

컴포넌트 기반 개발(CBD) 프로젝트 감리에 대한 사례 비교 연구

전순천*, 박기호**, 조남재**

*㈜이탐스, **한양대학교

A Comparative Case Study of Audit Criteria in Component Based Development Project

Jeon, Soon Cheon*, Park, Ki Ho**, Cho, Nam Jae**,

*Etoms Inc., **Hanyang University

E-mail : scj1472@hanmail.net, seaport77@lycos.co.kr, njcho@hanyang.ac.kr

요 약

폭포수 모델 개발 생명주기를 근간으로 한 전통적 프로그램 개발방법론과 달리, 부품 형태의 컴포넌트를 조립하여 개발하는 방법론이 확산되고 있다. 컴포넌트 기반 개발(Component Based Development)의 감리에는 전통적 개발방법에 근간을 둔 감리기준이 아닌 컴포넌트 기반 개발방법에 적합한 감리가 실시되어야 한다.

본 연구에서는 전통적 개발방법론 근간의 감리기준과 CBD 적용 프로젝트 사례 분석을 통하여 감리기준의 개선점 여부를 탐색하고자 하였다. 연구 사례로는 정부 공공프로젝트와 금융기관의 차세대정보시스템 구축 프로젝트를 선정하였다. 기존의 정보시스템감리 기본점검표와 사례에서의 감리검토항목을 비교 분석한 결과 감리검토항목의 개선점을 발견할 수 있었다. 사례분석의 결과로 CBD 프로젝트에 적합한 감리검토항목을 제시하였으며, 연구결과는 소프트웨어 제품 개발의 효율성 제고 및 품질향상에 이바지 할 것이다.

1. 서론

최근 e비즈니스의 기회가 확대되면서 효과적인 정보시스템 개발을 위해 CBD 개발방법론의 필요성이 대두되었다. 정부에서도 S/W 컴포넌트 활성화 정책(1999년 7월 “컴포넌트 기술 개발계획”을 정통부에서 발표)에 따라 주요 사업에는 컴포넌트 기반기술의 적용을 국가적으로 권장하고 있다.

소프트웨어 개발방법론이 본격적으로 적용되기 시작한 1990년대 초부터 10여 년 동안 폭

포수(단계적) 개발생명주기를 가진 구조적인 방법론과 정보공학을 기반으로 한 소프트웨어 생산기술이 발전하였다. 1990년대 말 이후 소프트웨어 개발 프로젝트가 대형화되고 웹 기반의 개방형 아키텍처가 확산되면서 폭포수 개발생명주기의 구조적 방법론 및 정보공학을 기반으로 한 소프트웨어 생산기술에 한계가 드러나게 되어 반복적 개발생명주기와 아키텍처 중심 설계기술을 기반으로 한 컴포넌트 기반 소프트웨어 공학이 대두 되었다 [4].

CBD(Component Based Development) 방법은 기존의 사상, 기법, 도구와는 다른 접근방식을 요구하는 개발방법이다. CBD 방법의 적용은 프로젝트의 성격 및 관리에 있어 많은 변화를 초래하며, 따라서 개발방법론의 변화에 따라 정보시스템 감리 방법도 달라져야 할 것이다. 그 주요 이유는 다음과 같다.

첫째, 기존 감리기준에 대한 보완의 필요성: 기존에 적용되는 개발방법론과는 그 사상, 활동, 산출물 및 기법의 관점에서 전혀 다른 개발방법론을 적용하게 되므로, 이들 활동의 수행 확인 및 산출물의 품질평가 방식도 달라질 필요가 있다. 둘째, 감리 및 품질평가의 어려움: CBD 방법을 적용하는 개발방법론에서는 해당 업무 및 시스템에서 필요한 컴포넌트를 파악하는 것이 중요하며, 이는 시스템에서 요구되는 기능을 묘사하는 것보다 훨씬 더 많은 노력을 요한다. 따라서 이와 같은 활동의 수행 및 결과물의 품질평가를 체계적으로 수행하기 위해서는 개선된 기준이 필요하다. 셋째, CBD방법에 대한 감리지침의 부재: 현행 정보시스템 감리에 있어서 Method/1에 입각한 감리지침과 객체지향 개발 감리지침은 마련되어 있으나, CBD 방법에 따른 정보시스템 감리지침에 대한 연구가 미미한 실정이다. 넷째, CBD 개발 프로젝트 관리 및 통제의 어려움: CBD와 기존 방법론 사이에는 프로젝트 과정에서 수행되는 태스크가 상이하고, 이에 적합한 프로젝트 관리와 통제 활동이 필요하다.

CBD를 적용한 대규모 정보시스템 개발 프로젝트가 증가하고 있으나, CBD 방법의 적용은 프로젝트에 관련된 위험과 불확실성을 크게 증대시킬 가능성을 안고 있다. 아울러 CBD 개발 방법은 그 속성상 매우 추상적인 접근방법이므로 개발노력의 타당성 및 적정성을 평가하기 위해서는 새로운 접근방법을 필요로 한다.

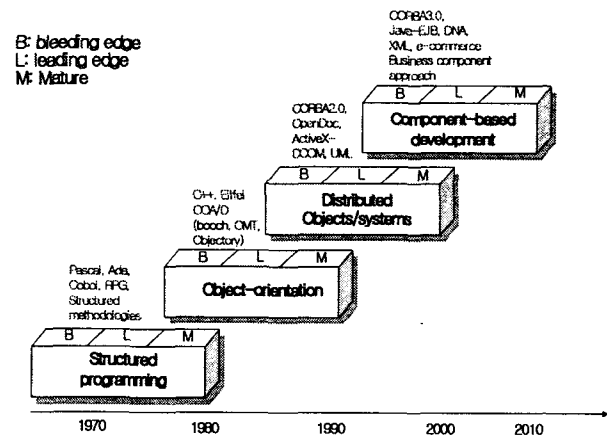
본 연구의 목적은 CBD의 특성과 장점을 활

용하여 개발, 운영되고 있는 두 가지 정보시스템 사례의 비교 분석을 통하여 CBD 프로젝트 감리 시 적절한 접근방식을 도출해 보는 것이다.

2. CBD 방법론

2.1 CBD의 등장 배경

소프트웨어 개발 방법론은 그림1과 같이 1970년대의 구조적 방법론에서 시작하여 정보공학 방법론, 패키지통합 방법론, 객체지향 방법론, 클라이언트/서버 방법론, 모델기반 방법론, 컴포넌트기반 방법론으로 발전해 왔다[6].



[그림 1] 소프트웨어 개발의 진화

소프트웨어 개발 수준 발전의 주요 동기요인으로는 구조적 복잡성과 생산성 향상을 들 수 있다. 1970 년대에 소프트웨어의 복잡성이 증가함에 따라 그 해결책으로 절차적인 방법론과 데이터 중심 모델링이 연구되었다. 그러나 방법론만으로는 소프트웨어 생산성 문제를 해결할 수는 없었다. 따라서 1980 년대에 재사용성을 강조한 객체지향 개념이 등장하였으나 여러가지 한계를 가진 것으로 나타났다. CBD 는 객체지향방법론의 문제점을 보완하는 의미에서 태동된 것으로 볼 수 있다

2.2 CBD의 개념과 특성

2.2.1 CBD의 개념

CBD란 일반적으로 “재사용 가능한 소프트웨어

모듈을 제작하여 기계 부품과 같이 이를 조립하여, 보다 복잡한 소프트웨어를 만드는 방식"이다. 대다수의 개발 프로젝트에는 공통된 기능을 하는 모듈들이 있으며, 이의 재사용으로 부가가치를 획득할 수도 있기 때문이다[15]. 즉 CBD는 컴포넌트라는 소프트웨어 모듈의 재사용성과 독립성을 보장하여 소프트웨어의 복잡성과 생산성 문제를 해결하고자 하는 개발 패러다임이다.

CBD는 단순한 소프트웨어 개발이 아니라 개발을 중심으로 한 일련의 전략을 모두 포함하고 있다. 즉 요구사항 관련 지식과 컴포넌트 모델의 재사용에 대한 전략, 기존 시스템의 재활용 전략, 컴포넌트 유통시장을 통한 컴포넌트 구매 전략 등이 전체 개발 생명주기를 통해 강조되고 있다. 또한 개발 프로젝트의 아키텍처, 프로세스, 기술 등을 고려한 전략에 입각하여 컴포넌트를 사용하고, 구현하는 것이 중요하다[8].

컴포넌트 조립단계에서는 컴포넌트로부터 어떻게 어플리케이션을 설계하고 구현할 것인가에 대해서 고려해야 한다. 따라서 컴포넌트간 상호작용성을 파악하고 이를 구현하는 것이 CBD의 중요한 부분을 차지하게 된다[8].

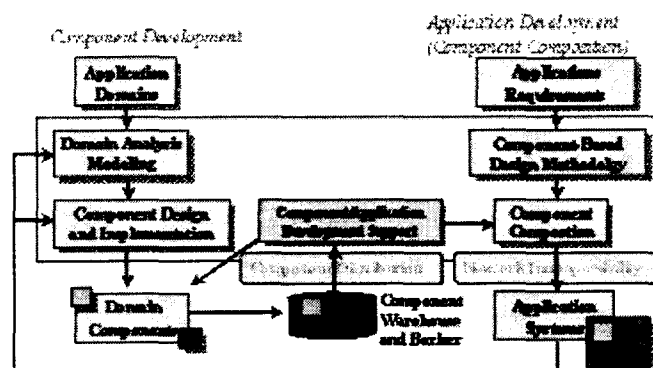
2.2.2 CBD 프로세스의 특징

CBD 프로세스는 모듈화, 부품화된 아키텍처 중심적이며, Black Box 형태의 컴포넌트를 단위로 한다. 또한 점진적, 반복적, 병행적인 프로세스를 가지며, 조립에 의한 개발을 지향한다[4]. CBD는 모듈 방식의 아키텍처를 따르기 때문에 시스템을 부분적으로 개발할 수도 있고, 컴포넌트의 추가 및 대체에 의해 기능을 점진적으로 확장할 수도 있다.

컴포넌트는 재사용 부품 소프트웨어로서 컴포넌트 기반의 재사용은 소스코드를 바탕으로 재사용하는 것이 아니라 실행코드를 기반으로 재사용하는 것이다. 이를 위해서는 컴포넌트가 명세화 되어 있어야 하고, 표준에 따라 개발

되어야 하며, 패키지화되어 있어야 한다. 개발방법은 그림 2 와 같이 컴포넌트 개발과 컴포넌트 조립을 포함한다. CBD 는 인터페이스를 통한 컴포넌트의 합성에 초점을 두기 때문에 여러 컴포넌트들의 상호 작용에 대한 설계를 필요로 한다.

컴포넌트 생성과 컴포넌트 조립이라는 프로세스의 분리에 따라 조직은 컴포넌트 벤더와 컴포넌트 사용자로 전문화된다. 이러한 전문화는 두 조직간의 중개자인 컴포넌트 유통업자를 필요로 하게 된다[4,8].



[그림 2] CBD 개발방법

2.3 CBD 방법론의 종류

산업계나 학계에서 많이 사용되고 있는 CBS 방법론의 종류에는 RUP, Catalysis, Select Perspective, 마르미-III 등이 있다. 본 연구에서 분석한 프로젝트들은 한국 전자통신연구원에서 개발한 마르미-III를 사용 하였다.

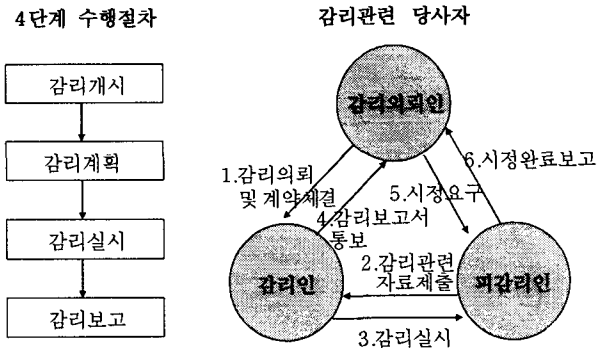
3. 정보시스템 감리

3.1 정보시스템 감리 개요

정보시스템감리라 함은 감리대상으로부터 독립된 감리인이 정보시스템의 효율성, 효과성 및 안전성 향상을 목적으로 정보시스템의 구축·운영에 있어서 합목적성, 적용기술의 적합성, 자원사용의 적정성을 점검, 평가하고, 감리의뢰인 및 피감리인에게 개선이 필요한 사항을 권고하는 것을 말한

다(정보시스템감리기준 제 2조) [5].

정보시스템 감리는 한국전산원에 의뢰하는 경우와 민간기업에 의뢰하는 경우가 있다. 감리관련 당사자는 감리의뢰인(발주자), 피감리인 (개발자, 발주자), 감리인으로 구성되며, 감리의뢰는 감리의뢰인과 감리인간의 계약체결과 동시에 시작된다.



[그림 3] 감리 수행절차

정보시스템 감리는 그 분야에 따라 기술감리, 비용감리 및 성과감리로 구분되며, 사업단계에 따라 사전감리, 진행감리, 사후감리로 구분된다. 한다. 감리대상에 따른 감리는 사업감리와 운영감리로 구분된다. 사업감리는 개발사업의 성공적 수행을 목적으로 하며, 개발사업의 진행 단계에 따라 실시하며, 운영감리는 컴퓨터시스템의 설비조직, 업무의 운영관리, 오류대책 등을 점검하는 감리로서 주기적 또는 특별한 사안 발생시에 실시한다[3].

본 연구에서 대상으로 하는 것은 사업감리로서 소프트웨어 개발 생명주기를 기준으로 분석, 설계, 구현, 시험 등 전 개발단계 감리에 따른 것을 중심으로 한 것이다[3].

개발 감리지침은 개발실시기준과 감리검토항목들로 구성되어 개발시스템 감리시 객관성 있는 기준에 따라 감리가 수행될 수 있도록 한다. 정보시스템 감리의 일관성 확보를 통해 감리 품질을 유지하기 위한 것이다.

[표 1] 감리 유형 구분

구분	내용	
기술감리	<ul style="list-style-type: none"> 기술측면에서의 정보기술 타당성 검토 정보기술자원의 품질 검토 정보기술자원의 내부통제 심사 	
비용감리	<ul style="list-style-type: none"> 정보시스템 사업의 계약내용과 비용간의 타당성, 실행의 적정성을 검토 실시시기에 따라 사전원가 계산과 정산으로 구분 	
성과감리	<ul style="list-style-type: none"> 기술감리 정보를 이용한 정보기술 관리측면에서의 평가 정보기술 활용의 최종 성과를 평가 정책결정을 위한 신뢰성 있는 정보제공 	
감리대상에 따른 구분	사업감리	<ul style="list-style-type: none"> 정보시스템 중장기계획 감리 응용시스템 분석/설계 감리 응용시스템 구현 감리 시스템 통합사업 감리
	운영감리	<ul style="list-style-type: none"> 시스템 통제준거성 감리 시스템 안전성 감리 시스템 효율성 감리 시스템 효과성 감리
사업단계에 따른 구분	사전감리	신규사업이 착수되기 이전에 사업 전반에 대한 검토
	진행감리	시스템 개발과정에서 진행의 적정성을 확인
	사후감리	종료된 사업에 대해 종합적인 적정성의 평가

개발감리의 범위는 개발공정 준비, 시스템 분석, 시스템구조 설계, 소프트웨어 분석, 소프트웨어 설계, 구현, 소프트웨어 통합시험, 시스템시험, 설치 및 인수까지의 개발 전 단계를 포함하고 있다[3].

3.2 정보시스템 감리지침

프로젝트 관리 감리지침(1997)은 프로젝트를 수행하는 절차에 따라서 프로젝트 착수, 프로젝트 계획, 수행과 통제, 검토와 평가, 완료, 프로젝트 지원 등으로 구분하여 정의한다[11].

정보시스템 분석 감리지침(1997)은 개발 실시기준에 대해 보다 세부적으로 감리검토항목을 기술하고 있다. 분석 감리지침은 시스템 분석활동과 소프트웨어 분석활동을 통합하여 다룬다. 따라서 분석기법을 통해 산출된 문서와 활동의

적정성 평가를 위한 감리 검토항목을 중심으로 기술되어 있다[9].

정보시스템 설계 감리지침(1997)은 설계활동을 위한 감리지침으로 시스템 구조 설계와 소프트웨어 설계활동을 통합하여 설계 감리지침으로 작성되어 있다[10].

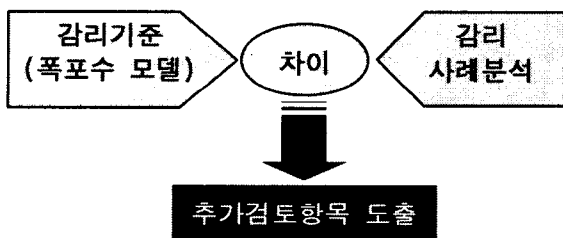
시스템 시험 감리지침(1997)은 ISO/IEC 12207 에 따른 시험관련 활동에 대한 감리지침을 제공하고 있다. 시스템 시험을 단위시험, 소프트웨어 통합시험, 소프트웨어 자격요건 시험, 시스템 통합시험, 시스템 자격요건 시험 등의 활동으로 구분하여 정의하고 있으며, 구체적인 감리 검토사항을 기술하고 있다[12].

4. 사례 연구 및 결과

4.1 연구 방법

사례연구 대상 프로젝트는 사례 A(공공 프로젝트), 사례 B(은행의 차세대 프로젝트)이다. 두 개의 프로젝트 모두가 CBD 방법론으로 프로젝트를 수행하였고 단일 프로젝트로는 비교적 대규모의 프로젝트이다.

감리사례 분석은 중간 및 최종 감리보고서를 중심으로 하였다. 감리보고서 분석은 각 프로젝트 별로 상세검토사항 수준에서 감리기준의 검토항목과 실제 사례에서 수행한 검토항목간을 비교 분석하였다.



[그림 4] 감리기준과 사례분석 틀

4.2 사례 프로젝트 개요

표 2 와 같이 사례 A 프로젝트는 시스템 사용자가 국민 일반이며, 사례 B 프로젝트는 기업의 내부용 차세대 정보시스템 구축 프로젝트로서 시스템 사용자가 한정되어 있었다. 사례 A 는 사업추진팀과 개발자가 프로젝트의 추진을 주도한 반면, 사례 B 는 시스템사용자, 사업추진팀, 개발자, 통제팀이 공동으로 추진한 프로젝트이다. 사례 B 는 매 단계 종료시점이 계획에 못 미칠 경우 진척율에서 차감하는 진척 불삽입 기준을 적용하여 진척관리를 하였으며, 통제팀을 별도의 조직으로 두어 개발자 및 사용자 간의 중재역할을 담당하였다.

[표 2] 사례 프로젝트 개요

구분	사 례 A	사 례 B
프로젝트 특성	대국민 대상 공공 프로젝트	금융기관 차세대 시스템
추진 조직	사업추진팀, 개발자	사업추진팀, 개발자, 통제팀, 사용자
개발방법론	PPC CBD 경로	마르미-III(ETRI)
프로젝트 통제	사업추진팀	사업추진팀, 통제팀
품질보증 활동	개발자 품질팀	통제팀, 개발자 품질팀
진척관리	개발주기에 따른 관리	별도의 관리기준
사업목표	<ul style="list-style-type: none"> 민원행정의 효율성 극대화 전자민원 전자화 웹서비스 기반의 시스템 구축 	<ul style="list-style-type: none"> 위험자산의 관리 및 적정 수익성유지 기간제 업무시스템개발 정보시스템 운영 및 관리 선진화

4.3 감리보고서 구성

감리보고서는 감리개요와 감리실시 결과로 구분되었다. 감리개요 부분의 감리계획서는 감리수행 근거, 감리목적, 감리기준, 감리대상, 감리중점사항, 감리기간, 감리절차, 감리계획서 및 보고서 통보기관, 행정사항, 감리인 편성 등으로 구성되어 있다. 사업개요는 사업명, 사업기간, 사업목표, 사업범위, 시스템 개발적용

방법론, 사업비, 투입 인력 등에 대하여 기술되어 있다.

감리실시 부분에서 총평은 감리보고서에 대한 평가의견을 기술한 부분으로 그 내용은 전제조건, 분야별 현황, 종합의견 등으로 구성되어 있다. 개선권고사항은 감리 지적 사항에 대한 개선 유형과 세부개선 사항으로 구분된다. 상세검토사항에는 각 검토항목에 대한 현황 및 문제점을 기술하고, 개선 권고사항에 대해 개선방향을 기술한다.

[표 3] 감리보고서 구성

구분		주요 내용
감리개요	감리계획서	감리수행에 대한 전반적인 사항 기술
	사업개요	감리대상 사업에 개괄적인 사항을 기술
감리실시	총 평	감리인의 검토의견을 종합하여 기술
	개선권고사항	지적 사항의 개선 유형을 기술
	상세검토사항	개선권고사항의 세부 검토 내용을 기술

5. 연구결과 및 시사점

5.1 감리보고서 분석

감리보고서 분석은 중간감리보고서 및 최종감리보고서의 상세검토사항을 중심으로 정보시스템감리 기본점검표(이하 감리기준이라 함)(정통부 고시 1999- 104 호)와 사례에서 실제로 검토한 내용을 비교하였다.

감리기준 중 개발공정인 분석단계, 설계단계, 구현단계, 시험단계를 중심으로 분석을 실시하였으며, 분석은 감리기준을 중심으로 작성한 점검표 (이하“체크리스트”라 함)와 감리보고서를 비교 분석하였다. 여기서 보고서에서는 검토를 하였으나 체크리스트에는 없는 검토항목을 도출하였다.

[표 4] 감리기준과 사례의 검토사항 비교

검리영역	검토사항	검리 기준	사례 A		사례 B	
			중간	최종	중간	최종
일반 관리	범위관리	범위관리의 적정성	✓	✓	✓	✓
		변경관리의 적정성	✓	✓	✓	✓
	일정관리	일정 및 진척관리의 적정성	✓	✓	✓	✓
	위험관리	위험의 식별 및 추적관리	✓	✓	✓	✓
	예산관리	예산관리의 적정성	✓	✓	✓	✓
	품질보증	품질보증 활동의 적정성	✓	✓	✓	✓
		프로젝트 표준설정 및 적용	✓	✓		✓
		정점 및 미결사항 관리	✓	✓	✓	✓
		개발방법론 선정 및 적용	✓	✓	✓	✓
		외주 인력관리의 적정성	✓			✓
품질 단계		프로젝트 단계 실행				✓
	응용시스템	사용자 요구사항 명시화	✓	✓		✓
		요구사항 추적성	✓	✓	✓	✓
		프로서스 모형적 적정성	✓	✓		✓
		유즈케이스 모형적합성		✓		✓
		개발모형적합성		✓		✓
		UML모형적합성		✓		✓
		제사용 검토판도작성				✓
	데이터베이스	데이터 모형적 적정성	✓	✓		✓
		프로세스 모델과의 일관성	✓	✓	✓	✓
시스템 구조	시스템 요구사항 명시화	✓	✓		✓	
	시스템구조의 설계 및 설계	✓	✓		✓	
	안전·신뢰성 분석 및 도출	✓	✓		✓	

검리영역	검토사항	검리 기준	사례 A		사례 B	
			중간	최종	중간	최종
설계 단계	응용시스템	업무기능분석에 요구사항 반영	✓	✓		✓
		사용자 인터페이스 설계 일관성	✓	✓		✓
		프로그램 설계 명시화	✓	✓		✓
		사업계획 수립의 적정성	✓	✓		✓
	사케 설계	설계 전략적합성		✓		✓
		핵스킨스킬모던 설계		✓		✓
		컴포넌트 특독		✓		✓
		업무 컴포넌트 핵스킨스킬모던 설계		✓		✓
	데이터베이스	데이터베이스 설계의 적정성	✓	✓	✓	✓
	시스템	응용시스템 설계와의 일관성	✓	✓	✓	✓
구조	운영 시스템구조 설계의 적정성	✓	✓		✓	
	안전·신뢰성 설계 반영	✓	✓		✓	
	개발기 사케모드 프로젝트별 계획평가		✓		✓	
	특평가 비나프로젝트 작업계획		✓		✓	
구현 단계	응용시스템	프로그램 구현 충분성	✓	✓		✓
		단위시험의 적정성	✓	✓		✓
		S/W 통합및시스템통합시험계획 수립적정성	✓		✓	
		지침서 작성 및 교육계획 수립의 적정성	✓		✓	
	데이터베이스	데이터베이스 구현 적정성	✓	✓		✓
	시스템 구조	운영 시스템구조 구현의 적정성	✓	✓		✓
		안전·신뢰성 구현의 적정성	✓	✓		✓
		클래스 구조설계			✓	✓
		컴포넌트 상호 작용설계			✓	✓
		드라이브 클래스 상세 설계			✓	✓
	유즈케이스 분석			✓	✓	
	비나프로젝트별 비나 프로젝트 평가				✓	
시험 및 평가	시험 및 평가	사용자/운영자 교육의 실시	✓	✓		✓
		소프트웨어 통합시험의 적정성	✓	✓		✓
		시스템 통합시험의 적정성	✓	✓		✓
		시스템 원리의 적정성	✓	✓		✓
		인수시험의 적정성	✓			✓
	완전후 관리 시스템 결과보고서			✓	✓	

분석 결과의 객관성 확보를 위하여 정성적 평가를 실시하였다. 평가기준은 체크리스트의 검토항목이 적정(70%이상)하게 검토된 것은 3 점, 보통(50%이상)은 2 점, 미흡(50%미만)한 것은 1 점으로 구분하여 평가 하였다.

평가 방법은 경험과 지식이 비슷한 두 사람(기술사, 교수, SE 전공, 감리경험 풍부)이 각각 별개의 장소에서 평가를 하고, 그 점수의 합계가 4 점 이상인 경우 검토항목으로 그 항목을 반영하였다. 전체 내용을 취합하여 표 4 를 작성했다. 또한 두 사람간의 신뢰도를 측정하기 위해 처음부터 58 개(29 개를 두 사람이 분석)의 항목에 대해 분석한 결과 전체를 58 점으로 할 때 차이항목이 7 점으로 나타났으며 이를 계산하면 신뢰도가 87.9%((58-7)/58)나타났다.

사례에서의 감리는 중간감리, 최종감리로 구분되어 실시되었다. 감리시점은 폭포수 개발 모형에 따라 분석 및 설계가 종료된 시점에서 중간감리가, 구현이 끝나고 통합시험 및 시스템 시험 시점에서 최종감리가 실시되었다.

일반관리의 감리영역은 범위관리, 일정관리, 위험관리, 형상관리, 품질보증, 프로젝트 표준 및 기타로 구분한다. 사례를 분석한 결과 대부분의 감리기준에 대해 검토를 한 것으로 나타났으나, CBD 의 특성인 반복 개발모델에 관련된 통제 활동이 사례(표 4 의 일반관리의 “프로젝트 통제 활동”)에서 발견되었다. 이 활동에서는 프로젝트가 진척됨에 따라 진척율에 대하여 통제하는 활동으로 매 단계에서 목표에 도달하지 못한 경우 진척율에 포함시키지 않는 것이 특징이었다.

분석단계의 감리영역은 응용시스템, 데이터베이스, 시스템 구조로 분류되며, 사례도 이에 따라 검토되었다. 한편 분석단계에 해당되는 유즈 케이스 모형작성, 개념 모형작성, UI 프로토타이핑, 재사용 컴포넌트 조사 등은 현재의 체크

리스트에 없는 항목이나 감리시에 검토한 것으로 나타났다. 재사용 컴포넌트의 조사는 사례에서 검토되지 않았다. 이는 기존의 컴포넌트가 없어 검토되지 않은 것이다.

설계단계의 감리영역은 분석단계와 동일하게 응용시스템, 데이터베이스 시스템구조로 구분되나, 세부 점검사항은 업무기능 설계, 사용자 인터페이스 설계, 프로그램 설계, 시험계획 수립 등으로 구분하여 검토되었다. 사례에서는 체크리스트에 없는 아키텍처를 정의, 비즈니스 컴포넌트 설계, 개발계획 평가 등을 감리영역으로 구분하고 이에 대한 검토가 이루어 졌다. 검토한 내용은 설계 전략정의, 비즈니스컴포넌트 설계(컴포넌트 명세서), 컴포넌트 인터페이스 정의서, 컴포넌트 순서도, 컴포넌트 협력도 등이었다. 이러한 부분에서 폭포수 개발방법과 CBD개발방법의 차이가 나고 있었다.

구현단계의 감리영역은 설계단계와 동일하게 응용시스템, 데이터베이스, 시스템 구조로 기술되어 있다. 체크리스트의 내용이 사례에서 모두 검토되었다. 그러나 사례에서는 CBD 의 검토항목으로 컴포넌트 설계, 미니프로젝트 점검 등이 추가로 검토된 것으로 나타났다. 컴포넌트 설계는 클래스 구조설계, 상호작용설계, 클래스 상세 설계, 유즈케이스 분석과 미니프로젝트 평가로 구성되었다.

시험단계의 검토사항은 사용자 및 운영자에 대한 교육, 종합적인 시험(통합, 시스템 시험 등), 시스템 설치의 적정성, 인수시험의 적정성 등에 대한 검토가 수행되었다. 그러나 사례에서는 시스템 설치 후에 시스템에 대한 관찰을 하고 그 결과를 보고한 시스템 관찰보고서에 대한 검토가 추가로 수행되었다.

5.2 분석결과 시사점

CBD 방법론의 경우는 폭포수 모델처럼 분석, 설계, 구현, 시험, 인도 등으로 구분하지 않고

착수단계, 아키텍처 단계, 점진적 개발단계, 인도단계로 분류되어 두 가지 방법에서 수행된 결과가 정확하게 대응되지 않았다.

본 연구의 결과를 통하여 다음과 같은 시사점을 얻을 수 있다. 우선 기존 감리기준에 따른 검토사항과 CBD 프로젝트에서의 검토사항이 상당 부분 차이가 있었다. 이는 폭포수 모델에서 수행하는 프로세스와 CBD의 프로세스에 차이가 있어 동일한 방식으로 중간감리, 최종감리를 실시하는 것은 감리품질의 문제를 야기시킬 수 있음을 의미한다.

또한 CBD는 폭포수 모델과 달리 데이터 모델링보다 프로세스 모델링을 중요시 한다는 논리와 일관되게 기존 감리기준에 명시되어 있지 않은 추가 검토항목을 발견할 수 있었다. CBD의 특성이 재사용을 강조하므로 개발되어 있는 컴포넌트들을 검색하고 이를 획득하는 활동들에 대한 점검이 필요하다.

이는 기존의 검토항목과는 상이한 항목들과 반복계획, 반복진행, 반복종료에 관련된 검토가 중요함을 의미한다. 먼저 반복계획에서의 검토사항은 태스크가 각 반복에 할당되는 내용, 반복의 범위, 반복에서 수행할 WBS, 각 반복의 기준선, 반복을 평가할 기준 등을 포함하여야 한다. 두 번째로는 반복진행 기간내 정해진 활동의 종료, 산출물의 일관성, 반복에서의 위험관리 등에 대한 검토가 필요하며, 반복종료 시점에 검토할 내용으로는 반복 점검실시, 진척도 검토, 미완료된 내용, 변경관리 등이 혼합되어야 한다. 이와 같이 아키텍처 단계와 점진적 개발단계가 유기적으로 반복되므로 매 단계에서 주요 마일스톤을 설정하고 확인하는 절차가 필요하다. 감리 검토항목에 대한 설명에 있어서도 구체적으로 점검내용을 제시 할 필요가 있다.

6. 결론 및 논의사항

정보시스템(Information System) 구축에 CBD 방법론을 적용하는 사례가 증가하고 있다. 컴포넌트 기반 개발(CBD)방법론의 경우 기존의 개발방법론과 개발과정이 달라 그에 상응하여 감리기준도 개선이 될 필요가 있다. 본 연구에서는 전통적 개발방법론을 근간으로한 감리기준 대비 CBD 적용 사례 프로젝트를 분석하여 감리기준의 개선점 여부를 탐색하였다. 정부 공공프로젝트와 금융기관의 차세대정보시스템 구축 프로젝트를 사례로 선정하여 비교분석한 결과 기존의 정보시스템감리 기본점검표와 사례의 감리검토항목 간에 차이점을 발견할 수 있었고, 그에 따른 개선점을 유추할 수 있었다.

본 연구 결과의 일반화를 위해서는 다음과 같은 후속 연구가 추가되어야 할 것이다.

- CBD 방법론으로 개발된 산출물의 품질보증 활동에 대한 연구
- 아키텍처 정의단계에 대한 감리시의 중점 검토 항목 도출과 이에 관련된 품질활동에 대한 점검 방안에 대한 연구

결론적으로 본 연구에서는 CBD 프로젝트에 적합하다고 판단되는 감리검토 개선항목을 탐색적으로 제시하였으며, 이는 CBD를 기반으로 한 소프트웨어 제품 개발의 효율성 제고 및 품질향상에 이바지 할 것으로 기대된다.

[참고문헌]

- [1] 나희동, "Architecture driven Component Development" : SoftEXPO 2001, 2001.11.
- [2] 박준성, "CBD 개론", 한국소프트웨어컴포넌트 컨소시엄, 2001.12
- [3] 문대원, 장시영, 정보시스템 감리, 명경사, 1998. 7.20
- [4] 오영배, 박준성, "CBD 적용사례 연구", 한국

정보과학회 소프트웨어공학회지, 제12권 제3호, 1999.9.

- [5] 정보통신부, 정보시스템 감리기준고시, 고시 제1999-104호, 1999.12.22
- [6] 최 성 외, "CBD 현황과 전망", 한국정보처리학회, 2003.5. 17쪽
- [7] 한국소프트웨어컴포넌트컨소시엄, 컴포넌트란 무엇인가? 알기쉬운 소프트웨어 컴포넌트, 2001.12
- [8] 한국소프트웨어컴포넌트컨소시엄, 소프트웨어 컴포넌트, 제 2호, 2002.10
- [9] 한국전산원, 정보시스템 분석 감리지침 연구, 1997. 12
- [10] 한국전산원, 정보시스템 설계 감리지침 연구, 1997. 12
- [11] 한국전산원, 정보시스템 프로젝트관리 감리지침 연구, 1997.12
- [12] 한국전산원, 정보시스템 객체지향 개발감리지침 연구, 1998.11
- [13] 한국전자통신연구소, 마르미-III 절차서, 2001.9.
- [14] Carliss, Y.B. & Kim, B.C., "Managing in an Age of Modularity," Harvard Business Review, September/October, 2001
- [15] Benko, C. & McFarlan, F.W., Connecting the dots - Aligning Project with Objectives in unpredictable times, Harvard Business School Press, 2003, PP 85-118