 컴포넌트 명세를 정제하고 명확히 기술한다. 인터페이스 정보 모델을 바탕으로 컴포넌트 상태를 정의하고, 인터페이스 내부의 오퍼레이션에 따른 사전/사후 조건을 기술하며 제약조건을 사용하여 표현한다.

3.2 온톨로지를 위한 컴포넌트 명세 매핑 단계

컴포넌트를 온톨로지 상에 저장하기 위해서는 각

표 1. 컴포넌트 명세의 의미론적 매핑

| 구성자 (constructor) | 신택스 (syntax) | 시멘틱스 (semantics) | 컴포넌트 |
|-----------------------------------|---------------|---|-------------------------------|
| 개념 이름 (concept name) | C | $\{x C(x)\}$ | 사용자 인터페이스에 대하여 컴포넌트 타입을 정의한다. |
| 결합 (conjunction) | $C \cap D$ | $\{x C(x) \wedge D(x)\}$ | 컴포넌트의 의존관계에 대해 정의 한다. |
| 값 제한 (value restriction) | $\forall R.C$ | $\{x \forall y,R(x,y) \rightarrow C(y)\}$ | 컴포넌트의 수치적인 값 제한에 대해 정의한다. |
| 존재정량 (existential quantification) | $\exists R.C$ | $\{x \exists y,R(x,y) \wedge C(y)\}$ | 컴포넌트 존재정량 대해 정의한다. |
| 수 제한 (number restriction) | $(\geq nR)$ | $\{x \{y R(x,y)\} \geq n\}$ | 컴포넌트 상호작용의 수 제한에 대해 정의한다. |

명세에 따라서 의미론적인 수식으로 매핑을 해야 한다. 표 1은 컴포넌트의 정의를 정형적인 의미론적 표현으로 나타낸 것이다.

3.2.1 인터페이스 명세

3.2.1.1 오퍼레이션 명세

구조적 타입, 인터페이스 타입, 불변식의 각 속성, 연관관계에 따른 명세를 해준다. 구조적 타입은 컴포넌트 내부의 구조를 명세하고 인터페이스는 컴포넌트 외부를 명세하며 불변식은 공리로써 추론의 기본이 되는 명제로 표 2와 같이 분류 한다.

3.2.1.2 인터페이스 상세 명세

인터페이스 정보 모델은 해당 인터페이스를 제공하는 컴포넌트가 수행해야 할 기능과 인터페이스 사용자가 기대하는 기능을 정확하게 정의하기 위해서 필요한 모든 명세물과 인터페이스를 한 곳에 모아놓은 산출물로 표 3과 같다.

3.2.1.3 사전/사후 조건 명세

인터페이스의 명세를 통하여 표 4와 같이 스테레오 타입의 사전/사후 조건으로 산출된 제약사항을 명세한 것이다.

표 2. 인터페이스 명세

| 오퍼레이션 명세 | 명세 부분 | 표현 | 컴포넌트의 내용 |
|----------|-------|----------------------------------|--------------------|
| 구조적 타입 | 속성 | rdfs:Resource rdf:value | 식별 개념 실제 구조화된 값 |
| | 연관관계 | rdfs:isDefinedBy rdfs:comment | 자원 정의 자원 상세 서술 |
| 인터페이스 타입 | 속성 | rdfs:label | 인터페이스의 태그 |
| | 연관관계 | rdf:type | 인스턴스를 클래스에 연결 |

표 3. 상세 명세

| 내용 | | 표현 | 내용 |
|-------------|-------|-----------------|------------|
| 인터페이스 상세 명세 | 패키지 | rdf:Description | 각 컴포넌트를 표현 |
| | 정보 모델 | rdf:resource | 목적부의 자원 참조 |
| | 오퍼레이션 | rdf:ID | 컴포넌트의 ID |

표 4. 사전/사후 조건 명세

| 명세 | 표현 | 컴포넌트의 내용 |
|---------|-----------|---------------------|
| 사전/사후조건 | rdf:about | coms:ID에 정의된 자원을 참조 |

3.3 시멘틱 웹 기반 컴포넌트 저장소의 등록 단계

앞에서 제시한 컴포넌트 명세 중 인터페이스 상세 명세에 대한 부분을 표 5에서 나타내었다.

3.4 시멘틱 웹 기반 컴포넌트 저장소의 컴포넌트 식별 및 획득 단계

온톨로지에 적합한 내용으로 매핑되어 표현된 컴포넌트는 식별 및 획득하기 위하여 다음 그림 6과 같은 과정을 거친다.

4. 제안시스템 성능 평가

4.1 시스템 구현

표 5. 인터페이스 상세 명세의 RDF 표현

```
<?xml version="1.0" encoding="euc-kr">
<!DOCTYPE rdf:RDF[
  <!ENTITY rdfs "http://www.w3.org/2000/01/rdf-schema#">
  <!ENTITY rdf "http://www.w3.org/1999/02/22-rdf-syntax-ns#">
]>
<rdfs:RDF xmlns:rdfs="&rdfs;"
  xmlns:rdf="&rdf;"
  xml:base="http://se.ce.daejin.ac.kr/ont-book">
  <rdfs:Class rdf:ID="Student"
    rdfs:label="학생">
    rdfs:subClassOf rdf:resource="&rdfs:Resource"/>
  </coms:Class>
  <rdfs:Class rdf:ID="Professor"
    rdfs:label="교수">
    rdfs:subClassOf rdf:resource="&#Univ"/>
  </rdfs:Class>
  <rdfs:Class rdf:ID="Univ"
    rdfs:label="대학교">
    rdfs:subClassOf rdf:resource="&rdfs:Resource"/>
  </rdfs:Class>
  <rdfs:Property rdf:ID="have"
    rdfs:label="소속되다">
    <rdfs:domain rdf:resource="&#Student">
    <rdfs:range rdf:resource="&#Univ">
  </rdfs:Property>
</rdfs:RDF>
```

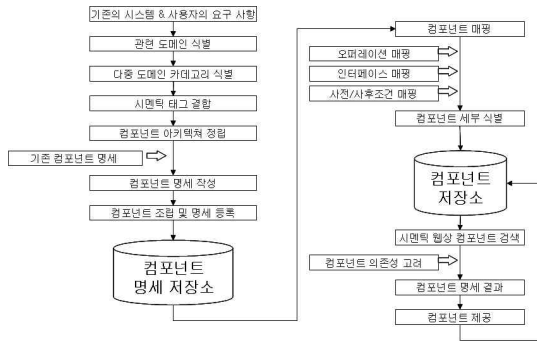


그림 6. 시멘틱 웹 기반 컴포넌트 저장소의 컴포넌트 식별 및 획득 절차

본 논문에서는 타 시스템과의 비교 및 평가를 위하여 시멘틱 웹기반 컴포넌트 저장소 관리 시스템의 프로토타입을 구현하였다.

Basic Step 에서는 컴포넌트의 기본 식별을 위한 컴포넌트의 이름, 가치 문서 검색과 다중연결 및 낱말 설명에 적용되는 시멘틱 태그(키워드), 컴포넌트의 범위 한정을 위한 컴포넌트 도메인, 컴포넌트 제작자, 컴포넌트 제작일, 컴포넌트 제작에 따른 버전, 간단한 기본 설명에 대해서 그림 7과 같이 입력한다.

Step 2는 인터페이스 명세 단계로 오퍼레이팅 명세에서 컴포넌트 오브젝트에게 제공되거나 전달되어지는 정보를 명시하기 위한 입력 매개변수, 컴포넌트 오브젝트에 의해서 수정 혹은 반환되는 정보를 명시하기 위한 출력 매개변수, 컴포넌트 오브젝트의 상태 변화 결과, 적용해야 할 예외상황을 입력한다. 그리고 인터페이스 정보 모델에서 인터페이스의 모든 오퍼레이션 명세를 작성할 수 있을 만큼의 정보를 담고 있는 타입, 속성, 관계, 불변식, 스냅샷을 입력한

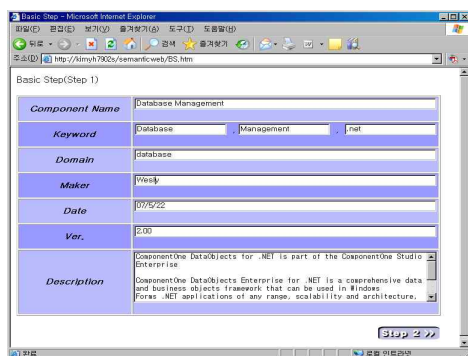


그림 7. 컴포넌트 식별을 위한 명세 기본 단계

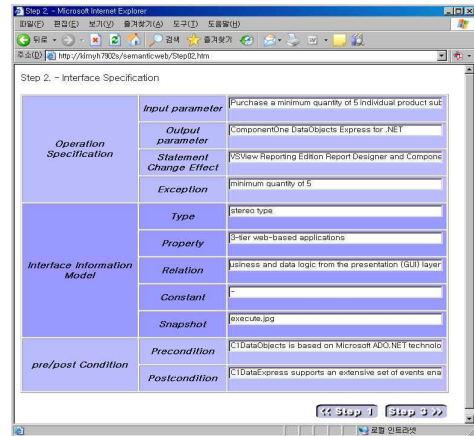


그림 8. 컴포넌트 식별을 위한 명세 2단계

다. 마지막으로 선/후 조건에서 선 조건과 후 조건을 입력한다. 선/후 조건은 알고리즘이나 구현에 대한 설명이 아니라, 오퍼레이션의 파급 효과만을 명세화 하며 오퍼레이션 사용자와 간단한 계약서 구실을 한다. 또한 오퍼레이션이 해야 할 일을 상세하게 명세화하며 항상 쌍으로 입력한다. 사후조건은 사전조건이 참일 경우에 오퍼레이션의 결과가 무엇이 될 것인지 명세하며 이 단계의 모습은 그림 8과 같다.

Step 3은 컴포넌트 명세 단계로써, 제공되는 인터페이스와 사용되는 인터페이스에 대한 명세를 하고, 컴포넌트 상호작용의 제약조건, 인터페이스간의 제약조건을 명세하고 그림 9처럼 표현된다.

이러한 세 단계를 거쳐 명세된 컴포넌트는 저장소에 저장되고 이를 검색하기 위한 모듈로 추천 키워드 기능에 대한 프로토타입을 구현하였다.

명세 단계에서 명세된 컴포넌트의 다양한 정보들을 온톨로지 상에 저장하고 이를 키워드 추천 방식에 적용하여 그 내용은 그림 10, 11처럼 검색된다.

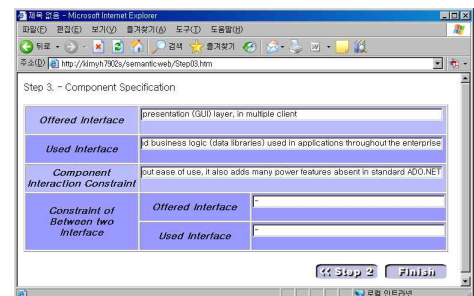


그림 9. 컴포넌트 식별을 위한 명세 3단계

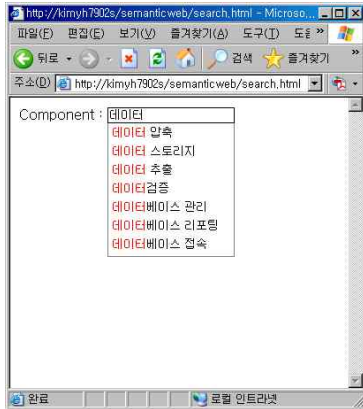


그림 10. 컴포넌트 획득을 위한 검색

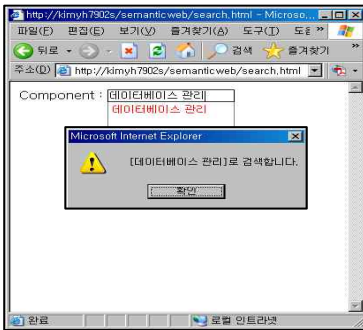


그림 11. 컴포넌트 검색 결과

적용되는 시멘틱 웹의 특성상 온톨로지의 저장소 추론의 기능을 탑재하지만, 다양한 추론 결과를 도출해 내기 위해서는 장기간의 많은 데이터의 축적이 필요하다.

4.2 타 시스템과 비교

기준에 컴포넌트 저장소 관리 시스템들은 그림 12

표 6. 제안 시스템의 비교 분석

| 시스템 | 항목 | 주요기능 | 개발 프로세스 제공 | 메타 데이터 수집 | 명세 제공 | 에이전트 | 검색방법 | | | 추론 |
|------------------|----|--------------------------|------------|-----------|-------|------|------|------|--------|-----|
| | | | | | | | 키워드 | 카테고리 | 키워드 추천 | |
| Yes Data | | 컴포넌트 판매 | No | No | Yes | No | Yes | Yes | No | No |
| Component Source | | 컴포넌트 판매 | No | No | Yes | No | Yes | Yes | No | No |
| e-BCOS | | 에이전트 지원의 컴포넌트 정보 등록 및 검색 | Yes | No | Yes | Yes | Yes | Yes | No | No |
| CMSS | | 컴포넌트 온톨로지 저장 및 추론, 검색 | Yes | Yes | Yes | No | Yes | Yes | Yes | Yes |

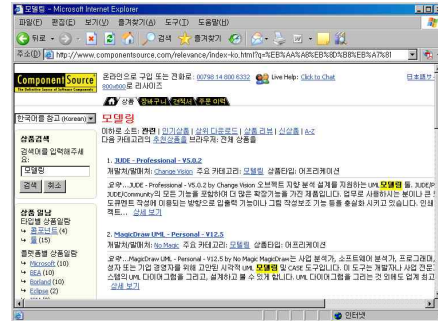


그림 12. 컴포넌트를 판매하는 타 시스템

와 같은 획득 과정에서 에이전트를 사용하고 키워드 및 카테고리 검색을 지원하는 등 다양한 시도가 있었다.

하지만, 시멘틱 웹으로 발전되어가고 있는 지금 좀 더 유용성, 확장성, 신뢰성, 상호운영성이 있는 저장소 관리 시스템의 개발이 필요하다.

본 연구에서는 근래의 연구에 발맞추어 시멘틱 웹 기반 컴포넌트 저장소 관리 시스템을 설계하고 프로토타입을 구현하였다.

표 6은 현재 웹상에서 운영되고 있는 컴포넌트 유통 사이트들과 본시스템의 특징을 비교하였다. CMSS는 시멘틱 웹 기반 컴포넌트 저장소 관리 시스템으로, 명세 생성 단계, 명세 매핑 및 등록단계, 검색 및 사용의 3단계로 나뉘었다. 이러한 과정들은 요구 사항 분석에 따른 분석, 설계, 구현의 3단계를 충실히 이행할 수 있게 하며, 일반적인 컴포넌트 명세의 등록과 해당 명세의 온톨로지 매핑으로 시멘틱 웹에서 지원하는 추론기능이 포함된다. 카테고리 및 키워드 검색뿐만 아니라 단어의 단계에 따른 추론 기능을 내포한 키워드 추천기능을 사용하여 확장성 있는 시스템을 보여 주었다.

5. 결론 및 향후 연구

정보의 양이 증가함에 따라 소프트웨어 에이전트를 이용한 기존의 웹 서비스의 활용은 그 한계점을 극복하고자 하였으나, 단순 작업만이 가능한 한계점으로 인하여 사용자 요구사항에 대해 능동적인 서비스를 제공하지 못하였다. 이에 따라, 시맨틱 웹 기반의 컴포넌트 저장소 관리 시스템에서는 사용자 서비스 제공에 대한 능동적인 대처를 위하여 컴포넌트에 대한 지식 기반 형태의 서비스 제공이 필요하다.

본 논문에서는 시맨틱 웹상에서 사용자에 대한 능동적 서비스를 위하여 컴포넌트를 획득하기 위한 저장소관리 시스템에 대해 제시하였다. 이는 다음과 같이 세 가지 측면으로 정리할 수 있다.

첫째, 개발자에 의해서 제공되는 컴포넌트에 대한 명세를 온톨로지에 적합한 명세로 매핑 한다.

둘째, 시맨틱 웹상에서 개발자가 컴포넌트를 온톨로지에 등록하여 저장소에 저장한다.

셋째, 지식 기반 검색 기법으로 기존의 키워드 또는 카테고리 기반의 검색을 지원하는 소프트웨어 에이전트에 비해 사용자에게 능동적인 정보를 제공한다.

향후연구로는 자연어 적인 처리 요소에 대한 시맨틱적 내용을 좀 더 자동화 하여 지식 검색에 대한 최적화를 통하여 사용자 측면에서 좀 더 편리한 시스템을 구성해야 할 것이다.

참 고 문 헌

[1] Dieter Fensel, James Hendler, Henry Lieberman, and Wolfgang Wahlster, *Spinning the Semantic Web*, MIT Press, 2003.
 [2] Geigoris Antoniou and Frank van Harmelen, *A Semantic Web Primer*, MIT Press, 2004.
 [3] Tim Berners-Lee, James Hendler, and Ora Lassila, *The Semantic Web*, Scientific American, May 2001.
 [4] R. Kazman, M.Klein, M. Barbacci, T. Longstaff, H. Lipson, and J. Carriere, "The architecture tradeoff analysis method," *The*

4th IEEE International Conference on Engineering of Complex Computer Systems, pp. 68-78, Aug. 1998.
 [5] Dobrica, L. and Niemela, E. "A survey on software architecture analysis engineering," *IEEE Transactions on Software Engineering*. Vol. 28, No. 7, pp. 638-653, July 2002.
 [6] Klaus Schmid. "The economic impact of product line adoption and evolution," *IEEE software*. Vol. 19, No. 4. pp. 50-57, 2002.
 [7] Grundy. J, "Storage and retrieval of software components using aspects," *Computer Science Conference. ACSC 2000. 23rd Australasian*, pp. 95-103, 2000.
 [8] 김국보. "웹 기반 컴포넌트 저장소 e-BCOS 설계 및 구현," 퍼지 및 지능 시스템학회 논문지, 제15권, 제2호, pp. 211-223, 2005.
 [9] 문미경, 염근혁. "컴포넌트 기반 개발(CBD)을 위한 도메인 공학," 정보처리학회지, 제10권, 제3호, pp. 40-47, 2003.
 [10] John Davies, *Toward The Semantic Web-Ontology-driven Knowledge Management*. John Wiley & Sons. 2003.
 [11] Quan. Liu, Xinjuan. Jin, and Yihong. Long. "Research on Ontology-based Representation and Retrieval of Components," *Eighth ACIS International Conference on*. pp. 494-499, 2007.



김 양 훈

2005년 대진대학교 컴퓨터공학과 졸업(학사)
 2007년 대진대학교 대학원 컴퓨터공학과 졸업(석사)
 2007년~현재 대진대학교 컴퓨터공학과 박사과정
 관심분야 : 소프트웨어 공학, 컴포넌트, 시맨틱 웹



장 준 식

2003년 대전대학교 컴퓨터공학과(학사)
2005년 대전대학교 대학원 컴퓨터공학과(석사)
2006년~현재 대전대학교 컴퓨터공학과 박사과정
관심분야 : 멀티미디어 HCI, 멀티미디어 ontology, web2.0



김 국 보

1984년 서울산업대학교 전자계산학과 졸업(학사)
1986년 연세대학교 공학대학원 전자계산학과 졸업(석사)
1997년 대구카톨릭대학교 전산통계학과 졸업(박사)
1988년~1990년 : 해군 중앙전산소장
1990년~1993년 : 부경대학교 교수
1993년~현재 : 대전대학교 컴퓨터공학과 교수
관심분야 : 소프트웨어공학, 시스템 분석 및 설계, e-Biz 시스템