

응용시스템 개발 산출물간의 불일치 요소 검출 자동화 도구 설계 및 구현

진 광 윤* · 최 신 형** · 한 판 암***

요 약

분석 및 설계단계의 산출물을 작성할 때는 대부분 정형화와 표준화를 따른다. 그러나 실제 시스템 개발현장에서는 단계별 산출물들의 개별적인 특성으로 인하여 모든 산출물간에는 연속적이며 자동화된 과정을 통해 산출물들이 작성될 수는 없다. 그 결과 작성된 개발 산출물간의 일관성이 유지되지 못하는 관계로 최종 산출물에서 여러 가지 문제가 야기됨을 알 수 있다. 그러므로 본 논문에서는 객체지향 방법론에 따라 개발되는 시스템에 대해 분석 및 설계단계의 산출물간 일관성을 유지하기 위한 방안을 제시하고, 이를 지원하기 위한 도구를 개발한다.

Design and Implementation of an Automated Tool for Inconsistency Detection among Application System Development Products

Kwang Youn Jin* · Shin Hyeong Choi** · Pan Am Han***

ABSTRACT

Normally, formalization and standardization are followed while doing any system analysis and design. But, in actual system development, products can't be written using the automated tool for all products due to their own characteristics. That results the inconsistency among development products and various kinds of problem can occur in the products. In this paper, we present a technique that supports consistency among products of analysis and design phase developed according to object oriented method. And, this paper also presents the implementation tool.

키워드 : 산출물(Product), 일관성(Consistency), 객체지향방법론(Theory of Object-oriented Methodology)

1. 서 론

객체지향 프로그래밍은 객체라는 작은 단위로서 모든 처리를 기술하는 기법으로 모든 처리는 객체에 대한 요구의 형태로 표현되며, 요구를 받은 객체는 내부에 기술되어 있는 처리를 실행한다.

객체지향 방법론에 따라 시스템을 개발할 경우 개발 단계별로 각종 개발 산출물이 작성된다[4, 7, 10]. 그러므로 각종 산출물간의 일관성 유지는 매우 중요하며, 주의를 기울여야 한다[1, 2, 8, 9].

실제 업무를 개발하는 현장에서 산출물간에는 모두가 자동화 과정을 따르는 것은 아니며, 개발자들이 단계별로 작성된 산출물을 참고하여 개발을 진행하는 것이 일반적이다. 이로 인해서 개발 산출물간에는 많은 불일치성이 발생하기

때문에 개발기간이 지연되는 경우가 흔히 발생한다.

이와 같은 문제를 해결하기 위해서 본 논문은 분석 및 설계단계 산출물간의 일관성을 유지하기 위한 방안을 제시하고, 이를 지원하기 위한 도구를 구현한다. 2장은 관련연구로서 객체지향 방법론과 개발산출물에 대해 설명하고, 3장은 산출물간의 일관성 유지 지원체계를 제시한다. 4장에서는 일관성 유지 지원도구를 구현하며, 5장은 실제 사례에 적용한 결과와 분석을 나타내고, 6장에서 결론을 맺는다.

2. 관련 연구

이 장에서는 소프트웨어 개발을 위한 객체지향 방법론, 단계별 개발 산출물과 산출물간 일관성 점검방법을 기술한다.

2.1 객체지향 방법론

소프트웨어를 개발하는데 방법론을 적용한다는 것은 요구분석부터 유지보수에 이르는 모든 과정을 일관성 있게

* 종신회원 : 삼척대학교 컴퓨터공학과 교수
** 정 회 원 : 삼척대학교 컴퓨터제어계측공학과 교수
*** 종신회원 : 경남대학교 공과대학 컴퓨터공학과 교수
논문접수 : 2004년 3월 26일, 심사완료 : 2004년 7월 12일

연계시키는 것을 의미한다. 객체지향 프로그래밍 기법은 시스템을 상호작용하는 객체의 집합으로 묘사한다. 이 방법론에서 점진 반복적인 과정이 중요성을 가지게 된 것도 객체지향 프로그래밍 기법이 갖는 재사용성과 쉬운 유지보수성 때문인 것으로 파악할 수 있다. 이를 이용하면 부분적으로 기능 구현이 바뀐다 해도 그 영향을 추상화된 클래스 내부로만 최소화할 수가 있기 때문이다[4, 7, 10].

2.2 개발 산출물

객체지향 분석 및 설계과정에서 UML을 사용하면 기존의 구조적방법론에 익숙한 개발자에게 개발단계에서 객체지향 모델을 쉽게 활용할 수 있도록 지원하고, 이들 산출물을 이용하면 일관성 있고, 부분적으로 자동화가 가능한 방법으로 각 단계마다의 산출물을 쉽게 재사용할 수 있다.

<표 1> 객체지향 방법론에 따른 단계별 표준 산출물

단계	표준 산출물
분석	현업요구사항 정의서, 유즈케이스 다이어그램, 클래스 목록, 클래스 다이어그램, 시스템 청사진
설계	클래스 다이어그램, 시퀀스 다이어그램, 프로그램 목록, 화면 레이아웃, 보고서 레이아웃, 프로그램 사양서, 프로그램 테이블 상관도, 테스트 계획서, 통합 테스트 시나리오, 시스템 테스트 시나리오, 테이블 목록, 테이블 정의서
개발	소스 코드, 통합 테스트 시나리오(결과), 시스템 테스트 시나리오(결과)
구현	구현 계획서, 데이터베이스, 애플리케이션 시스템, 운영 시스템, 사용자 매뉴얼, 운영 매뉴얼

객체지향 방법론에 따른 개발 산출물은 <표 1>과 같이 분석, 설계, 개발, 구현단계로 구분하여 작성된다. 하지만, 표준 산출물과 실제 작성되는 산출물간에는 개발할 시스템에 따라서 산출물이 생략 및 추가되기도 한다[3, 6, 7, 10, 12, 14].

2.3 산출물간 일관성 점검방법

2.3.1 역공학을 이용한 일관성 점검 방법

기존의 시스템에 새로운 요구사항이나 신기술을 반영하여 변경하고자 할 때, 기존의 시스템에 대한 정확한 이해가 필요하다. 기존의 시스템을 정확히 이해하고, 설계문서와 같은 필요한 정보를 추출하기 위해서 역공학을 이용한다. 역공학을 이용한 일관성 점검 방법은 원시코드로부터 설계모델을 추출하여 기존의 설계단계에서 작성한 설계모델과 비교하여 일관성을 검증하는 방법이다[5]. 이러한 방법은 설계와 구현사이의 일관성을 검증할 수 있다.

2.3.2 제약언어를 이용한 방법

이 방법은 객체모델을 생성할 수 있는 객체모델 작성기

와 작성된 객체모델을 제약언어를 기반으로 검증할 수 있는 객체모델 검증기를 포함한 객체모델 검증시스템을 이용하는 방법이다[2]. 그러나, 현재는 OMT 방법론에 근거한 객체모델만을 지원하고 있기 때문에, 모든 UML 다이어그램의 오류검출과 일관성을 점검할 수 없다.

2.3.3 지식베이스를 이용하는 방법

이 방법은 객체모델, 기능모델, 동적모델을 정형명세화하여 정보를 추출하고, 지식베이스에 담겨있는 추론규칙을 이용하여 분석하는 오류검사방법이다[1, 4]. 하지만, UML에서 제공하는 모든 다이어그램에 대한 관계와 특징을 제공하지는 못하고, 모델들 사이의 일관성 점검을 수행하기에는 제시된 규칙의 완전성 면에서 부족하다.

3. 산출물간 일관성 유지지원체계

3.1 산출물 관리

실제 시스템 개발현장에서는 분석단계에서 발생한 요구사항에 대해서 현업 요구사항 정의서 및 요구사항 추적표를 통하여 관리하는 것이 일반적이다[1, 7-9].

하지만 이 방법 또한 산출물간의 일관성 유지를 위해서는 사용자 요구사항 추적표를 중심으로 산출물간의 일관성 검증이 수작업으로 이루어진다. 이로 인해서 실제 개발현장에서는 많은 산출물간 불일치성이 발견되며, 이로 인해서 전체 개발공정의 지연을 야기시킨다.

그러므로 성공적으로 예정 공정대로 개발 작업을 수행하기 위해서는 분석 및 설계단계에서 도출된 사용자 요구사항 뿐만 아니라 향후 실시될 사용자 테스트 및 시범운영을 통하여 발생하는 사용자 요구사항 등도 체계적이고 지속적으로 관리하여 사용자들이 실제 업무에 활용할 수 있는 시스템이 되도록 해야 한다.

3.2 개발 산출물간 불일치성 분류

본 논문에서는 우선 각종 응용 시스템 개발과정 중 분석 및 설계단계에서 작성된 개발 산출물에 대해 실시한 감리 자료를 바탕으로 불일치 요소가 많이 발생하는 항목을 중심으로 산출물간 불일치 요소를 분류하여 크게 표준 명명 규칙, 유즈케이스, 엔티티 정의, 화면, 클래스, 보고서, 프로그램, 레이아웃 및 테이블 목록 관련 부분으로 나누었다[10, 11, 13].

3.3 산출물간 일관성 유지 프로세스

본 논문에서 제안한 산출물간 일관성 유지 프로세스는 분석 및 설계단계에서 도출된 개발 산출물의 일관성 및 항

후 유지보수의 용이성을 위하여, 일관성 유지 프로세스를 구현하기 위한 전체적인 구조와 관련활동들을 기술하고 있다. 일관성 유지 프로세스는 소프트웨어 개발을 위한 표준 및 절차 매뉴얼에 객체지향 방법론 및 산출물 표준작성 규정 등을 토대로 만들었다. 개발 산출물간의 일관성 유지 프로세스는 (그림 1)과 같이 도식화 할 수 있다.

1. 표준 명명규칙 관련 산출물 비교 프로세스
2. 유즈케이스 관련 산출물간 비교 프로세스
3. 엔티티 관련 산출물 비교 프로세스
4. 화면 관련 산출물 비교 프로세스
5. 클래스 관련 산출물 비교 프로세스
6. 보고서 관련 산출물 비교 프로세스
7. 프로그램 관련 산출물 비교 프로세스
8. 레이어 관련 산출물 비교 프로세스
9. 테이블 관련 산출물 비교 프로세스

(그림 1) 개발 산출물간 일관성 유지 프로세스

(그림 1)에 나타난 내용을 요약해서 살펴보면 아래와 같다.

<표 2> 표준명명규칙 관련 산출물 비교 프로세스

산출물명	비교 항목		작업 절차
표준 명명규칙	업무구분ID		표준명명규칙에서 정의를 기초로 해당 클래스 ID, 클래스명, 프로그램 ID, 프로그램명, 화면ID, 화면명, 보고서ID, 보고서명 사이의 일관성을 확인
	코드	내용	
	산출물구분ID		
	코드	내용	
	산출물 내용식별자 ID		
	부여규칙		
클래스목록	클래스ID	클래스명	
프로그램목록	프로그램ID	프로그램명	
화면목록	화면ID	화면명	
보고서목록	보고서ID	보고서명	

<표 2>는 개발 산출물 중 표준 명명규칙에 관련된 산출물간의 불일치 요소를 검출하기 위한 절차를 나타낸 것이다. 즉, 표준 명명규칙 관련 산출물 비교 프로세스의 활동

내용은 표준 명명규칙에 규정된 작성샘플과 실제 부여된 ID를 비교한다.

<표 3> 유즈케이스 관련 산출물 비교 프로세스

산출물명	비교 항목		작업 절차
유즈케이스 다이어그램	기능명	유즈케이스명	산출물 항목 중 기능명과 유즈케이스명, 시퀀스명 사이의 일관성을 확인
유즈케이스 시나리오	기능명	유즈케이스명	
시퀀스 다이어그램	기능명	시퀀스명	

<표 3>은 개발 산출물 중 유즈케이스에 관련된 산출물간의 불일치 요소를 검출하기 위한 절차를 나타낸 것이다. 즉, 유즈케이스 관련 산출물인 유즈케이스 다이어그램, 유즈케이스 시나리오, 시퀀스 다이어그램 간의 일관성을 비교한다. UML을 이용한 객체지향 개발방법은 요구분석 과정 중에 Use Case와 여기서 산출되는 시나리오를 기반으로 사용자 요구사항을 모형화하여 객체지향 분석모델을 생성하는 것을 원칙으로 한다[15, 16]. 이로 인해서 다이어그램과 테이블 정보와 일관성 검증을 위해서는, 해당 다이어그램에 대해서 정형명세화 변환과정을 실시해야 한다[13].

<표 4> 엔티티 관련 산출물 비교 프로세스

산출물명	비교 항목		작업 절차
엔티티정의서	엔티티명	항목명	엔티티명과 속성명 사이의 일관성을 확인
ERD	엔티티명	속성명	

<표 4>는 개발 산출물 중 엔티티에 관련된 산출물간의 불일치 요소를 검출하기 위한 절차를 나타낸 것이다.

<표 5> 화면 관련 산출물 비교 프로세스

산출물명	비교 항목		작업 절차
화면목록	화면ID	화면명	해당 화면명과 화면ID 사이의 일관성을 확인
화면레이아웃	화면ID	화면명	

<표 5>는 개발 산출물 중 화면에 관련된 산출물간의 불일치 요소를 검출하기 위한 절차를 나타낸 것이다.

<표 6> 클래스 관련 산출물 비교 프로세스

산출물명	비교 항목		작업 절차
클래스목록	구분	클래스ID	클래스 목록 상에 나열된 클래스ID의 일관성을 확인

<표 6>은 개발 산출물 중 클래스에 관련된 산출물간의 불일치 요소를 검출하기 위한 절차를 나타낸 것이다.

<표 7> 보고서 관련 산출물 비교 프로세스

산출물명	비교 항목		작업 절차
보고서목록	보고서ID	보고서명	해당 보고서 ID와 보고서명 사이의 일관성을 확인
보고서레이아웃	보고서ID	보고서명	

<표 7>은 개발 산출물 중 보고서에 관련된 산출물간의 불일치 요소를 검출하기 위한 절차를 나타낸 것이다.

<표 8> 프로그램 관련 산출물 비교 프로세스

산출물명	비교 항목		작업 절차
보고서목록	프로그램ID	프로그램명	해당 프로그램명과 프로그램 ID 사이의 일관성을 확인
보고서레이아웃	프로그램ID	프로그램명	

<표 8>은 개발 산출물 중 프로그램에 관련된 산출물간의 불일치 요소를 검출하기 위한 절차를 나타낸 것이다.

<표 9> 레이어 관련 산출물 비교 프로세스

산출물명	비교 항목		작업 절차
레이어목록	레이어ID	레이어명	해당 레이어명과 레이어ID 사이의 일관성을 확인
레이어정의서	레이어ID	레이어명	

<표 9>는 개발 산출물 중 레이어에 관련된 산출물간의 불일치 요소를 검출하기 위한 절차를 나타낸 것이다.

<표 10> 테이블 관련 산출물 비교 프로세스

산출물명	비교 항목		작업 절차
테이블목록	테이블ID	테이블명	해당 테이블명과 테이블ID 사이의 일관성을 확인
테이블정의서	테이블ID	테이블명	

<표 10>은 개발 산출물 중 테이블에 관련된 산출물간의 불일치 요소를 검출하기 위한 절차를 나타낸 것이다.

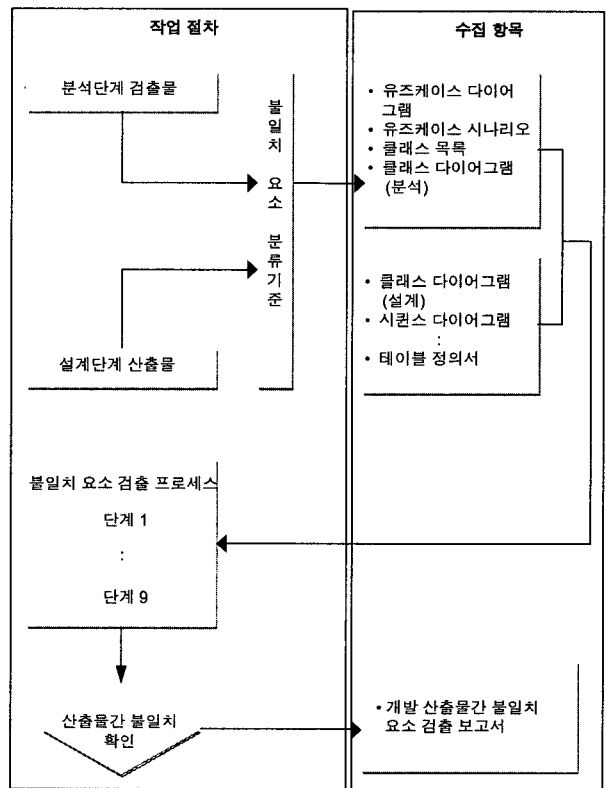
4. 일관성 유지 지원도구의 구현

이 장에서는 분석 및 설계단계에서 개발 산출물간의 일관성을 유지할 수 있도록 불일치 요소를 검출할 수 있는 방안을 제안한다.

4.1 불일치 요소 검출절차

(그림 2)는 개발 산출물로 구성된 정보를 가지고 본 논문에서 제안한 개발 산출물간 일관성 유지 지원방안을 작업 절차, 수집항목으로 나누어진다.

우선적으로 분석 및 설계단계에서 작성된 개발 산출물 리스트에서 불일치 요소 분류기준에 따라서 해당 산출물을 구한다. 다음은 앞의 (그림 1)에서 제시된 개발 산출물간 불일치 요소 검출 프로세스 9단계를 통해 분석 및 설계단계 산출물간 불일치 요소 정보를 수집한다. 이 정보를 개발 산출물간 불일치 요소 검출보고서로 작성한다.

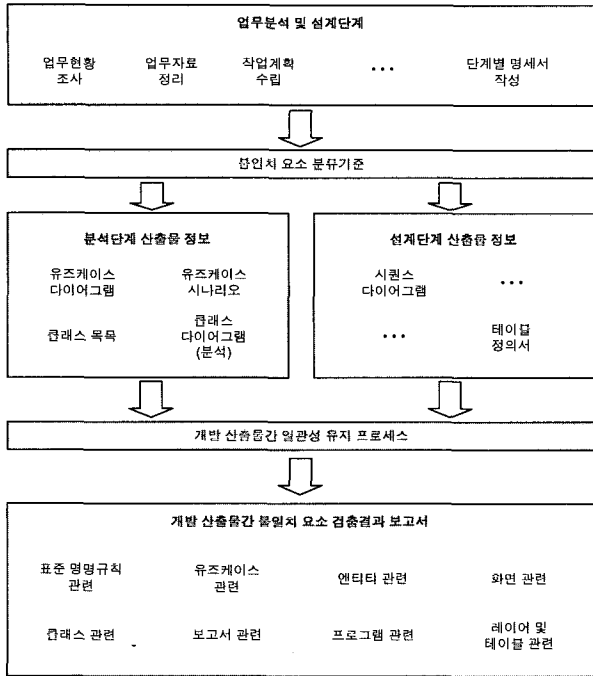


(그림 2) 일관성 유지 지원도구의 작업진행도

4.2 일관성 유지 지원도구의 구현

본 논문에서 제안한 개발 산출물간 불일치 요소 검출절차에 따른 일관성 유지 지원도구는 (그림 3)과 같이 구성된다. (그림 3)에서 알 수 있듯이 분석 및 설계단계 산출물 중 본 논문에서는 감리 수행 중 불일치 요소로 많이 지적되는 산출물들에 대해 불일치 요소 검출절차를 실시한다.

분석 및 설계단계의 표준 산출물 중 명칭과 ID의 불일치 요소가 발생할 수 있는 항목(분석단계 산출물 : 4개, 설계단계 산출물 : 14개)을 입력받은 다음, 4.1절의 불일치 요소 검출절차(단계 1~9)에 따라 단계별로 불일치 요소를 찾아내어 보고서 파일로 저장한다.

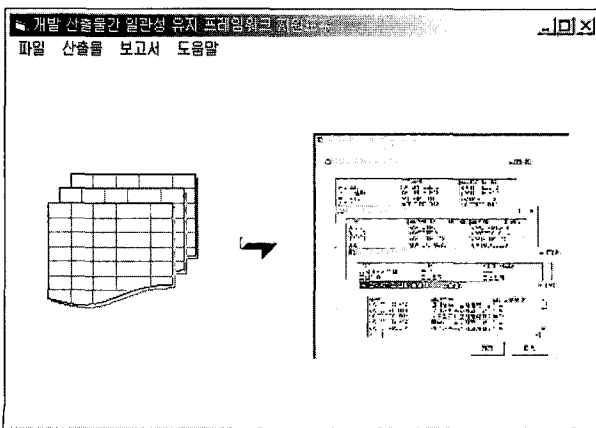


(그림 3) 일관성 유지 지원도구의 구조

<표 11>은 본 논문에서 구현한 개발 산출물간 일관성 유지 지원도구의 모듈을 나타낸다.

<표 11> 개발 산출물간 일관성 유지 지원도구의 구성

순번	폼 명칭	프로그램 기능
1	main_menu.frm	개발 산출물간 일관성 유지지원 도구 메인 폼, 변수선언, 메뉴바 구성
2	analysis_input.frm	분석단계 산출물 입력(유즈케이스 다이어그램 등 4개 산출물)
3	design_input.frm	설계단계 산출물 입력(클래스 다이어그램 등 14개 산출물)
4	report_create.frm	개발 산출물간 일관성 유지지원 도구 보고서 생성
5	report_print.frm	개발 산출물간 일관성 유지지원 도구 보고서 출력



(그림 4) 일관성 유지 지원도구의 초기화면

(그림 4)와 같이 개발 산출물간 일관성 유지 지원도구의 주메뉴는 산출물, 보고서로 구성되어 있다. 산출물 메뉴는 분석단계 산출물 입력, 설계단계 산출물 입력이라는 서브 메뉴로 구성되며, 보고서에 대한 서브 메뉴는 불일치 요소 검출과정, 불일치 요소 검출 결과보고서 출력 기능을 포함한다.

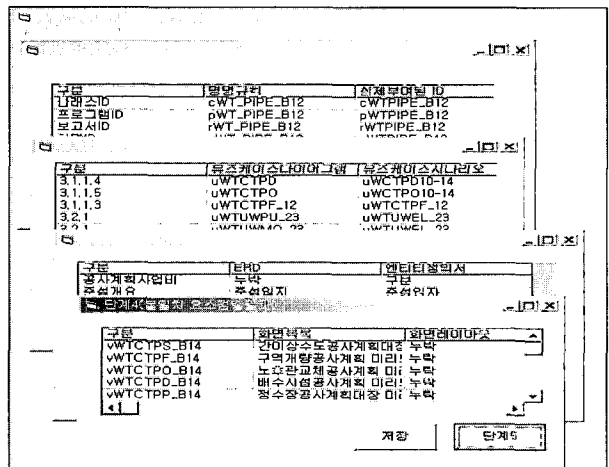
5. 적용 사례

본 연구를 통해서 개발한 개발 산출물간 일관성 유지 지원 도구를 S사의 시설물 종합관리 시스템 개발에 적용하여 불일치 요소를 검출하는 과정을 고찰한다.

5.1 개발 산출물간 불일치 요소 검출결과

개발 산출물간 일관성 유지지원 도구는 우선 표준 명명 규칙을 비롯하여 분석 및 설계단계에서 생성된 개발 산출물들을 입력받는다. 그 다음 개발 산출물간 불일치 요소 검출절차에 따라 불일치 요소를 9단계에 걸쳐 검출한다.

(그림 5)는 단계별로 진행되는 불일치 요소 검출과정을 나타내고 있으며, <표 12>~<표 20>은 적용 산출물들에 대해 검출된 불일치 요소를 표로 정리한 것이다.



(그림 5) 개발 산출물간 불일치 요소 검출과정

<표 12> 표준 명명규칙 관련 산출물간 불일치 요소

구분	표준에 작성된 샘플	실제 부여 ID
클래스ID	cWT_PIPE_B12	cWTPIPE_B12
프로그램ID	pWT_PIPE_B12	pWTPIPE_B12
보고서ID	rWT_PIPE_B12	rWTPIPE_B12

<표 12>는 입력받은 표준 명명규칙과 실제 산출물 상에 부여된 각종 ID 사이의 불일치 요소를 나타낸 것으로 형식이 일치하지 않음을 보여준다.

〈표 13〉 유즈케이스 관련 산출물간 불일치 요소

구 분	유즈케이스 다이어그램	유즈케이스 시나리오	시퀀스 다이어그램
3.1.1.4	uWTCTPD	uWCTPD10-14	uWTCTPD

〈표 13〉은 해당 기능명 3.1.1.4에 대해 유즈케이스 관련 산출물간에 서로 상이하게 표현하고 있음을 보여준다.

〈표 14〉 엔티티정의 관련 산출물간 불일치 요소

구 분	ERD	엔티티정의서
공사계획사업량	누락	사업량3
하수시설물유지보수이력	공사명	누락

〈표 14〉는 해당 엔티티명(예, 공사계획사업량)에 대해 엔티티 정의서에는 사업량 3으로 기입되어 있으나 ERD에는 누락되어 있음을 보여준다. 또한, 불일치 요소도 있음을 알 수 있다.

〈표 15〉 화면관련 산출물간 불일치 요소

구 분	화면목록	화면레이아웃
vWTCTPS_B14	간이상수도공사계획 대장 미리보기창	누락
vSWDDAY_B12	준설일지정보창	준설일지대장창

〈표 15〉는 해당 화면 ID에 대해 화면 목록과 화면레이아웃 상의 화면명이 다르거나 누락되어 있음을 보여준다.

〈표 16〉 클래스 관련 산출물간 불일치 요소

구 분	클래스 ID	구 분	클래스 ID
27	cWTWUTL_E20	24	cWTWUTL_B20

〈표 16〉은 같은 종류의 클래스 간에 표기가 상이함을 보여준다.

〈표 17〉 보고서 관련 산출물간 불일치 요소

구 분	보고서목록	보고서레이아웃
1	rWTCTPS_B12	rWTCTPS_B14
2	rWTCTPF_B12	rWTCTPF_B14

〈표 17〉은 동일한 보고서명에 대해 해당 ID가 상이하게 표시되어 있음을 보여준다.

〈표 18〉 프로그램 관련 산출물간 불일치 요소

구 분	프로그램목록	프로그램사양서
pWTCTPF_B14	구역개량공사계획 미리보기창	구역공사계획대장 미리보기창
pSWDBRF_B11	준설개요목록창	pSW_DBRF_B11

〈표 18〉은 동일한 프로그램 ID에 대해 프로그램명이 다

르게 기입되어 있음을 보여준다.

〈표 19〉 레이어 목록 관련 산출물간 불일치 요소

구 분		목 록	설계서
레이어	상수	간이상수도	WTL_SWSA_AS
	공통	도로중심선	BML_RDCT_SL*
			WTL_SWSA_AM
			BML_RDCT_SL*

〈표 19〉는 레이어의 목록이 상이한 산출물간의 불일치된 요소들을 보여주고 있다

〈표 20〉 테이블 목록 관련 산출물간 불일치 요소

구 분		목 록	설계서
테이블	상수	구역 계량정보	WTT_BLK_MA
			WTT_BLK_DT

〈표 20〉은 테이블의 목록이 상이한 산출물간의 불일치된 요소들을 보여주고 있다

5.2 적용결과분석

본 논문에서 구현한 개발 산출물간 일관성 유지 지원도구를 시설물 종합관리 시스템 개발과정에서 작성된 개발 산출물에 적용한 결과, 분석 및 설계단계를 완료한 시점에서의 개발 산출물간에는 불일치 요소가 〈표 21〉과 같이 검출되었다.

〈표 21〉 적용결과 분석표

불일치 요소 분류기준	불일치 요소
표준 명명규칙 관련 산출물	4
유즈케이스 관련 산출물	5
엔티티 정의서 관련 산출물	5
화면 목록 관련 산출물	8
클래스 목록 관련 산출물	1
보고서 목록 관련 산출물	7
프로그램 목록 관련 산출물	6
레이어 목록 관련 산출물	3
테이블 목록 관련 산출물	1
합 계	40

분석 및 설계단계의 산출물에 대해 실시한 개발 산출물간 불일치 요소 검출결과 총 40개의 개발 산출물간 불일치 요소가 검출되었다.

이와 같은 개발 산출물간 불일치 요소가 검출된 이유는 다음과 같이 정리할 수 있다. 분석 및 설계단계에서 개발 산출물은 개발팀에 의해 수작업으로 산출물간 비교·분석 과정을 통해 작성되며, 개발 산출물의 특성상 동시에 작성될 수는 없기 때문이다. 특히 대규모 프로젝트 수행과 같이 소프트웨어에 관련된 산출물의 양이 많을 경우 이러한 문제가 더욱 심각하다. 이처럼 개발 산출물을 작성하는데 표

준 표현방법론에 기초하여 작성하더라도 개발 규모가 커짐에 따라 단계별로 작성되는 개발 산출물간의 일관성을 완벽히 유지하는 것은 불가능하다. 그러므로 본 논문에서 개발한 개발 산출물간 일관성 유지 지원도구를 이용한다면 기존의 검사팀에 의해 수행되는 방법보다 정확성이 높아 검사시간을 상당히 단축할 수 있다.

<표 22>는 기존의 수작업 방법과 본 논문에서 제안한 방안을 비교한 것이다. 수작업으로 진행되는 기존의 방법에 비해서 본 논문의 산출물 간 일관성 유지 지원도구를 사용할 경우 검출된 불일치 요소 수가 40개 많으며, 자동화 과정을 통해서 빠른 검사시간 내에 분석 및 설계단계에서 작성된 개발 산출물의 불일치 요소를 검출할 수 있다. 또한, 명세서와 다이어그램 부분에 대한 일관성 점검이 가능함을 알 수 있다. 본 논문에서 제안한 방안은 검출 전 불일치 요소분류기준에 따르는 산출물에 대해 일관성 점검을 하므로 제약언어와 지식베이스를 이용한 방법에 비해 효율성이 높고 점검 시간이 단축할 수 있다.

<표 22> 기존 방식과의 비교

비교항목 방법	불일치 요소 수	지 원 단 계	일관성 점검 대상	분 류 기 준	검 사 시 간
기존의 방안 1 (수작업)	α 개	분석, 설계	명세서, 다이어그램	×	길 다
기존의 방안 2 (제약언어, 지식베이스)	-	분석, 설계	명세서, 다이어그램	×	짧 다
제안 방안	$(\alpha + 40)$ 개	분석, 설계	명세서, 다이어그램	○	짧 다

6. 결 론

실제 시스템을 개발하는 현장에서는 분석 및 설계단계를 거치면서 단계별로 해당 산출물들을 작성한다. 단계별로 작성되는 산출물들은 개별적인 특성으로 인해 모든 산출물간에 자동화된 도구를 사용하여 작성될 수는 없다. 이로 인해 개발 현장에서는 대부분 개발팀이 개별적으로 산출물을 작성하고 있으며, 개발진행과정상 이전 단계에서 작성된 산출물을 참고하여 단계별로 산출물을 작성한다.

그러므로 시스템 개발현장에서는 모든 산출물간에는 연속적이며 자동화된 과정을 통해 산출물들이 작성될 수는 없다. 그 결과 작성된 개발 산출물간의 일관성이 유지되지 못한다.

본 논문에서 제시한 개발 산출물간 불일치 요소 검출과정에 따른 개발 산출물간 일관성 유지 지원도구를 이용하면 기존의 방식으로 개발 산출물을 작성할 때의 여러 가지 문제점을 다음과 같이 개선할 수 있다. 그리고 출력물인 개발 산출물간 불일치 요소 검출 결과보고서가 소프트웨어 개발과정 중에 산출물간 일관성 유지 지원해주는 역할을 한다. 이를 바탕으로 응용 소프트웨어를 개발한다면 보다 정

확한 개발 산출물 작성이 가능하며, 개발과정 중의 수정 및 테스트나 유지보수에 소요되는 시간을 상당히 줄일 수 있다. 결과적으로 최종 산출물인 소프트웨어에 대한 품질을 향상시킬 수 있다.

참 고 문 헌

[1] I. Sommerville, 'Software Engineering,' Addison -Wesley, 1995.
 [2] E. Wallmuller, 'Software Quality Assurance A practical approach,' Prentice-Hall, 1994.
 [3] Martin Fowler, Kendall Scott, 'UML Distilled Second Edition : A Brief Guide to the Standard Object Modeling Language,' Addison-Wesley, 2000.
 [4] D. Kung, P. Hsia and J. Gao(Eds), "Testing Object-Oriented Software," IEEE Computer Society, 1998.
 [5] Roger S. Pressman, "Software Engineering," McGraw-Hill International edition, 1997.
 [6] Grady Larsen, "Designing Component-Based Frameworks using Patterns in the UML," Communications of th ACM, 1999.
 [7] 유혜영 역, '구조적 소프트웨어 공학과 객체-지향 소프트웨어 공학', 이한출판사, 2001.
 [8] 왕창중, '프로젝트 관리와 소프트웨어공학', 정익사, 1998.
 [9] 정기원, 윤창섭, 김태현, '소프트웨어 프로세스와 품질', 홍릉과학출판사, 1997.
 [10] 최은만 저, '소프트웨어 공학론(개정판)', 사이텍미디어간, 2001.
 [11] 양해술, "분석단계 산출물에 대한 품질평가툴킷의 설계 및 구현", 정보처리논문지, 제4권 제7호, 1997.
 [12] 김치수, 진영진, "객체지향 분석의 완전성과 일관성 검증을 위한 툴의 설계", 정보처리논문지, 제4권 제10호, 1997.
 [13] 강문설, "구조적 분석의 산출물로부터 UML 클래스 다이어그램으로 변환방법", 정보처리학회 소프트웨어공학연구회지, 제1권 제2호, 1998.
 [14] 이원우, 박수용, 류성열, "객체지향 어플리케이션 개발을 위한 UML기반의 요구공학 프로세스", 제1회 한국소프트웨어공학 학술대회, 1999.
 [15] 임봉춘, 신인철 역, UML입문, 홍릉과학출판사, 1999.
 [16] 정기원, UML 객체지향 분석·설계, 홍릉과학출판사, 2000.

진 광 윤



e-mail : kyjin@samcheok.ac.kr

1984년 서울산업대학교 전자계산학과(학사)

1987년 건국대학교 전자계산학과(석사)

2004년 경남대학교 컴퓨터공학과(박사)

1990년~현재 삼척대학교 컴퓨터공학과

교수

관심분야 : 소프트웨어 품질관리 및 신뢰성, 정보통신보안, 임베디드 시스템 등



최 신 형

e-mail : shchoi@samcheok.ac.kr
1993년 울산대학교 전자계산학과(학사)
1995년 경남대학교 전자계산학과(석사)
2002년 경남대학교 컴퓨터공학과(박사)
1995년~1998년 해군사관학교 전산과학과
전임강사

2003년~현재 삼척대학교 컴퓨터 제어계측공학과 전임강사
관심분야 : 소프트웨어 테스트 및 품질평가, 신뢰도 분석, 컴포
넌트 기반 개발방법론 등



한 판 암

e-mail : pahan@kyungnam.ac.kr
1969년 동국대학교 졸업
1975년 동국대학교 경영대학원(경영학석사)
1989년 명지대학교 대학원(공학석사)
1992년 인천대학교 대학원(경영학박사)
1980년~현재 경남대학교 공과대학 컴퓨터
공학과 교수

관심분야 : 소프트웨어 품질관리 및 신뢰성, 소프트웨어 개발환경,
정보공학 등