

요구사항 온톨로지 기반의 시맨틱 태깅을 활용한 산출물의 재사용성 지원을 위한 요구사항추적 방법

(A Method for Requirements Traceability for Reuse of Artifacts using Requirements-Ontology-based Semantic Tagging)

이 준 기 [†] 조 혜 경 ^{**} 고 인 영 ^{***}
(Junki Lee) (Hae-Kyung Cho) (In-Young Ko)

요약 산출물들의 추적 관계 정의를 이용한 요구사항 추적을 통해, 기존의 컴포넌트 자체의 재사용뿐만 아니라 컴포넌트 개발과정에서 나오는 다양한 산출물들을 요구사항 기반으로 재사용할 수 있다. 이러한 재사용성 증가를 목적으로 하는 요구사항추적을 지원하기 위해서는 산출물들이 요구사항을 기반으로 표현될 수 있어야 하고, 표현된 요구사항을 기반으로 하여 추적관계를 추론하는 메커니즘이 제공되어야 한다. 이를 위해, 본 논문에서는 시맨틱스 기반의 요구사항추적을 지원하기 위해서 요구사항 온톨로지를 하여 기술한다. 그 다음에 산출물들이 요구사항 온톨로지를 통해서 시맨틱 태깅되는 기술을 기술한다. 본 논문은 이와 같이 요구사항 추적을 위한 메커니즘을 제안하고, 요구사항 온톨로지의 구조를 정의하며 프로토타입을 제시한다.

키워드 : 요구사항 추적, 산출물 재사용, 시맨틱 태깅, 요구사항 온톨로지

Abstract Requirements traceability enables to reuse various kinds of software artifacts, which are the results from software development life cycle, rather than reuse source code only. To support requirements traceability for reuse of software artifacts, 1) artifacts should be described based on requirements and 2) a requirements tracing method should be supported. In this paper, we provide a description model for annotating requirements information to software artifacts by using requirements ontology. We also provide semantic tagging method users to efficiently annotate artifacts with the requirements ontology. And we finally present how requirements traceability is supported based on requirements ontology and also suggest the system architecture for requirements traceability support.

Key words : Requirements traceability, Artifact reuse, Semantic tagging, Requirements ontology

1. 서론

현재 컴포넌트 기반 개발 방법론에서의 재사용은 기존의 소프트웨어 컴포넌트 자체의 재사용뿐만 아니라, 컴포넌트 개발과정에서 나오는 여러 가지 산출물들(요구사항 명세서, 디자인 산출물, 테스트 케이스 명세서 등)의 재사용까지 확대되고 있다.

요구사항추적(Requirements traceability)이란, 고객의 요구사항으로부터 소프트웨어 개발과정에서 생산되는 산출물들 간의 관계를 정의한 것을 말한다[1]. 이러한 요구사항추적을 통하여 실제로 개발된 산출물들이 요구사항에 잘 부합하는지 확인할 수 있고, 개발 과정에서 요구사항이 변경되었을 때, 그에 따라서 변경되어야 하는 산출물들이 어떠한 것이 있는지 파악할 수 있게 된다.

요구사항추적은 위에서 예로 제시되었던 요구사항관리 분야에 이득만을 가져다 줄 뿐 아니라, 소프트웨어

· 본 연구는 방위사업청과 국방과학연구소의 지원으로 수행되었습니다.

(2008-SW-12-DM-03)

· 이 논문은 제34회 추계학술대회에서 '요구사항 온톨로지 기반의 시맨틱 태깅을 활용한 산출물의 재사용성 지원을 위한 요구사항추적 방법'의 제목으로 발표된 논문을 확장한 것임

[†] 학생회원 : 한국정보통신대학교 공학부
haedal2@icu.ac.kr

^{**} 정 회 원 : 한국정보통신대학교 공학부 연구교수
hgcho@icu.ac.kr

^{***} 정 회 원 : 한국정보통신대학교 공학부 교수
iko@icu.ac.kr

논문접수 : 2008년 1월 11일

심사완료 : 2008년 4월 26일

Copyright©2008 한국정보과학회 : 개인 목적이거나 교육 목적인 경우, 이 저작물의 전체 또는 일부에 대한 복사본 혹은 디지털 사본의 제작을 허가합니다. 이 때, 사본은 상업적 수단으로 사용할 수 없으며 첫 페이지에 본 문구와 출처를 반드시 명시해야 합니다. 이 외의 목적으로 복제, 배포, 출판, 전송 등 모든 유형의 사용행위를 하는 경우에 대하여는 사전에 허가를 얻고 비용을 지불해야 합니다.

정보과학회논문지: 소프트웨어 및 응용 제35권 제6호(2008.6)

개발 산출물들의 재사용 증가에도 도움을 줄 수 있다. 다시 말해서, 요구사항추적 정보를 활용하여, 여러 단계의 산출물들을 요구사항 기반의 패키지(Package) 형태로 묶어서 재사용하여, 기존의 컴포넌트 자체의 재사용보다 효과적인 재사용의 지원을 가능하게 할 수 있다 [2]. 또한 기존의 산출물들의 검색이 각 산출물들의 명세방법에 따라 다른 종류의 질의어를 통해서 가능했던 반면에, 요구사항추적을 통하여 요구사항이라는 한 종류의 질의어를 통한 검색을 가능하게 한다.

본 연구는 현재 국방 환경 내에 분산되어 있는 다양한 종류의 소프트웨어 개발 산출물들의 재사용을 위해, 요구사항을 기반으로 컴포넌트 및 산출물들을 검색 및 추출할 수 있게 해주는 개발환경에 대해서 연구를 진행 중에 있다. 또한 그 개발환경을 사용자가 찾고자 하는 산출물 외에 같은 요구사항을 가지고 있는 다른 종류의 산출물들까지 검색 추출함으로써 소프트웨어 개발 프로세스 전반에 걸친 재사용을 지원할 수 있게 해준다. 이를 위해 온톨로지 기반의 요구사항 표현 모델 (Ontology-based requirements description model)과 요구사항추적을 통한 산출물 추출 등의 연구가 수행되어야 한다.

본 논문에서는 컴포넌트 기반 소프트웨어 개발 방법론이 적용된 소프트웨어 산출물들의 재사용 증대를 위한 시맨틱스(Semantics) 기반의 요구사항 추적 지원의 방법과 요구사항 추적기에 대한 구조를 서술한다. 이를 위해서 각 산출물들의 내용을 요구사항 온톨로지를 기반으로 표현하는데, 시맨틱 태깅(Semantic tagging) 기술을 사용하여 이를 실현한다.

본 논문의 구성은 다음과 같다. 2절의 관련연구에서는 본 연구와 기존 연구들과의 차이점을 기술한다. 3절에서는 요구사항추적을 위해서 어떠한 요구조건들이 필요한지 설명한다. 이를 토대로 4절에서는 온톨로지와 시맨틱 태깅을 활용한 의미 기반의 요구사항 추적기의 구조를 제시하고 5절에서는 우리가 사용하는 요구사항 온톨로지에 대해서 설명한다. 6절에서는 구현된 프로토타입(Prototype)을 제시하고 3절에서 제시한 요구사항 추적의 라이프 사이클에서 요구조건들이 어떤 방법으로 만족되었는지 설명한다. 그리고 마지막으로 결론과 향후 연구 방향에 대해서 설명한다.

2. 관련연구

2.1 프로젝트 개발 라이프 사이클에서의 요구사항 추적

소프트웨어 개발과정에서의 관리, 유지 보수의 중요성이 점차 증가함에 따라서 요구사항추적에 대한 중요성도 증가하고 있다. 여기서 요구사항추적이란, 고객의 요구사항으로부터 소프트웨어 개발과정에서 나오는 산출

물들의 관계를 추적 관계를 통해 기술하는 것을 말한다. 이러한 요구사항 추적을 통하여 개발된 소프트웨어가 요구사항들을 만족하는지 확인할 수 있고, 요구사항이 변경되었을 때 그에 따라서 변경되어야 하는 여러 종류의 산출물들(디자인 산출물, 소프트웨어 코드, 테스트 산출물 등)이 무엇인지 확인하고 변경하는 등의 이점을 가질 수 있다.

요구사항 추적을 지원하기 위해서는 서로 다른 종류의 산출물들 간의 요구사항추적 관계가 정의되어 있어야 하는데 이러한 작업들은 주로 사람에 의해서 정의되고 관리되기 때문에 많은 시간과 노력이 요구되고 실수가 생기기 쉬운 단점들을 가지고 있다. 이러한 문제들을 해결하기 위해서 추적 관계 복잡도를 낮추는 방법[3,4], 요구사항들 간의 논리적 모순을 추론할 수 있도록 해주는 요구사항 분석 및 추출 방법의 제안[5,6] 등의 연구가 존재한다. 이러한 연구들을 통해서 요구사항추적 관계 정의와 관리에 필요한 노력과 시간을 줄여줄 수 있으나, 여전히 초기 요구사항추적 관계는 사람이 정의해 줘야 하는 문제점을 가지고 있다.

한편, 역공학 측면으로, 요구사항추적 관계 작업을 자동화 하려는 연구들이 진행 되고 있는데 이러한 연구들은 주로 정보추출(Information retrieval) 기술들을 이용하여 자동화를 추구하고 있다. 참고문헌 [7,8]에서, 산출물들간의 관계를 추출하기 위해서 산출물들에서 키워드를 추출하고, 서로 비슷한 키워드(Keyword)를 포함하고 있는 산출물들끼리 관계를 정의함으로써 요구사항추적 관계를 지원하고 있다. 이러한 방법들은 사람의 노력을 최소화 시킨다는 장점을 가지고 있다. 하지만 서로 다른 종류의 산출물들은 서로 다른 내용들을 포함하고 있기 때문에, 그 안에서 공통적인 키워드를 추출하기 쉽지 않고 또한 낮은 정확성을 나타낸다는 문제점을 가지고 있다.

2.2 재사용성 증가를 위한 요구사항 추적

요구사항추적은 위에서 언급한 소프트웨어 관리적인 측면에서의 이득뿐만 아니라, 컴포넌트를 포함한 다양한 산출물들의 재사용성 증가의 이득 또한 가져다 줄 수 있다. 예를 들어 컴포넌트 재사용에 있어서, 단순한 소스코드의 재사용을 넘어 소스코드가 만들어 지기까지의 다양한 산출물들(요구사항 명세서, 유스케이스 명세서 등)까지 재사용할 수 있다면, 이는 재사용성이 증가되었다고 말할 수 있다.

그리하여, 본 논문에서는 기존에 개발되어 저장된 컴포넌트 및 관련 산출물을 요구사항 기반으로 찾아서 관련된 산출물들의 패키지로 묶어서 재사용할 수 있도록 하는 것을 재사용성 증가를 위한 요구사항 추적이라 정의한다.

재사용 증가를 위한 요구사항추적은 프로젝트 개발

라이프 사이클에서 요구사항추적과 차이점들이 있다. 소프트웨어 관리에서의 요구사항추적은 초기에 정의된 요구사항 추적 표(Requirements traceability matrix)를 이용하기 때문에 요구사항 명세나 다른 산출물들의 명세가 자연어로 기술되어도 추적을 지원하는데 문제가 없다. 하지만 재사용을 위한 요구사항추적은 이미 개발되어 저장되어 있는 산출물들에서 추적 관계를 추출해야 하기 때문에 산출물들의 내용을 이해하고 유사한 산출물들을 추출할 수 있는 방법이 지원되어야 한다.

이를 지원하기 위한 연구들은 크게 정보추출 기술들을 사용한 방법과 정형화된 요구사항 명세와 산출물들의 메타모델(Metamodel)을 이용한 방법[9,10]들이 있다. 하지만 정보추출 기술을 사용한 방법의 경우 앞 절에서 언급했던 것처럼 정확성에 문제점이 있고, 정형화된 요구사항 명세방법의 경우, 새로운 방법을 요구함으로써 자연어를 기반으로 개발되는 산출물들의 재사용이 불가능하고 또한 사용자에게 부담을 주는, 따라서 실효성이 떨어진다는 단점이 있다.

2.3 온톨로지 기반의 요구사항 추적

2.3.1 온톨로지의 정의

온톨로지란, 어떠한 객체의 개념(Concept)을 표현하고, 객체들간의 관계(Relation)를 정의함으로써, 객체를 시맨틱스(Semantics) 기반으로 표현할 수 있도록 지원하는 표현 모델이다[11]. 온톨로지는 크게 세 가지 요소로 표현되는데, 객체(Class), 객체의 속성을 표현하는(Property) 그리고, 객체들간의 관계(Relation)가 그것이다. 온톨로지는 객체들을 표현하는데 있어서 공통 어휘(Common vocabulary)를 제공함으로써, 서로 다른 도메인 지식이나, 어휘로 표현된 객체들도 검색을 가능하게 하고, 따라서 기존 키워드 검색 방법과 비교하여 의미 기반의 검색을 지원하며, 이는 검색의 정확성을 높이는데 기여한다.

2.3.2 온톨로지 기반의 요구사항 추적

[12,13]은 요구사항 추적 관계를 온톨로지 기반으로 찾고 관리하는 방법을 제안하였다. 이들 연구에서는 소스코드와 산출물들의 구조를 온톨로지화 하고, 소스코드 온톨로지(Source code ontology)와 문서 온톨로지(Document ontology)를 기반으로 요구사항 추적 관계를 찾는데 사용하고 있다. 비록 이들 연구에서는 시맨틱스 기반의 요구사항 추적 방법을 제안하고 있지만, 현재에는 문서 및 소스코드의 구조에 대한 온톨로지만을 제공함으로써, 도메인 지식을 기초로 하는 시맨틱스 기반 검색은 지원하지 않고 있다.

3. 재사용성 지원을 위한 요구사항 추적기의 요구조건

재사용성을 위한 요구사항 추적에는 프로젝트 개발 라이프 사이클에서의 요구사항추적과 다른 요구 조건들을 찾아 볼 수 있다. 따라서 이 절에서는 특히 재사용성 지원을 위한 요구사항 추적기의 요구 조건들을 기술한다.

- 개발 프로세스와의 독립성: 본 연구의 요구사항추적의 목적은 다양한 프로세스에서 개발된 여러 종류의 산출물들의 재사용을 지원하는 것이다. 따라서 특정한 소프트웨어 개발 프로세스에 국한되지 않는 일반적인 방법을 통하여 요구사항추적을 지원할 수 있어야 한다.
- 산출물 검색 결과에 대한 정확성: 산출물들의 재사용성 증가를 위하여 각 산출물들은 요구사항에 대한 정확한 정보를 담고 있어야 하고, 사용자가 요구사항을 기반으로 검색하였을 때, 정확한 추적정보를 얻을 수 있게 해주는 메커니즘이 제공되어야 한다.
- 요구사항 추적 관계 관리의 편의성: 요구사항 추적표의 작성처럼, 요구사항 추적 관계의 관리에 많은 시간과 노력을 필요로 한다. 따라서, 여러 가지 요구사항 추적 관계의 관리 방법이 있음에도 불구하고, 실제로는 제대로 관리되지 않거나, 개발이 완료된 후에 추가적인 작업을 통해서 요구사항 추적 관계를 재정의 하는 문제점들이 있다. 따라서 이러한 관리에 드는 부담을 줄여줄 수 있는 방법을 제공하여, 요구사항 추적 관계가 효과적이고 효율적으로 관리될 수 있어야 한다.

4. 온톨로지기반 시맨틱 태깅을 이용한 요구사항 추적기의 구조

이 절에서는 위의 요구조건을 만족시키기 위한 우리의 접근방법과 저장소 구조를 설명한다. 기존의 키워드 매칭 방법에서 벗어나 요구사항 추적과 검색의 정확성을 높이기 위해서, 본 연구에서는 온톨로지를 이용한 시맨틱스 기반의 요구사항 추적을 지원한다. 그리고 요구사항 추적기의 편의성과 독립성 지원을 위해 시맨틱 태깅을 이용하여, 기존의 요구사항 명세 방법이나 산출물 기술 방법의 변화 없이 요구사항추적을 지원할 수 있도록 한다.

그림 1은 요구사항 추적기의 구조를 보여준다. 요구사항 추적기는 크게 두가지의 기능으로 나뉘어진다. 하나는 요구사항 온톨로지를 기반으로 한 시맨틱 태깅을 수행하여, 각 산출물들에 시맨틱스를 부여하는 기능과, 다른 하나는 산출물들에 부여된 시맨틱스를 기반으로 요구사항을 추적하여, 관련 있는 재사용 대상인 산출물 패키지를 추출하는 기능이다.

먼저 요구사항 온톨로지를 기반으로 산출물들의 추적 관계를 추출하기 위해서는 산출물들이 요구사항 온톨로지를 기반으로 표현되어 있어야 한다. 하지만 주로 요구사항과 그 외 다른 산출물들은 자연어로 명세 되어 있

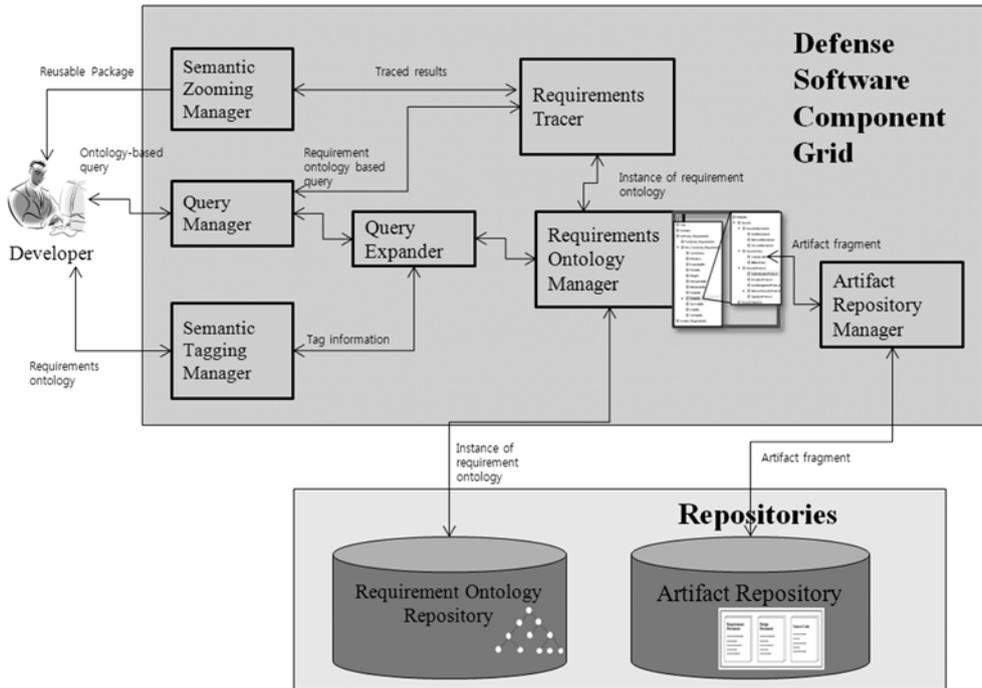


그림 1 시맨틱 태깅 기반 산출물 재사용을 위한 요구사항 추적기의 구조

기 때문에 온톨로지 기반의 표현이 어렵다. 기존의 다른 연구에서는 이러한 문제점을 해결하기 위해서 새로운 명세와 메타모델을 정의하였는데[9,10], 사용자에게 정형화된 명세 방법을 요구함으로써 실효성의 문제점을 가진다. 또한 산출물들은 그림 및 UML 다이어그램과 같은, 다양한 표현방법으로 기술되는데, 기존의 방법으로는 다양한 기술 방법을 가진 산출물들의 추적의 지원은 불가능하다.

따라서 본 연구에서는 사용자들이 쉽게 사용할 수 있고, 다양한 명세모델로 표현된 산출물들뿐 아니라, 기존의 자연어 기반의 산출물 명세 방법을 변화시키지 않으면서 산출물들에 시맨틱스를 부여하는 방법으로 시맨틱 태깅이라는 기술을 사용한다. 태그(Tag)는 다양한 의미를 가질 수 있는 메타데이터를 활용하여 정보의 분류와 검색을 용이하게 하기 위한 방법이다[14]. 시맨틱 태깅은 이러한 태그에 의미 정보를 부여함으로써 컴퓨터가 문장의 의미를 이해할 수 있도록 하는데 도움을 준다.

그리하여, 본 논문에서는 산출물의 재사용 향상을 위해 시맨틱 태깅에 사용되는 요구사항 온톨로지를 제시한다. 레거시 산출물들의 대한 의미 정보는 그 요구사항 온톨로지를 이용하여 부여되게 된다.

그림 1의 산출물 재사용을 위한 요구사항 추적기의 주요 구성 요소의 기능은 다음과 같다.

- 질의 관리자(Query Manager): 사용자로부터 온톨로지 기반의 요구사항을 입력 받는다. 입력 받은 온톨로지는 Requirements Tracer로 전송하여, 관련 산출물과 요구사항 추적을 할 수 있도록 지원한다.
- 시맨틱 태깅 관리자(Semantic Tagging Manager): 산출물들에 요구사항기반의 시맨틱스를 넣기 위하여, 사용자로부터 관련 있는 요구사항 온톨로지 개념을 입력 받는다.
- 질의 확장기(Query Expander): 사용자가 쉽게 관련 요구사항 개념을 검색하기 위해서, 자동 완성 기능을 제공한다. 따라서 사용자는 온톨로지 개념과 유사한 단어를 입력함으로써, 요구사항 온톨로지 내에 관련 있는 개념이 무엇이 있는지 찾아볼 수 있다.
- 요구사항 추적기(Requirement Tracer): 사용자로부터 입력 받은 온톨로지를 기반으로, 유사한 산출물들을 검색할 수 있도록 지원한다. 또한 같은 시맨틱스를 가지고 있는 다른 종류의 산출물들 또한 검색하여, 사용자가 재사용 가능성이 있는 대상들을 패키지 단위로 묶어서 재사용할 수 있도록 지원한다.

5. 시맨틱 태깅을 위한 요구사항 온톨로지 명세 모델

요구사항 온톨로지 기반 시맨틱 태깅을 위해서는, 먼

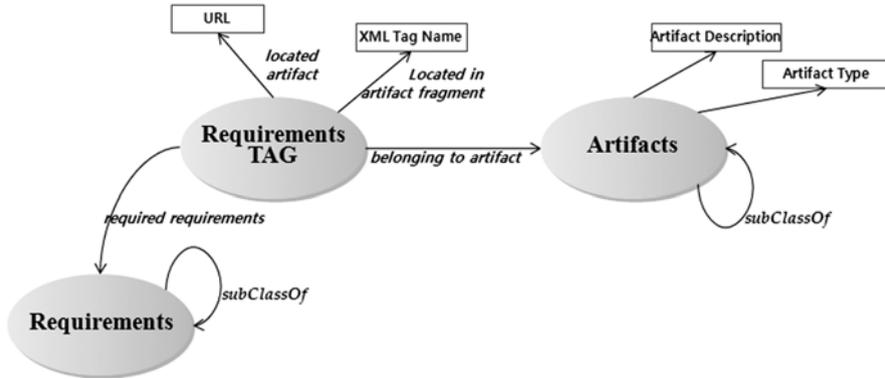


그림 2 요구사항 온톨로지를 이용한 시맨틱 태그의 표현 모델

저 요구사항 온톨로지가 정의되고 그 요구사항 온톨로지를 이용하여 시맨틱 태깅이 어떻게 수정되는지가 언급되어야 한다. 그리하여 본 절에서는 먼저 국방 도메인에 특화된 요구사항 온톨로지 구축 내용을 기술하고, 그 다음 시맨틱 태깅시 각 태깅에 대한 정보를 저장하는 시맨틱 태깅 명세 모델에 대해 설명한다.

먼저, 그림 2는 요구사항 온톨로지를 이용한 시맨틱 태그의 표현 모델 (Representation model) 을 보여준다. 이 표현 모델은 시맨틱 태깅 내용 및 태깅 대상에 대한 정보를 얻는데 이용된다. 즉 그림에서, “Requirements” 온톨로지는 태그가 의미하는 요구사항을 요구사항 온톨로지로서 표현한 것을 의미하고, “Artifacts” 온톨로지는 태깅되는 산출물의 종류 및 다른 산출물들과의 관계 등을 표현하여, 요구사항 추적 시 해당 산출물의 개발 단계 및 산출물의 메타 정보 등을 파악할 수 있도록 도와준다. 또한 “URI”와, “URL”은 태깅된 산출물의 위치 및 고유한 ID를 부여하기 위해 사용되며, “XML Tag Name”은 산출물 전체 중에서 태깅된 부분을 XML 문서의 태그를 기반으로 알 수 있도록 도와준다.

그림 3과 그림 4는 시맨틱 태그의 표현 모델 중, 요구사항 온톨로지의 구축 모습을 보여준다. 먼저 그림 3은 기능적 요구사항 온톨로지를 보여주는데, 기능적 요구사항은 재사용이 이루어지는 도메인에 특정한(Domain specific) 기능 분류(Classification) 제공한다. 본 논문은 국방 소프트웨어분야의 재사용을 연구가 중요 관심이므로, 현재 국방 도메인에서 사용되는 컴포넌트에 대한 기능적인 분류인 공통 운용환경의 세그먼트 분류 방법을 기반으로 기능적 요구사항을 구체화 하였다. 참고로, 세그먼트란 국방 도메인에서 만들어 지는 소프트웨어를 기능적 단위로 나눈 것으로써 소프트웨어 컴포넌트 개념을 포함하고, 시행 시, 자동적인 업그레이드가 가능한 모듈을 의미한다.



그림 3 기능적 요구사항 온톨로지

그림 4는 요구사항 온톨로지 명세 모델에 따라서 현재 구축중인 비기능적 요구사항 온톨로지의 모습을 나타낸다. 비기능적 요구사항에 대한 일반적인 개념을 추

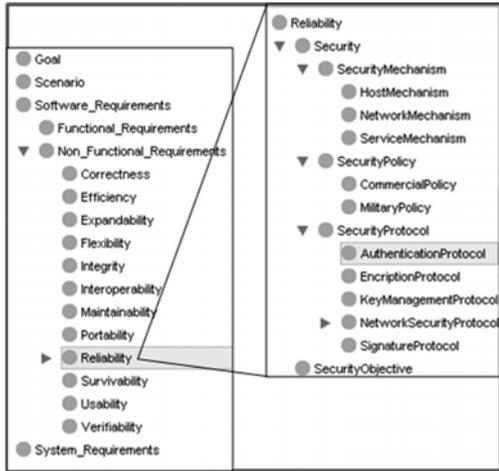


그림 4 비기능적 요구사항 온톨로지

출하기 위해서 IEEE 표준 문서[15,16]를 활용하고, 효율성(Efficiency), 상호 운용성(Interoperability), 신뢰성(Reliability) 등 비기능적 요구사항을 분류하였으며, 특히 미국 해군에서 구축한 온톨로지[17]을 이용하여, “Security”부분의 온톨로지를 구체화 하였다.

6. 요구사항 추적기의 프로토타입 구현 및 검증

본 논문에서 제시된 연구를 기반으로, 웹 기반의 프로토타입을 구현하였다. 먼저 그림 5는 요구사항 온톨로지

를 이용한 시맨틱 태깅의 방법을 보여준다.

먼저 사용자는 산출물 내에서 자신이 태깅하고자 하는 부분을 선택하여, 요구사항 온톨로지를 기반으로 해당 요구사항 개념을 입력하여, 시맨틱 태깅을 하게 된다. 이 과정에서 사용자는 기능적 혹은 비기능적인 요구사항 온톨로지에서 적절한 개념을 선택하게 되며, 이렇게 태깅된 산출물에 대한 정보는 “Requirement Ontology Repository”에 인스턴스를 생성하여 저장하게 된다. 본 연구의 프로토타입에서는 산출물들을 효율적으로 관리하기 위하여, 산출물들을 XML로 기술하였다.

시맨틱 태깅 수행시에 저장되는 정보는 먼저, URI를 통하여 산출물의 고유한 ID를 부여 받게 되고, URL에는 산출물이 있는 실제 위치를 저장하여 후에 산출물들을 찾을 때 URL에 있는 위치정보를 이용하여 추출할 수 있도록 지원한다. 또한 산출물 타입정보와 산출물의 XML Tag 정보를 활용하여 산출물 내에 정확히 어떠한 곳에 온톨로지 정보가 매핑(Mapping)되었는지 알 수 있다.

그림 6은 태깅된 정보를 기반으로 산출물을 검색한 결과의 화면이다. 사용자가 재사용하고자 하는 산출물이, 경보서비스 기능을 제공하는 기능적 요구사항을 가지고 있고, 그 중에서, 경보 서비스의 정확도에 대한 내용을 포함하는 산출물을 찾고자 할 때, 사용자는 기능적 요구사항 온톨로지의 개념인 “경보서비스_세그먼트”와 비기능적 요구사항 온톨로지 개념인 “Accurateness(정

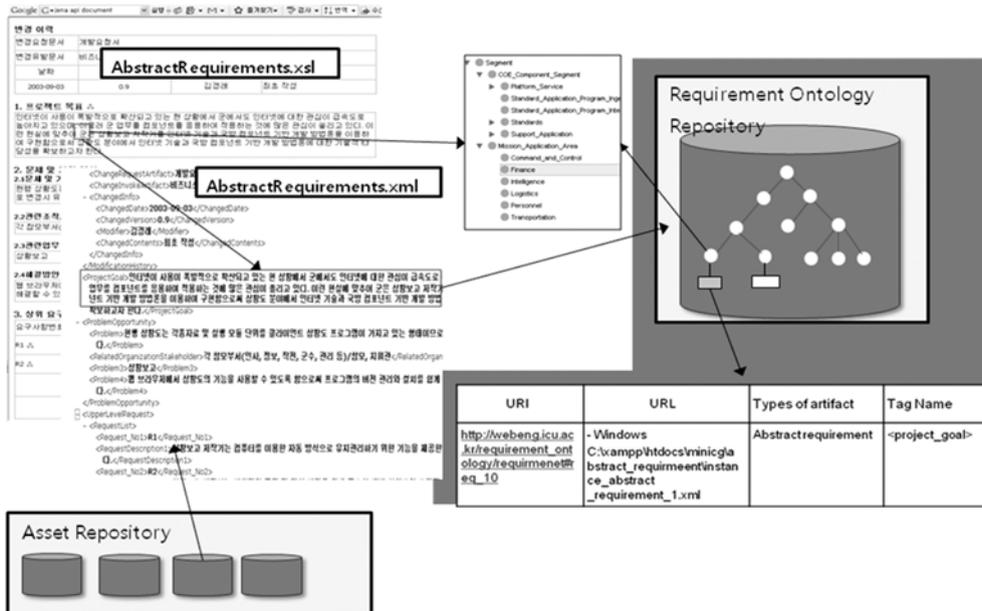


그림 5 산출물의 시맨틱 태깅 과정

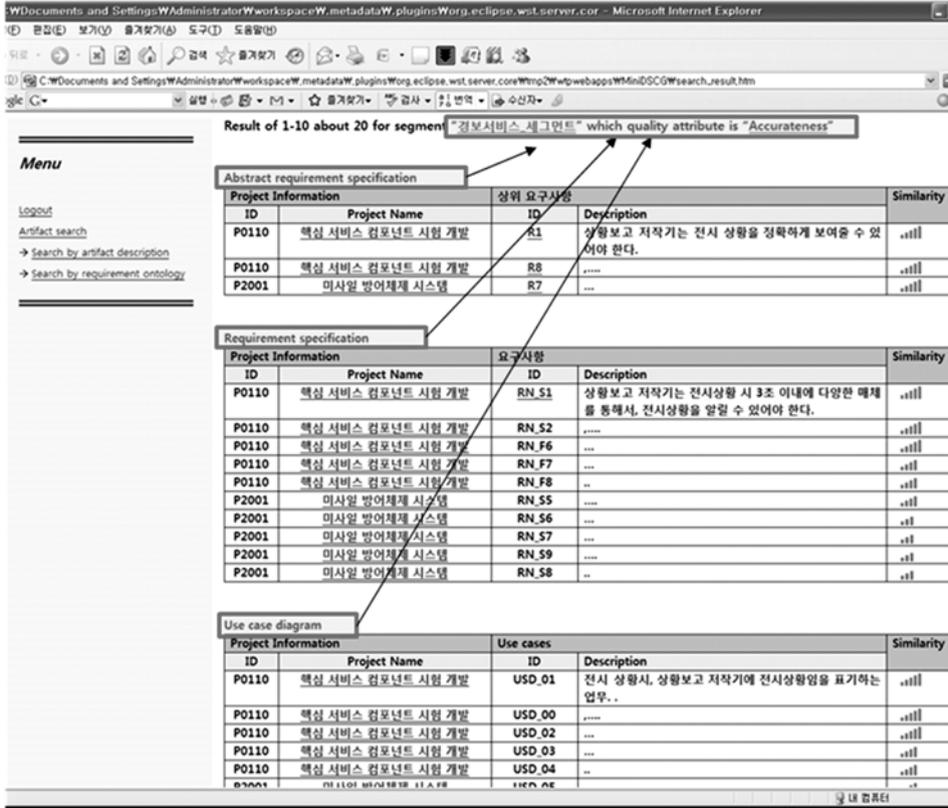


그림 6 요구사항 온톨로지를 활용한 산출물 검색 결과

확도)을 이용하여, 산출물을 검색하게 된다. 이러한 검색의 통해서, 요구사항 추적기는 각 요구사항을 만족하는 산출물들을 종류별로 찾을 수 있게 해준다.

또한 검색된 결과 중 하나를 선택하여 자세한 결과를 확인할 수 있는데 그에 대한 내용은 그림 7과 같다. 그림 7에서는 사용자로부터 선택된 산출물이 속해 있는 프로젝트 정보와 내용을 보여줄 뿐 아니라, 이 산출물에 대해서 기존에 다른 사용자에게 의해 태깅된 요구사항 온톨로지 정보를 보여줌으로써, 사용자가 선택한 산출물이 어떠한 요구사항과 관련된 있는지를 파악할 수 있도록 해준다. 또한 태깅된 온톨로지 개념 각각에 관련 있는 산출물들을 또한 찾아주어, 사용자가 자신에 초기에 찾고자 하는 요구사항 외에 추가적으로 필요한 요구사항이 무엇인지 등을 파악할 수 있도록 도와준다.

다음은 3절에서 제시한 요구사항 추적기를 위한 요구조건과 비교하여 본 연구의 효과를 설명한 내용이다.

- 개발 프로세스와의 독립성: 본 논문의 요구사항 온톨로지 기반 웹 기반의 시맨틱 태깅 기법은 기존 산출물의 작성 방법이나, 개발 프로세스를 변경하는 것이 아니라, 기존의 산출물들에 온톨로지 기반의 태그를

삽입하는 방법을 지원한다. 메타 모델이 개발 프로세스에 종속적이기 때문에, 기존의 메타 모델 기반의 산출물 재사용 및 요구사항 추적 기법의 경우, 특정 개발 프로세스(예, UML 기반 방법론)에만 적용가능하였다. 그러나 시맨틱 태깅 기법은 특정한 개발 프로세스에 의존되지 않는, 적용기법을 지원함으로써, 개발 프로세스와의 독립성을 만족하였다.

- 산출물 검색 결과에 대한 정확성: 요구사항 온톨로지를 기반으로 산출물에 시맨틱스 정보를 삽입함으로써, 의미적인 검색을 가능하였다. 특히 소프트웨어 산출물의 경우, 종류에 따라서 서로 다른 도메인 지식과 다른 종류의 개발자에 의해서 산출물들이 작성되기 때문에 같은 의미를 지니더라도, 다른 표현 방법으로 서술되는 경우가 많다. 예를 들어 "경보 서비스"의 경우, 요구사항 단계에서는 "위급 사항을 알리는 기능"으로 표현될 수 있고, 디자인 단계에서는 "경보 서비스", 그리고 실제 개발되는 코드에서는 "Alarm_Service"등으로 표현될 수 있는데, 기존의 키워드 기반 검색으로는 위와 같은 산출물들을 동시에 찾아낼 수 없었다. 하지만, 온톨로지를 활용할 경우, 위의 내용들은 모두 "경

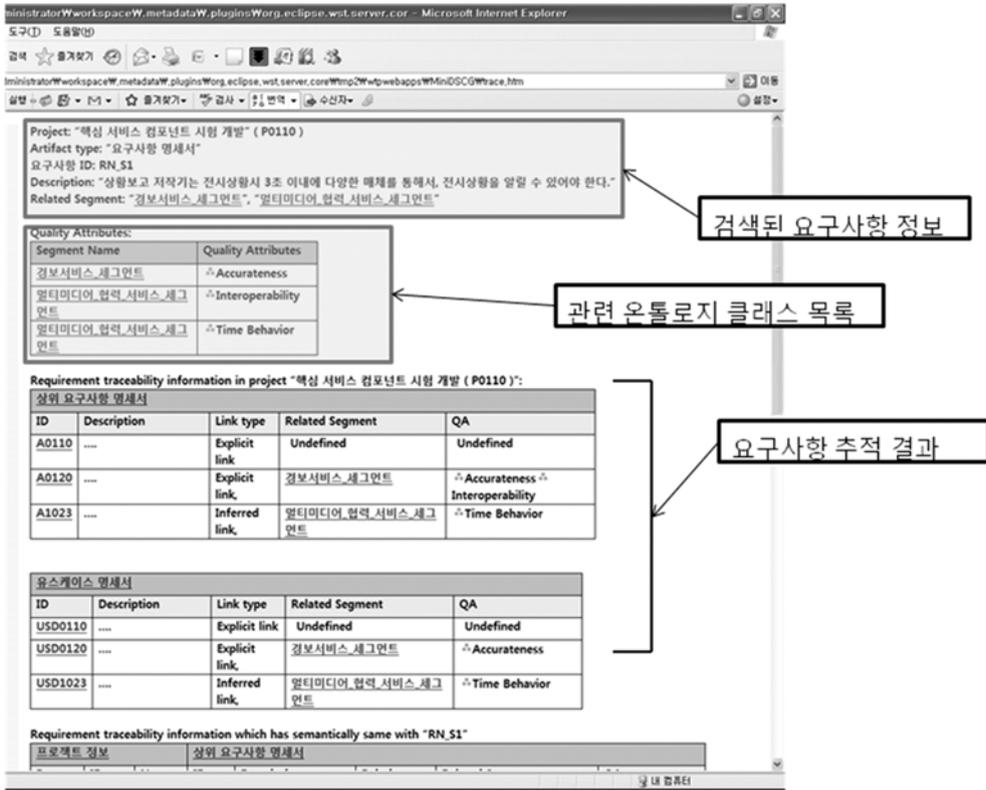


그림 7 검색된 산출물에 대한 요구사항 추적 결과

보서서비스_세그먼트" 라는 기능적 요구사항 온톨로지의 개념으로 표현되기 때문에, 효과적인 검색을 지원하게 된다.

- 요구사항 추적 관계 관리의 편의성: 기존의 사람에 의해서 관리되고 정의되는 요구사항 추적 관계에 비해, 본 논문은 태깅된 온톨로지 정보를 활용하여, 산출물 간의 요구사항 추적관계를 추론할 수 있다. 태깅 자체에도 사람의 노력이 필요하지만, 이전의 요구사항 추적표와 같은 추적 관계 정의가 소수의 요구사항 관리자에 의해 관리되고 정의되었던 반면에, 다수의 개발자 및 사용자들이 시맨틱 태깅에 참여함으로써, 여러 사람들에 걸쳐 시맨틱 태깅의 노력을 분산시킬 수 있다. 이러한 다수의 사람들이 참여하고 평가하는 것을 집단 지성(Collective intelligence)이라고 하는데, 이를 통해서 노력을 분산시킬 수 있을 뿐 아니라, 여러 사람들이 검증하고, 평가함으로써 태깅되는 정보의 질 또한 높일 수 있을 것이다.

7. 결론 및 향후 연구

본 논문은 재사용성 증가를 위한 요구사항추적을 지

원하기 위해서 시맨틱스 기반의 요구사항 추적기의 구조 및 기능과 시맨틱 태깅 방법을 제시하고 프로토타입을 구현하였다. 기존의 다양한 개발 프로세스에 따라서 나오는 산출물들의 요구사항추적을 지원하기 위해, 개발 프로세스에 독립적이고, 산출물에 대한 기존 명세방법의 변화를 주지 않는 방법으로, 이에 따라서 요구사항 온톨로지와 시맨틱 태깅을 활용한 요구사항 추적방법을 제시하였다. 본 방법은 산출물들의 종류와 개발 프로세스에 상관없이 요구사항추적을 지원할 수 있고, 시간이 흐르면서 풍부한 시맨틱스 가지고 따라서 높은 정확성을 가지는 요구사항추적이 가능해질 것이다. 이를 위한 기본 작업으로 요구사항 온톨로지가 구축되어 있어야 하며, 웹 기반의 구현이 필요하다. 향후 연구 방향으로는 현재 제안 하고 있는 요구사항추적 방법이 실제 쓰일 수 있도록 하기 위하여, 먼저 요구사항 온톨로지의 확장이 필요하다 따라서 향후 국방 도메인 내에서의 다양한 기능적, 비기능적인 요구사항들의 개념들을 포함할 수 있도록 요구사항 온톨로지의 확장이 이루어질 것이다. 또한, 기존의 산출물들에 요구사항 온톨로지를 매핑 시키는 작업을 쉽게 할 수 있는 방법을 지원할 계획이다.

참 고 문 헌

- [1] Gotel, O., and A. Finkelstein, "An Analysis of the Requirements Traceability Problem," Proceedings of the First International Conference on Requirements Engineering, Colorado Springs, Colo., pp. 94-101. April 1994.
- [2] Wieggers, "Software Requirements 2nd edition," Microsoft Press.
- [3] Matthias Heindl, Stefan Biffl, "A case study on value-based requirements tracing," Proceedings of the 10th European software engineering conference, 2005.
- [4] Sangim Ahn, Kiwon Chong, "A Feature-Oriented Requirements Tracing Method: A Study of cost-benefit Analysis," Proceedings of International Conference on Hybrid Information Technology - Vol2 (ICHIT'06), 2006.
- [5] Jane Cleland-Huang, Raffaella Settini, "Requirements and specifications: Goal-centric traceability for managing non-functional requirements," Proceedings of the 27th international conference on Software engineering ICSE '05.
- [6] Patricio Letelier, "A Framework for Requirements Traceability in UML-based Projects," 1st International Workshop on Traceability in Emerging Forms of Software Engineering, In conjunction with the 17th IEEE International Conference on Automated Software Engineering, U.K., September 2002.
- [7] G. Antoniol, G. Canfora, "Recovering traceability links between code and documentation," IEEE Transactions on Software Engineering, 28(10): pp. 970-983, 2002.
- [8] Hans-Peter Kruger, P.S. Kritzing, "Latent Semantic Indexing as a Technique for Discovering Traceability Links," In Proceedings Southern African Telecommunications Networks and Applications Conference (SATNAC 2006).
- [9] Oscar López, Miguel A. Laguna, Francisco J. García: "Metamodeling for Requirements Reuse," WER 2002: 76-90.
- [10] Patricio Letelier, "A Framework for Requirements Traceability in UML-based Projects," 1st International Workshop on Traceability in Emerging Forms of Software Engineering, In conjunction with the 17th IEEE International Conference on Automated Software Engineering, U.K., September 2002.
- [11] "Wikipedia," <http://en.wikipedia.org/wiki/Ontology>.
- [12] Rodrigo Perazzo Noll and Marcelo Blois Ribeiro, "Enhancing Traceability using Ontologies," Proceedings of the 2007 ACM Symposium on Applied computing, 2007.
- [13] "Wikipedia," <http://en.wikipedia.org/wiki/Tag>
- [14] Xuefeng Zhu, Zhi Jin, "Inconsistency Measurement of Software Requirements Specification: An Ontology-based Approach," ICECCS05.
- [15] [11]IEEE 1061-1998, "IEEE Standard for a Software Quality Metrics Methodology," IEEE Computer Society Press.
- [16] P. Bourque, R. Dupuis, A. Abran, J. W. Moore, and L. Tripp. "The Guide to the Software Engineering Body of Knowledge," IEEE Software, Vol.16, pp. 35-44, November/December 1999.
- [17] Anya Kim, Jim Luo, Myong H. Kang, "Security Ontology for Annotating Resources," Proceedings of OTM Conferences 2005: 1483-1499.
- [18] A.Egyed, "A Scenario-Driven Approach to Trace Dependency Analysis," IEEE Transactions on Software Engineering, 2003.
- [19] 국방과학연구소, 국방 CBD 방법론 (ADDMe Version 1.1), 2006.9.
- [20] Yonggang Zhang, René Witte, Juergen Rilling, Volker Haarslev, "An Ontological Approach for the Semantic Recovery of Traceability Links between Software Artifacts," IET Software journal.



이 준 기

2006년 한국정보통신대학교 전산학과(학사). 2006년~현재 한국정보통신대학교 공학부 석사과정. 관심분야는 웹 공학, 소프트웨어 공학, 요구공학



조 혜 경

1996년 전북대학교 전산통계학과 석사
2001년 전북대학교 전산통계학과 박사
1996년~1997년 전북대학교 컴퓨터학과 조교. 1998년~1999년 한동대학교 GIS연구소 연구원. 2000년~2003년 한동대학교 전산전자공학부 시간강사. 2004년~2005년 포항공과대학교 컴퓨터공학과 연구원. 2006년~현재 한국정보통신대학교 공학부 연구교수. 관심분야는 온톨로지, 컴포넌트 기반 소프트웨어 개발 프로세스, 프로덕트 라인 공학



고 인 영

1990년 서강대학교 전산학과(학사). 1992년 서강대학교 전산학과(석사). 2003년 미국 Univ. of Southern California(박사). 1993년~1996년 공군사관학교 교관(전임강사). 1997년 미국 USC Information Sciences Institute(ISI) 연구조교. 2003년 미국 USC ISI Postdoctoral Research Associate. 2004년~현재 한국정보통신대학교 공학부 조교수. 관심분야는 소프트웨어공학, 웹공학