

# 테이블 분석방법을 통한 소프트웨어 오류감소방안

한판암\*, 최신형\*\*, 진광윤\*\*\*

\*경남대학교 정보통신공학부

\*\*삼척대학교 컴퓨터응용제어공학과

\*\*\*삼척대학교 컴퓨터공학과

e-mail:pahan@kyungnam.ac.kr

## Software Error Decrease Plan through Table Analysis Method

Pan Am Han\*, Shin Hyeong Choi\*\*, Kwang Youn Jin\*\*\*

\*\*Dept of Computer Engineering, Kyungnam University

\*\*Dept. of Computer Control Engineering, Samcheok National University

\*\*\*Dept. of Computer Engineering, Samcheok National University

### 요약

통합 테스트 단계에서 검사팀에 의해 일반적으로 사용되는 테스트 시나리오는 요구분석 및 설계단계에서 도출된 각종 명세서를 기반으로 하여 작성된다. 본 논문에서는 이런 테스트 시나리오를 작성할 때 오류를 최소화하기 위한 테이블 분석방법을 제시한다. 이를 이용하면 통합 테스트 단계에서 특별한 검토과정 없이 요구분석 및 설계단계에서 얻어진 명세서에 의존해서 테스트 시나리오를 작성하는 방법에 비해 테이블 정보에 대한 오류를 최소화함으로써 테스트 능률향상 뿐 아니라 테스트 비용을 크게 줄일 수 있다. 이를 바탕으로 통합 테스트를 수행한다면 보다 정확한 테스트가 가능하므로 최종산출물인 소프트웨어에 대한 품질을 향상시킬 수 있다.

### 1. 서론

컴퓨터 등장과 더불어 소프트웨어 또한 지속적으로 발전해 왔다. 그로 인해 업무처리는 더욱 고속화되고 더 이상 컴퓨터를 사용하지 않는 업무처리는 상상할 수 없는 현실이 되었다.

사용자가 컴퓨터를 이용함으로써 기대하는 것은 신속한 일 처리와 정확한 결과를 얻을 수 있다는 점이다. 컴퓨터의 신속성은 어떤 알고리즘을 사용하는가에 따라 좌우될 수 있지만 하드웨어의 급속한 발전을 통해서도 충분히 해결되고 있는 추세이다. 그러나 정확성 즉, 소프트웨어의 품질은 적절한 개발 방법론의 사용과 테스트 등의 품질보증 활동과 같은 고품질 소프트웨어 확보를 위한 노력을 통해서만 얻

을 수 있는 것이다.

이와 같이 소프트웨어의 품질을 평가하고 문제점을 발견하기 위한 테스트는 소프트웨어의 생명주기에서 매우 중요하다. 특히, 통합 테스트는 분석단계에서 도출한 현업 요구사항 및 설계명세를 시스템이 만족하고 있는지를 검증하며, 단위 프로그램간의 인터페이스가 설계명세대로 이루어져 기능이 정상적으로 수행되고 있는지를 검토하는 작업으로 현업에서는 주로 요구분석 및 설계단계에서 작성된 각종 명세서를 바탕으로 테스트 시나리오를 작성하여 실시한다. 하지만, 시나리오의 특성상 사용자와 시스템의 상호작용을 사용자가 이해하기 쉽게 단계별로 항목을 수작업에 의해서 자연어로 기술하는 것이 보통이다. 그 결과 요구사항의 누락이나 잘못된 검색

조건의 사용, 테스트 시나리오 작성에 소요되는 많은 시간 등의 문제점을 가진다.

이와 같은 문제점을 해결하고 개발되는 소프트웨어의 품질을 높이기 위하여 본 논문에서는 통합 테스트를 수행할 때 사용되는 테스트 시나리오 작성에서의 오류를 줄이기 위해 요구분석 및 설계단계에서 도출한 각종 명세서에 기초한 테이블 분석방법을 제시한다.

## 2. 관련연구

### 2.1 소프트웨어 품질

소프트웨어 품질은 소프트웨어 기능, 성능 및 품질에 관련된 제반 사항에 있어 소프트웨어가 명시된 요구사항 및 내재된 요구사항을 충족할 수 있는 “소프트웨어 특성의 총체”를 말한다[1].

소프트웨어 품질에 대한 다양한 정의가 있지만 다음과 같은 중요한 세가지 사실을 알아야 한다. 첫째 소프트웨어의 요구사항은 품질이 측정되는 기본이 된다. 요구사항과 일치성 부족은 품질의 빈약성을 나타낸다. 둘째 명시된 표준들은 소프트웨어가 계획되는 방식을 유도하는 개발기준을 정의한다. 만약 기준이 뒤따르지 못하면 품질의 빈약성을 초래한다. 셋째 언급되지 않은 여러 묵시적 요구사항들도 존재한다. 만약 소프트웨어가 명시적 요구사항과는 일치하나 묵시적 요구사항에 일치하지 못하면 소프트웨어의 품질은 의문시된다.

### 2.2 소프트웨어 품질보증

소프트웨어 품질보증은 어떤 실체가 품질 요구사항을 충족시킬 것이라는 적절한 신뢰감을 주기 위하여 품질 시스템에서 실시되고 필요에 따라 실증되는 모든 계획적이고 체계적인 활동이다[2,3,4,5,6,7].

소프트웨어 품질보증 체계는 소프트웨어 제품의 품질 및 생산성 향상을 통하여 제품을 규격화된 요구조건에 일치시키고 충분한 안정성과 신뢰성을 제공하는데 필요한 모든 활동을 사전에 계획하고 조직화시킨 활동 체계이다. 소프트웨어 품질보증은 완성된 제품이 미리 설정된 기술적 요구사항과 일치하는지를 확인하기 위해서 필요한 체계적이면서 계획적인 작업으로서 소프트웨어가 개발되는 과정이나 개발환경 자체에 포함될 수 있는 에러를 수정하고 나아가 에러가 존재하지 않도록 사전에 대비하는 체계

적인 활동을 말한다. 또한, 사용자 요구사항이 제품에 정확히 반영되고 구현되도록 하는 것이라 할 수 있다[2,5].

이 소프트웨어 품질보증의 목적은 그 동안 체계적인 프로세스나 기준이 없이 임의적으로 수행되어 왔던 테스트나 검사를 보다 더 체계적으로 수행하여 초기에 에러를 제거하고, 더 나아가 개발 초기부터 품질 설계를 통해 고품질의 소프트웨어를 생산해 내고자 하는 것이다.

품질보증 활동이란 시스템 구축을 위한 모든 산출물 및 이를 위한 행위가 질서있게 통제됨으로써, 개발 및 변경내용이 체계적으로 일관성 있게 수용되도록 하여 시스템의 품질을 보증하고 생산성을 향상시킨다는 목적을 가지고 있으며, 이를 통해 개발 시스템의 품질확보와 개발 및 이후 유지보수의 품질체계를 수립할 수 있다.

품질보증 활동의 과정을 살펴보면, 프로젝트 내부의 산출물 검토활동, 설계 산출물을 대상으로 하는 감리활동, 개발 시스템을 테스트 계획서에 따른 테스트 시나리오를 통해 결함율을 측정하는 테스트 활동, 프로젝트 종료후의 보증 활동으로 나눌 수 있다.

## 3. 제안하는 테이블 분석방법

소프트웨어 품질보증활동은 전체 시스템 개발과정을 통해 고품질의 소프트웨어를 만들어내기 위해 중요한 활동이다. 특히, 개발 시스템에 대해 테스트 계획서에 기반한 테스트 시나리오를 통해 결함율을 측정하는 테스트 활동은 비용뿐만 아니라 신뢰도 향상에 결정적인 역할을 한다.

본 논문에서는 테스트 활동 중에서도 통합 테스트 단계에서 일반적으로 사용하는 테스트 시나리오를 작성할 때 적용할 수 있는 테이블 분석방법을 제시한다.

### 3.1 테스트 시나리오

통합 테스트는 기능 테스트에서 검증한 시스템의 전체기능을 업무 흐름과의 정합성을 체크하는 것을 목적으로 요구사항에 대한 각종 이벤트 흐름이 올바르게 동작하는지를 검사하며[8,9], 통합 테스트를 수행할 때 일반적으로 테스트 시나리오를 작성하여 검사팀에 의해 실시한다. 즉, 테스트 시나리오에 따라 데이터에 기초하여 모든 테스트 케이스를 실행시켰을 때 테스트 시나리오의 예상결과대로 실행되었거

나 예러항목이 수정된 후 재테스트가 성공적으로 수행된 경우 통합테스트를 종료한다.

### 3.2 테이블 분석절차

본 논문에서 제시하는 테이블 분석방법은 테스트 시나리오를 기술할 때 참조하는 테이블 정보에 대한 정확성을 증가시키는데 중점을 둔다.

일반적으로 테스트 시나리오는 업무명, 프로그램명, 프로그램 ID, 테이블 정보, 이벤트 정보들로 구성되며, 이를 위해 참조하는 문서는 요구분석 및 설계단계에서 도출되는 요구사항정의서, 프로그램대 테이블 상관도, 프로세서 정의서 등이다. 이들 명세서 중 테이블 정보를 포함하고 있는 것으로는 프로그램대 테이블 상관도와 프로세스 정의 테이블이라고 할 수 있다.

그러므로 본 논문에서 제안하는 테이블 분석방법은 프로그램과 테이블 상관도 상에 표시된 프로그램별 관련 테이블 항목과 프로세스 정의 테이블에 있는 관련 테이블 항목값을 비교하여 일치하지 않는 경우에는 관련 테이블 항목값의 수정을 통해서 개발과정에서 작성되는 명세서들 사이의 일관성을 유지할 수 있다. 이는 결과적으로 작성되는 테스트 시나리오의 신뢰도를 높일 뿐만 아니라 최종 시스템의 품질을 높일 수 있다.

그 과정을 정리하면 다음과 같다.

표 1. 테이블 분석절차

단계 1 : 프로세스 정의 테이블에서 프로세스 ID에 해당하는 관련 테이블 정보를 얻음
단계 2 : Step 1에서 구한 테이블에 대해 테이블 정의서를 참고하여 테이블에 해당하는 칼럼들의 형식과 길이 정보를 얻음
단계 3 : 프로그램과 테이블 상관도 상에 표시되어 있는 테이블 정보와 프로세스 정의 테이블의 테이블 정보를 비교·분석함
단계 4 : 일치할 경우 소스코드분석결과를 참고하여 테스트 시나리오를 작성함 일치하지 않을 경우 해당 명세서에 대한 검토 작업을 수행함

일반적으로 요구분석 및 설계단계에서 작성되는 각종 명세서들은 개별적인 목적으로 작성되는 관계로 명세서간 관련 정보의 일관성 검토과정은 생략되

거나 무시하고 넘어가는 경우가 대부분이다. 이런 명세서들의 일관성 검토과정은 테스트 단계에서는 더욱 어려울 뿐 아니라 테스트 시나리오를 작성하는 검사팀은 이를 그대로 수용하여 작성하게 된다.

그러므로 본 논문에서 제시한 테이블 분석방법을 이용하면 통합 테스트 단계에서 작성되는 테스트 시나리오 작성에 오류를 줄임으로써 기존의 요구분석 및 설계단계에서 얻어진 명세서를 검증없이 테스트 시나리오를 작성하는 방법에 비해 테스트 능률향상 뿐 아니라 정확한 테스트가 가능하다. 따라서 결과적으로 최종산출물인 소프트웨어에 대한 품질을 향상시킬 수 있다.

### 4. 적용사례

본 논문에서 제시한 테이블 분석방법을 시설물 종합관리 시스템 개발에서 산출된 각종 명세서에 적용하여 테스트 시나리오를 작성하였다. 시설물 종합관리 시스템은 3가지 업무로 이루어져 있으며, 각 업무는 데이터베이스 입력 및 수정 등의 관리와 조회 업무로 구성된다.

시설물 종합관리 시스템 개발과정에서 도출한 프로그램과 테이블 상관도와 프로세스 정의 테이블 정보를 바탕으로 테이블 분석방법을 수행한 결과 <표 5>에 나열된 대로 총 21개의 단위 프로세스에 대해서 관련 테이블 항목값이 상이함을 알 수 있었다.

이와 같이 통합 테스트를 실시하기 전에 관련 테이블 정보의 불일치 여부를 파악할 수 있다. 이러한 수정과정을 통해서 정확한 테이블 정보를 사용하여 테스트 시나리오를 작성함으로써 테스트에 대한 신뢰성을 높일 수 있다.

표 2. 테이블 불일치 분석표

프로세스 ID	관련 테이블 항목 수		불일치 항목수	누락 테이블
	프로세스 정의 테이블(A)	프로그램대 테이블 상관도(B)		
1.1.1.1	6	7	1	A
1.1.1.2	6	7	1	A
1.1.4.1	6	4	2	A
1.1.4.2	6	4	2	A
1.2.2.1	7	9	2	A, B

프로세스 ID	관련 테이블 항목 수		불일치 항목수	누락 테이블
	프로세스 정의 테이블(A)	프로그램과 테이블 상관도(B)		
1.2.2.2	7	9	2	A, B
1.2.2.3	7	9	2	A, B
1.2.2.4	7	9	2	A, B
1.4.1.5	1	0	1	B
1.4.2.5	3	0	3	B
1.5.4.1	3	1	2	B
1.5.4.2	3	1	2	B
2.1.3.1	4	2	2	B
2.1.4.1	1	2	1	A
2.3.1.5	1	2	1	A
3.1.1.5	2	1	2	A, B
3.1.1.6	2	1	2	A, B
3.1.1.7	2	1	2	A, B
3.1.2.3	3	4	1	A
3.1.4.1	1	0	1	B
3.2.5.1	3	0	4	B
3.4.2.1	1	1	1	A

## 5. 결론

일반적으로 통합 테스트를 수행할 때 이용하는 방법은 테스트 시나리오를 작성하여 검사팀에 의해 실시하는 것이다. 이런 테스트 시나리오 작성에 기본적인 자료가 될 뿐만 아니라 결정적인 영향을 줄 수 있는 것이 요구분석 및 설계단계에서 도출되는 각종 명세서이다. 그러므로 개발되는 시스템의 품질향상을 위해서는 명세서를 바탕으로 작성되는 테스트 시나리오의 신뢰성 향상과 오류를 최소화하기 위한 방안이 필요하다.

본 논문에서는 요구분석 및 설계단계에서 도출된 각종 명세서를 기반으로 하여 통합 테스트 단계에서 작성되는 테스트 시나리오의 오류를 최소화하기 위한 방안으로서 테이블 분석방법을 제시한다. 이를 이용하면 통합 테스트 단계에서 작성되는 테스트 시나리오 작성에 오류를 줄일 수 있으며, 특별한 검토 과정 없이 요구분석 및 설계단계에서 얻어진 명세서에 의존해서 테스트 시나리오를 작성하는 방법에 비해 테이블 정보에 대한 오류를 최소화함으로써 테스트 능률향상 뿐 아니라 테스트 비용을 줄일 수 있다. 이를 바탕으로 통합 테스트를 수행한다면 보다 정확한 테스트가 가능하므로 최종산출물인 소프트웨어에 대한 품질을 향상시킬 수 있다.

## 참고 문헌

- [1] 정보통신부, "소프트웨어 품질보증 기준", 제 1998-123호, 1998.
- [2] 이운정, 최병주, 장우현, "패키지 소프트웨어 인증을 위한 테스트 프로세스", 『정보과학회지』 제19권 제11호, 2001.
- [3] 왕창중, 프로젝트 관리와 소프트웨어공학, 정익사, 1998.
- [4] 정기원, 윤창섭, 김태현, 소프트웨어 프로세스와 품질, 홍릉과학출판사, 1997.
- [5] 천유식, 소프트웨어 개발방법론, 대청미디어, 1995.
- [6] I. Sommerville, Software Engineering, Addison-Wesley, 1995.
- [7] E. Wallmuller, Software Quality Assurance A practical approach, Prentice-Hall, 1994.
- [8] Delamaro ME. Maldonado JC. Mathur AP, "Interface mutation: An approach for integration testing", IEEE Trans. on Soft. Eng, V.27 N.3, pp. 228-247, 2001.
- [9] 최은만 저, '소프트웨어 공학론(개정판)', 사이텍 미디어간, 2001.