

임베디드 소프트웨어 요구사항 관리를 위한 이력관리, 링크, 문서형식 편집 기능 제공 도구 개발

이성환*, 최경희*, 정기현**, 김상중***

*아주대학교 정보통신전문대학원

**아주대학교 전자공학과

***계명문화대학 컴퓨터학부

E-mail: {lethee, khchoi, khchung}@ajou.ac.kr, sjkim@kmcu.ac.kr

Implementing functions of Version Control, Requirement Link, and Document Style Requirement Management Tool

Sean Yee*, Kyunghee Choi*, Kihyun Jung**, Sangjung Kim***

*Graduate School of Information and Communication, Ajou University

**Division of Electronics Engineering, Ajou University

***Computer Faculty, Keimyung Collage University

요 약

본 연구원이 속한 연구실은 임베디드 소프트웨어 테스트 프레임워크(이하 프레임워크)를 구축하고 있으며 임베디드 소프트웨어 요구사항 관리 도구(이하 REED)를 개발하였다. REED 를 통해 작성된 요구사항은 프레임워크에서 테스트 케이스를 만들고 테스트를 수행하는 데 사용된다. 본 논문은 REED 에서 기존에 개발된 요구사항 작성 및 분석 기능에 요구사항 관리 기능을 추가하는 개발에 관한 연구이다.

1. 서론

본 연구에서는 “그래픽 언어를 이용한 임베디드 시스템의 단일 요구사항 모델링 기법[1]” “내장 시스템의 소프트웨어 요구사항 관리 도구 구현[2]”에서 개발된 요구사항 모델링과 설계 기능에 중점을 두었던 기존의 요구사항 관리 도구를 기반으로 요구사항 관리 기능에 중점을 두고 추가적 개발하는 연구를 시행하였다.

기존의 요구사항 관리 도구는 요구사항을 작성하고 모델링을 설계하는데 중점을 두었다. 본 연구에서 개발하는 기능은 요구사항을 작성하고 사용하는 사람들의 요구사항 이해를 돕기 위함이다. 도움을 줄 수 있는 다른 많은 요소들도 있지만 본 연구는 다음 세 가지 기능, 요구사항 이력관리, 요구사항 링크, 문서형식으로 편집 개발에 대한 연구를 중점적으로 하였다.

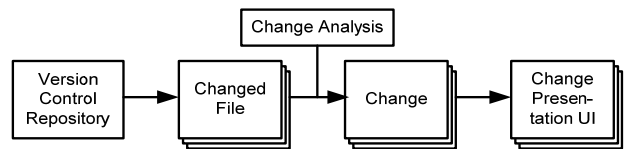
2. 요구사항 이력관리

요구사항은 개발이 진행면서 고객의 요구사항이 변경되거나 실제 구현문제를 수행함에 따라 반드시 변경 된다. 정확하게 기술한 요구사항은 특정 시점의 설계만을 포함한다. 요구사항이 어떻게 변경되었는가를 알 수 있어야 현재 시점에 요구사항에 포함되지 않은 내포된 의도나 변화의 경향을 살펴볼 수 있다. 따라서 요구사항의 이력을 살펴봄으로써 설계자, 개발자, 및 테스터에게 요구사항이 가지는 의미를 더 정확하게 전달하여 이해를 도울 수 있다.

사용자는 특정 시점에서 도구를 통해 버전관리 저

장소에 REED 에서 사용하는 모든 정보를 저장한다. 이력을 관리하기 위해 파일 버전 관리 도구인 Subversion¹을 사용하였다.

- 이전에 저장된 버전 사이/마지막 저장 버전과 현재 작업하는 버전 사이의 변경된 파일을 추출.
- 변경된 파일에서 의미있는 변화를 추출.
- 추출된 변화를 각 의미에 맞게 표현.



(그림 1) 이력관리 데이터의 흐름도

3. 요구사항 링크

REED 에서 요구사항은 계층구조를 통해 요구사항은 상하 관계를 가지며 링크는 이 상하 구조로 표현할 수 없는 요구사항간의 포괄적/구체적 관계를 나타내기 위해 사용했다[2]. 링크의 개념을 좀 더 일반화하여 요구사항간의 '영향을 끼치는 관계'를 표현하기 위하여 개선된 링크 기능을 개발한다.

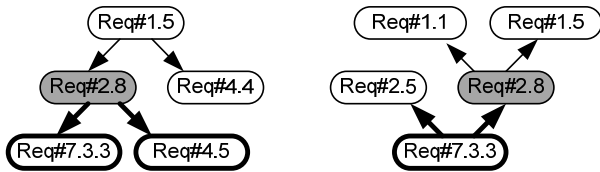
3.1 요구사항 링크

하나의 링크 양 끝에는 요구사항이 연결되어 있으며 한 쪽을 'From 요구사항', 다른 쪽을 'To 요구사항'

¹ CollabNet Inc. <http://subversion.tigris.org/>

이라 하며, 하나의 링크를 통해 'From 요구사항'은 'To 요구사항'에 영향을 미친다고 할 수 있다. 하나의 요구사항에서 링크를 통해 연결된 상대 요구사항을 찾고, 다시 찾은 상대 요구사항의 링크를 통해 연결된 상대 요구사항을 찾아 나가면서 요구사항간의 관련성을 탐색할 수 있다.

다음 (그림 2)은 위의 설명에 대한 예시를 보여준다. 각 요구사항에는 번호가 붙어 있는데, 이 번호는 계층정보를 '!'을 통해 표현하고 있다.



(그림 2) 링크 관계 탐색

(그림 2)에서 왼쪽의 트리는 요구사항 #1.5 를 기준으로 영향을 주는 방향으로 트리를 성장시킨 예제이다. #1.5 는 #2.8 과 #4.4 에 영향을 준다. 계층구조로 보면 #1.5 는 요구사항 #1 의 하위 요구사항이며 따라서 #2.8 이나 #4.4 와는 계층구조로 직접 연결되어 있지는 않다. 하지만 링크를 통해 계층구조에서 표현할 수 없는 요구사항간의 관계를 명시해 줄 수 있다.

(그림 2)에서 오른쪽의 트리는 #7.3.3 을 기준으로 영향을 받는 방향으로 트리를 성장시킨 예제이다. 계층구조로 보면 예를 들어, #7.3.3 은 #2.8 에 영향을 받고 다시 #2.8 은 #1.1 에 영향을 받는다. 이러한 영향 관계는 계층구조로 쉽게 드러나지 않지만, 링크를 통해 어렵지 않게 파악할 수 있다.

3.2 SuspectLink

만약 어떤 하나의 요구사항이 변경된다면, 변경된 요구사항의 링크를 통해 연결된 영향받는 상대쪽의 요구사항에도 당연히 영향을 미칠 것이다. 영향받는 요구사항은 요구사항을 작성하는 사람에게 영향을 주는 요구사항의 변경에 따라 수정할 점이 있을 수 있다는 점을 표현해 줄 필요성이 있다.

링크의 SuspectedMark 라는 특성은 앞의 설명한 점을 표현하기 위해 사용하며 'From 요구사항'의 변경으로 'To 요구사항'이 영향을 받아 변경 확인을 필요로 한다는 뜻한다. SuspectedMark 특성을 해제하는 경우는, 요구사항 작성자가 영향을 받은 요구사항을 수정하거나 수정할 필요가 없음을 확인한 후 해제를 한다.

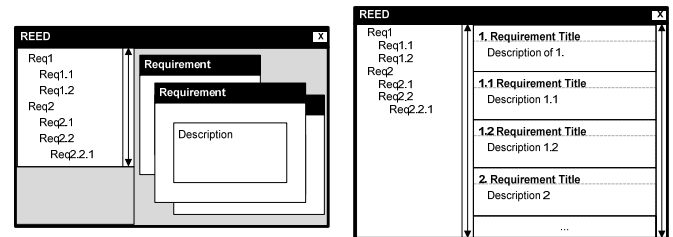
(그림 2)의 'To' 방향 트리에서 요구사항 #2.8 을 수정하면 #7.3.3 과 #4.5 에 영향을 준다. 이를 나타내기 위해 #2.8 을 From 요구사항으로 가지는 링크에 대하여 SuspectedMark 를 설정한다. SuspectedMark 가 설정된 링크에서 To 요구사항에 연결된 요구사항은 From 요구사항의 변경으로부터 영향을 받았으나 아직 작성자로부터 수정 확인을 받지 않았음을 나타낸다. (그림 #)에서 'From' 방향 트리에서도 마찬가지로 요구사항 #2.8 이 수정되어 #7.3.3 에 영향을 미친다는 것을 나타낼 수 있다. 이러한 요구사항을 REED 에서 표현할 때, 요구사항의 이름을 표현하는 곳에 SuspectLink 의

From 요구사항에는 (Changed) To 요구사항에는 (Suspected) 라는 문구를 추가하여 그 특성을 나타내었다.

4. 문서형식의 요구사항 관리

아래 (그림 3)은 기존 REED 에서 제공되는 요구사항 작성화면과 새로 추가된 문서형식 도구사항 작성법의 비교이다. 기존 도구는 요구사항을 작성하기 위해서 왼쪽 트리 탐색 영역에서 계층으로 구성된 요구사항 중 하나를 선택하고 열어서 요구사항창을 통해 요구사항의 세부 내용을 작성하였다. 이 경우 요구사항창이 상당히 크기 때문에 여러 개의 요구사항에 대한 내용 파악이 쉽지 않다. 또한 매번 요구사항창을 열고, 이동하며 파악해야하는 불편한 점이 있다.

문서형식의 요구사항 관리 도구는 하나로 연결된 화면에서 계층구조를 그대로 문서형식으로 보여주며 요구사항의 자연어 기술을 읽거나 작성할 수 있도록 도와준다. 왼쪽 요구사항의 계층 트리구조는 오른쪽 문서의 리스트 구조로 변환되어 보여지고, 추가적으로 요구사항의 이름 앞에 단락 번호로 계층이 표현된다. 계층구조가 변경되면 동시에 왼쪽의 트리구조나 오른쪽의 단락 리스트 구조에 변경되어 나타난다.



(그림 3) 기존 요구사항 편집과 문서 스타일 방법 비교

REED 에서 개발한 문서형식의 요구사항 관리 도구는 요구사항 기술 방식중 하나인 자연어로 기술된 요구사항을 문서형식을 빌어, 쉽게 전체 요구사항을 파악하며, 워드 프로세서와 비슷한 인터페이스로 요구사항을 기술할 수 있도록 돕는다.

5. 추후 연구 과제

요구사항 이력관리에서 그래픽 다이어그램에 대한 변화 분석과 비교 모델을 더욱 개선할 필요, 문서형식의 요구사항 관리는 그래픽 언어로 모델링된 요구사항과 자연어로 기술된 요구사항이 불일치 할 수 있는 단점을 보완하는 기술 개선이 필요하다.

참고문헌

- [1] 오정섭, 이홍석, 박현상, 김장복, 최경희, 정기현, “그래픽 언어를 이용한 임베디드 시스템의 단일 요구사항 모델링”, The KIPS Transactions : Part D, Vol. 15-D, No.4, (Serial Number 121), pp. 505~512, August 2008.
- [2] 박현상, “내장 시스템의 소프트웨어 요구사항 관리 도구 구현”, 아주대학교 정보통신전문대학원 학위논문, 2007.