추적성 맵핑 변경을 쉽게 반영할 수 있는 요구사항 추적성 매트릭스

정세린^o, 강혜진, 이선아
정보과학과, 경상대학교

{rin0608, master6280, saleese}@gnu.ac.kr

A Light-weight Requirement Traceability Matrix for Reflecting Traceability Mapping Changes

요 약

요구사항 추적성을 간편하게 관리할 수 있는 일반적인 방법은 스프레드시트 형태의 매트릭스이다. 이러한 매트릭스는 각 산출물에 대해서 열을 만들고 산출물 항목 간의 맵핑을 행으로 배열한다. 이러한 매트릭스에서는 임의의 산출물을 변경, 세분화, 추가, 삭제할 때 모든 산출물의 관계를 다시 맵핑해야 한다. 노력이 많이 필요하므로, 담당자는 변경이 발생할 때마다 매트릭스를 갱신하기 보다는 필요할 때 전체 매트릭스를 다시 만드는 방식으로 진행하여 추적성 관리의 지속에 어려움을 겪는다. 우리는 산출물의 맵핑 변경을 바로 넣을 수 있으며, 최종적으로 전형적인 추적성 매트릭스 형태의 산출물을 넣을 수 있는 스프레드 시트의 새로운 형태의 추적성 맵핑표를 제안한다. 이러한 추적성 맵핑표는 기존의 간편함을 유지하기 위해 스프레드시트의 형태를 유지하나 맵핑 관계를 분할함으로써 갱신을 용이하게 하고, 자동 매크로를 활용하여 최종적으로 관리자가 보기를 기대하는 전형적인 추적성 매트릭스 표를 자동 생성한다.

1. 서 론

요구사항 추적성은 요구사항을 구현하는 여러 산출물을 연관 짓는 능력으로서 [1], 개발하는 시스템에 요구사항을 얼마나 반영 하였는지를 파악하기 위해서 필요하다. 개발자들은 요구사항 추적성을 수립하기 위해 요구사항 추적성 매트릭스 (Requirement Traceability Matrix, 이하 RTM)를 사용한다. 추적성 매트릭스는 개별 요구사항과 다른 산출물 간의 논리적 맵핑을 기술한다. 개발자들이 추적성 수립을 위해서 간편한 형태의 스프레드시트를 사용하는 경우가 많다. 이때의 전형적인 추적성 매트릭스의 형태는 표 1과 같다.

[표 1] 전형적인 추적성 매트릭스 ([2]에서 가져옴)

요구사항	설계모듈	구현파일	시험항목
RE001	DE003	IM002	TE002
...	...	...	...

표 1은 요구사항, 설계모듈, 구현파일, 시험항목으로의 맵핑을 순차적인 배열로 표현한다 [2]. 이러한 매트릭스는 요구사항의 변경에 따라 차례대로 그 변경을 설계, 구현, 시험에 반영하는 것을 가정한다. 그러나, 소프트웨어 개발 및 진화에서의 변경은 임의의 소프트웨어 산출물에서 발생할 수 있다. 표 1 형태의 매트릭스는 이러한 변경에 취약하다. 이에 개발자는 변경 사항이 발생할 때마다 업데이트하기 보다는 필요할 때 전체 매트릭스를 다시 만드는 방식으로 진행한다 [1,2]. RTM을 위한 많은 도구[3,4]를

개발하였지만 상기의 문제는 여전히 존재한다.

우리는 투자 대비 효과(ROI)의 관점에서 RTM의 개선을 모색한다. 우리는 많은 RTM 도구가 새로 제안되었음에도 여전히 개발자들은 스프레드시트 형태의 RTM을 사용한다는 사실에 주목한다. 이는 작성에 들어가는 노력이 정교화된 도구를 사용하는 것보다 적기 때문일 추정한다. 이러한 관점에서 우리는 스프레드시트에서의 새로운 형식의 RTM을 제안한다. 제안하는 RTM은 개발자가 전체 매트릭스를 다시 만들지 않고, 변경사항이 발생할 때마다 업데이트를 할 수 있도록 도와준다. 또한 점검 시점에서 일관되지 못한 추적성 맵핑을 쉽게 발견할 수 있도록 돕는다.

본 논문의 구성은 다음과 같다. 먼저 2절에서는 관련 연구를 논의한다. 다음 3절에서는 제안 방법을 설명한다. 4절에서는 제안 방법을 적용한 예제를 보인다. 5절에서는 결론과 향후 연구 방향을 논의한다.

2. 관련 연구

요구사항 추적성 관련 연구는 추적성 문헌 서버이, 스프레드시트 형태의 요구사항 구현, 변경에 따른 갱신, 요구사항 관리 도구 개발 등이다.

먼저 서버이 논문으로는 요구사항 지식체계를 구축[5]하거나 로드맵을 구축하는 연구가 있다 [6]. Duarte와 동료들은 요구사항 추적성이 실제 산업계에서 50%미만으로 사용되며, 이는 소프트웨어 개발 시 변화하는 요구사항을 관리하기 어렵기 때문이라고 지적하였다. 이를 개선하고자 체계적인 요구사항 추적성 수립을 위한 추적성 지식체계를 구축하였다[5]. Gotel과

동료들은 서버이를 통해 실질적인 요구사항 추적성 수립 및 적용을 위한 요구사항 추적성 로드맵을 제안하였다 [6]. 또한 추적성에서 고려해야 할 8가지 사항을 명시하였는데, 이 중 본 논문에서는 비용 대비 효과(ROI)에 초점을 두고 연구를 진행한다.

Hussain과 동료들은 협업 개발 프로젝트에서의 요구사항을 관리하기 위한 스프레드시트의 활용을 조사하였다 [7]. 그 결과 파키스탄과 미국, 뉴질랜드와 미국, 캐나다의 협업의 두 가지 프로젝트를 조사하여 3가지 형태의 스프레드시트 활용 형태를 밝혀내었다. 그러나 주어진 현상을 연구했을 뿐, 스프레드시트를 활용하기 위한 방법을 제시하지는 않았다.

김과 동료들은 기존 추적성에서 변경관리 방법을 제시하고 있지 않다고 지적하고 변경 요구사항에 대해서 새로운 ID를 부여하자고 제안하였다 [8]. ID에 변경의 원래의 요구사항을 알 수 있도록 기재함으로써 변경 정도를 기존 연구에 비해 용이하게 추정할 수 있다고 주장하였다. 그러나, 그들의 제시 방법은 기존 추적성 방법에 요구 사항 ID를 변경에 따라 어떻게 추가할 것인가의 개선이었다.

최근 도구 연구들은 산출물 간의 명시적인 관계를 자동 도출하기 위해 데이터 마이닝 기법을 사용했다 [3, 4]. 예를 들어 TraceM 도구는 요구사항 산출물 간의 암묵적인 관계를 탐색하여 오픈 하이퍼미디어를 통해 명시적인 관계를 만든다 [3]. Trustrace는 마이닝과 정보검색 기술을 접목하여 소프트웨어 개정 이력에서 요구사항 추적성을 복구한다 [4]. 이러한 방법들은 낮은 정확도로 인하여 신뢰할 만한 하지 않다.

우리의 연구는 스프레드시트를 사용하여 RTM을 관리하는 새로운 형태와 기법을 제시하는 것으로서, 기존의 도구 개발과는 차별성을 보인다.

3. �핑 변경 가능한 요구사항 추적성 매트릭스

우리는 관리자가 변경된 �핑을 바로 반영할 수 있고, 필요할 때 아직 반영되지 않은 �핑을 기존의 요구사항 추적성 매트릭스 형태로 확인할 수 있는 새로운 스프레드 시트의 요구사항 추적성 매트릭스를 제안한다.

3.1 제안 아이디어

우리가 제안하는 방법에 담긴 아이디어는 다음과 같다. 첫째, 산출물 간 변경된 �핑을 바로 반영할 수 있기 위해서는 기존 한 행으로 배열된 산출물 간의 �핑을 분할해야 한다. 이렇게 분할한 경우 관리자는 변경된 �핑만을 집중하여 스프레드 시트를 갱신할 수 있다. 둘째, 최종적으로 전체 추적성 매트릭스를 생성하여 점검할 때 누락된 추적성의 정보가 어디에 있어야 하는지를 알려주면 된다.

이러한 아이디어에서 출발하여 우리가 제안하는 새로운 매트릭스는 다음과 같다. 첫째, 각 시트는 두 개의 산출물 간의 �핑 관계를 가진다. 예를 들어 각

시트는 요구사항:구현, 설계모듈:구현, 시험:구현 간의 �핑 관계를 표현한다. 둘째, 버튼을 누르면 각 시트의 �핑 관계를 분석하여 전형적인 형태의 RTM을 자동 생성한다. 생성된 RTM에서 내부적 변경이 발생한 셀은 초록색으로 표시하고 추적성 관계가 누락되거나 깨진 셀은 붉은색으로 표시한다.

3.2 제안 스텝

RTM을 자동 생성하는 과정은 �핑 관계가 포함된 시트의 내용을 RTM을 생성하는 시트로 복사 후 변경이 발생한 셀의 색상을 바꿔주는 것으로 수행된다.

- 입력물: 각 시트의 �핑 관계로 각 시트는 각 산출물이 소스 코드와 연결된 �핑을 보여준다.
- Step 1: 각 시트의 �핑 관계를 RTM 생성 시트로 삽입한다. 먼저 요구사항과 구현 간의 �핑 관계를 삽입한다. 다음 설계모듈과 구현 간의 �핑 관계를 삽입한다. 이 때 앞서 복사된 구현 산출물을 기준으로 누락된 구현 산출물이 있으면 추가한다. 마지막으로 시험 산출물과 구현의 �핑 관계를 체크하여 추가로 삽입하며, 누락된 구현 산출물이 있으면 추가한다.
- Step 2: RTM 생성 과정에서 발생하는 요구사항의 변경을 반영하기 위해 변경이 발생한 셀의 색상을 변경한다. 각 시트에서는 산출물의 내부 변경에 대한 처리를 �핑 관계에 기록한다. 이 기록을 바탕으로 RTM 생성 시트로 복사하는 과정에서 변경이 발생한 셀을 초록색으로 변경한다.
- Step 3: RTM이 생성된 후에 버튼을 클릭하여 �핑을 하는 과정에서 �핑의 내용이 없는 셀을 빨간색으로 표시한다. 이는 누락된 �핑을 의미한다.
- 출력물: 출력물은 Step 1, 2, 3을 거쳐 자동 생성된 RTM이다.

해당 출력 RTM은 기존의 �핑 정보를 바탕으로 최대한 추적성을 완성하여 보여준다. 개발자는 이를 검토, 수정하여 RTM을 완성할 수 있다.

4. 데모

그림 1은 본 논문에서 제안하는 요구사항 추적성 매트릭스의 예시이다. 그림 (a)는 각 시트에 포함되어 있는 산출물 간의 �핑 관계와 시트의 내용을 RTM 생성 시트로 복사하기 위한 버튼을 표로 나타낸 �핑 관계표이다. 그림 (b)는 요구사항과 구현 간의 �핑 관계 및 변경사항을 포함하는 시트이고 그림 (c)는 설계와 구현 간의 �핑 관계 및 변경사항을 포함하는 시트이다. 그림 (d)는 우리가 제안한 방식으로 자동 생성된 최종 추적성 표이다. 초록색으로 표시된 셀은 각 시트의 내용을 RTM생성 시트로 복사할 때 각 산출물의 변경 여부에 따라 해당 셀의 변경여부를 표시한 것이다. 붉은색으로 표시된 셀은 모든 �핑 관계를 RTM생성 시트로 복사한 후에 발생한 빈 셀로 산출물 간의

	A	B	C	D	E	F	G
1	X	Y	CopyButton				
2	RE	IM	copy_R				
3	DE	IM	copy_D				
4	TE	IM	copy_T				
5							
6							
7							
8							
9							
10							

(a) 맵핑 관계표

	A	B	C	D	E	F
1	RE	RE변경여부	IM	IM변경여부		
2	RE001		IM001			
3	RE002	c	IM002	c		
4	RE003		IM003			
5	RE004		IM004			
6	RE005	c	IM005	c		
7						
8						
9						
10						

(b)요구사항과 구현의 맵핑 예

	A	B	C	D	E	F
1	DE	DE변경여부	IM	IM변경여부		
2	DE001		IM001			
3	DE002	c	IM002	c		
4	DE006	c	IM006	c		
5	DE004	c	IM004	c		
6	DE008	c	IM008	c		
7						
8						
9						
10						

(c) 설계와 구현의 맵핑 예

	A	B	C	D	E
1	RE	DE	IM	TE	
2	RE001	DE001	IM001	TE001	ColorChange
3	RE002	DE002	IM002	TE002	
4	RE003		IM003	TE003	
5	RE004	DE004	IM004		
6	RE005		IM005		
7		DE006	IM006	TE006	
8		DE008	IM008		
9			IM007	TE007	
10					

(d) 자동 생성된 추적성표

[그림 1] 제안한 요구사항 추적성 매트릭스 예시

맵핑이 누락된 셀을 표시한 것이다.

6. 결론

우리가 제안한 방법은 스프레드시트를 사용하여 RTM을 관리하기 때문에 새로운 툴을 설치하지 않고 사용할 수 있다. RTM은 각 시트에 포함된 구현을 기준으로 한 요구사항, 설계, 시험 간의 맵핑 관계를 RTM 생성 시트에 복사함으로써 자동 생성된다. 변경이 발생한 경우 해당 변경만 개발자가 입력하면 되기 때문에 변경이 발생 하여도 개발자가 새로운 전체 매트릭스를 만들지 않아도 된다. 또한 변경이 발생한 셀은 녹색으로 링크가 누락된 셀은 붉은색으로 표현함으로써 개발자의 검증을 용이하게 하였다.

7. 참고 문헌

- [1] Garcia, Jorge Esparteiro, and Ana CR Paiva. "A Requirements-to-Implementation Mapping Tool for Requirements Traceability." J. of Software. 193-200. 2016
- [2] 박종렬, 류승희, 정세린, 이선아. "소프트웨어 요구사항 진화에서의 추적성 관리의 어려움 조사" 19th Korea Conference on Software Engineering. KCSE, 2017
- [3] Sherba, Susanne A., Kenneth M. Anderson, and Maha Faisal. "A framework for mapping traceability relationships." the 2nd International Workshop on Traceability in Emerging

Forms of Software Engineering, 2003.

- [4] Ali, Nasir, Yann-Gaël Guéhéneuc, and Giuliano Antoniol. "Trusttrace: Mining software repositories to improve the accuracy of requirement traceability links." IEEE Transactions on Software Engineering 39.5 (2013): 725-741.
- [5] Duarte, Ana Marcia Debiasi, Denio Duarte, and Marcello Thiry. "TraceBoK: toward a software requirements traceability body of knowledge." Requirements Engineering Conference (RE), 2016 IEEE 24th International. IEEE, 2016.
- [6] Gotel, Orlena, et al. "The quest for ubiquity: A roadmap for software and systems traceability research." Requirements Engineering Conference (RE), 2012 20th IEEE International. IEEE, 2012.
- [7] Hussain, Waqar, and Tony Clear. "Spreadsheets as collaborative technologies in global requirements change management." Global Software Engineering (ICGSE), 2014.
- [8] 김주영, 류성열, 황만수. "추적테이블을 이용한 요구사항 변경관리 및 추적 효과 연구." 정보처리학회논문지. D 17.4, 271-282. 2010.