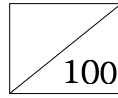


# 기능점수방식으로의 전환방안 및 지침

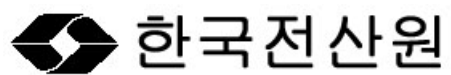
2002. 12.





# 기능점수방식으로의 전환방안 및 지침

2002. 12.



## 序 文

정보시스템이 조직업무의 필수적인 수단으로 정착됨에 따라 정보화에 투자되는 예산이 지속적으로 증가하고 있다. 정보화예산의 효과적 관리를 위해서는 정보화사업의 비용을 정확하게 예측할 수 있어야 한다. 정보화사업의 가장 중요한 부문은 소프트웨어개발이며 소프트웨어개발에 소요되는 비용예측의 정확성을 제고하기 위해 소프트웨어사업대가기준의 개정이 추진되고 있다.

기능점수방식은 사용자 관점에서 소프트웨어가 제공하는 기능의 양을 계량화하는 방법으로 기존 라인수 방식에 비해 소프트웨어의 규모를 보다 정확하게 측정할 수 있는 방법이다. 본 연구는 기능점수방식의 소프트웨어사업대가기준 도입에 따라 공공기관의 정보화사업 예산수립, 사업관리 절차가 어떻게 변화되어야 하는가를 분석하고 변화된 절차를 수행하기 위한 지침을 제시하는 데 그 목적이 있다. 또한, 소프트웨어개발사업의 수주자와 감리인이 사업관리나 정보시스템감리에 기능점수를 활용할 수 있는 방안도 함께 제시하고 있다.

예산수립시 기능점수방식을 활용하기 위해서는 사용자 요구사항에 대한 상세한 분석이 사전에 이루어져야 하며 이를 위해서는 정보전략계획수립 단계에서 구축할 소프트웨어의 기능내역이 상세하게 도출되어야 한다. 또한, 정보화사업 계약이 기능점수방식에 의해 이루어지면 사용자 요구사항의 추적관리와 범위관리 등이 보다 용이해지고 사업완료시 사업비 정산의 명확한 근거자료로도 활용될 수 있을 것이다.

끝으로, 본 연구결과를 활용함으로써 국내 정보화사업에 기능점수방식이 보급·정착될 수 있기를 기대한다.

2002년 12월

한 국 전 산 원

원 장 서 삼 영

## 제 출 문

정보통신부장관 귀하

본 연구보고서를 “기능점수방식에서의 전환방안 및 지침”  
의 최종 연구개발 결과보고서로 제출합니다.

2002년 12월

연구책임자 : 정보기술감리부장 이현옥

참여연구원 : 선임연구원 임재남 (지식정보기술단 정보기술감리부)

                  선임연구원 신수정 (지식정보기술단 정보기술감리부)

                  주임연구원 박찬규 (지식정보기술단 정보기술감리부)

                  주임연구원 김정미 (지식정보기술단 정보기술감리부)

                  연구원 성민정 (지식정보기술단 정보기술감리부)

# 요 약 문

## 1. 제목

“기능점수방식으로의 전환방안 및 지침”

## 2. 연구개발의 목적 및 필요성

소프트웨어개발사업의 규모를 정확히 측정하고 그에 따라 사업비를 정확하게 예측하고자 기능점수방식의 도입을 추진하고 있다. 그러나, 기능점수방식은 사업발주이전에 사용자의 기능요구사항을 상세히 분석하여 소프트웨어가 구현할 기능 및 저장자료 목록 등이 도출되어야 한다. 따라서, 기능점수방식의 도입으로 인한 혼란을 방지하기 위해서는 기존의 정보화사업 발주 절차나 발주이후의 사업관리 절차들이 적절히 변화되어야만 한다.

본 연구는 기능점수방식의 소프트웨어사업대가기준 도입에 따라 공공기관의 정보화사업 예산수립, 사업관리 절차가 어떻게 변화되어야 하는가를 분석하고 변화된 절차를 수행하기 위한 지침을 제시하는 데 그 목적이 있다. 또한, 소프트웨어개발사업의 수주자와 감리인이 사업관리나 정보시스템감리에 기능점수를 활용할 수 있는 방안도 함께 제시하고 있다.

## 3. 연구개발의 내용 및 범위

본 연구는 정보화사업의 추진단계를 계약이전과 계약이후로 크게 두 단계로 구분하여 기능점수방식의 도입에 따라 정보화사업 주

체가 수행해야 할 절차를 기술하였다.

정보화사업의 계약이전에 발주자가 수행하는 주요 업무로 예산 수립, 예산심의, 제안요청서 작성, 원가계산 등이 있다. 각 업무별로 기존에 수행되고 있는 절차를 분석하고, 기능점수방식의 도입에 따라 변화되어야 할 점을 제시하였다.

정보화사업의 계약이 체결되고 사업이 시작되면 요구사항의 추적관리, 범위관리, 일정 및 진척관리, 형상관리 및 단계말 승인 등에 기능점수방식을 적용할 수 있으므로, 각 분야별로 기능점수방식을 적용하는 방안을 제시한다.

마지막으로, 정보화사업의 각 단계말에 실시되는 정보시스템 감리시 기능점수방식을 활용할 수 있는 방안을 제시하였다.

#### 4. 연구결과

본 연구는 현행 공공기관의 정보화사업을 분석하여 정보화사업 추진절차를 예산수립 및 예산심의, 사업관리, 정보시스템 감리로 구분하여 공공기관의 발주자, 수주자, 정보시스템 감리인이 업무 수행시 참고할 수 있는 지침을 제시하였다. 본 연구의 결과를 개략적으로 정리하면 다음과 같다.

- (1) 공공기관의 정보화사업 추진절차
  - 정보화사업의 추진절차 분석
  - 기능점수방식의 도입에 영향을 받은 절차 도출
- (2) 예산수립시 기능점수방식 활용방안
  - 정보화예산수립시 문제점 분석
  - 예산수립시 기능점수 도출 방안
- (3) 정보화사업 관리시 점검사항
  - 소프트웨어 개발공정 분석

- 프로젝트 관리영역별 기능점수 활용방안

(4) 정보시스템 감리에서의 점검사항

- 정보시스템 개발감리 분석
- 정보시스템 감리시 기능점수 활용방안

## 5. 활용에 대한 건의

본 연구의 결과는 공공기관에서 기능점수방식을 사용하여 정보화사업의 예산을 수립하고 예산에 대한 심의를 수행하는 데 있어 유용한 지침으로 활용될 수 있을 것이다. 또한, 정보화사업을 관리하고 정보화사업의 단계별 산출물의 승인시에 점검해야 체크리스트로서 활용될 수 있으며, 정보시스템 감리시에 점검해야 할 항목을 도출하는 데에도 활용될 수 있을 것이다.

## 6. 기대효과

- 기능점수방식의 소프트웨어사업대가기준의 안정적인 정착
- 정보화사업의 적정비용 산정을 유도함으로써 소프트웨어의 품질제고에 기여
- 범위관리 및 일정/진척관리 등 정보화사업의 관리의 효율성 제고

## ■ 목 차 ■

제1장 서 론 .....	1
제2장 정보화 추진절차 .....	4
2.1 정보화추진절차 .....	4
2.2 기능점수 모형의 활용범위 .....	9
제3장 기능점수 산정지침(안) .....	11
3.1 소프트웨어 개발비 .....	11
3.1.1 개념적 절차 .....	1
3.1.2 구조적 절차 .....	2
3.1.3 실행 절차 및 산정 지침 .....	4
3.2 데이터베이스 구축 대가 산정 .....	5
3.2.1 범위 .....	6
3.2.2 사례분석 .....	5
3.2.3 대가산정의 절차 .....	8
3.3 컨설팅 사업비 .....	6
3.3.1 개념 및 구조적 절차 .....	6
3.3.2 실행 절차 .....	8
제4장 예산수립시의 기능점수 활용 .....	72
제5장 계약이후 발주기관의 점검 사항 .....	75
5.1 개발공정 단계 .....	75
5.1.1 착수 단계 .....	7
5.1.2 분석 및 설계 단계 .....	8
5.1.3 구현 및 설치 단계 .....	8
5.2 프로젝트 관리 영역 .....	8



5.2.1 범위관리 .....	8
5.2.2 일정관리 .....	9
5.2.3 형상관리 .....	9
5.3 기능점수 모형에 대한 점검사항 .....	9
5.3.1 비용 및 일정추정 관련 이슈 .....	9
5.3.2 이슈별 체크리스트 .....	9
제6장 정보시스템 감리에서의 점검사항 .....	103
6.1 정보시스템 감리의 영역 .....	103
6.1.1 정보시스템감리에의 영향 .....	103
6.1.2 정보시스템 개발감리에의 고려사항 .....	105
6.2 기능점수(Function Point)에 대한 감리 .....	106
6.2.1 기능점수 감리의 개요 .....	107
6.2.2 기능점수 결과에 대한 감사 절차 .....	108
제7장 참고문헌 .....	112

## ■ 표 목 차 ■

<표 1.1> 규모산정에서의 간이/상세 모형 비교 .....	3
<표 2.1> ISO12207 기본공정과 정보화사업관리단계 비교 .....	8
<표 2.2> 개발공정에서의 기능점수 모형 활용내역 .....	9
<표 3.1> 소프트웨어 개발비 산정 절차 .....	11
<표 3.2> 프로젝트 프로파일 작성 양식 .....	16
<표 3.3> 프로젝트 프로덕트 정의 양식 .....	17
<표 3.4> 프로젝트 프로세스 정의 양식 .....	18
<표 3.5> 단위프로세스명 또는 데이터기능유형 식별 양식 .....	21
<표 3.6> 기능유형별 가중치 식별 양식 .....	24
<표 3.7> 데이터기능유형의 복잡도 판정 기준표 및 가중치 .....	28
<표 3.8> 외부입력(EI)의 복잡도 판정 기준표 및 가중치 .....	30
<표 3.9> 외부출력(EO)의 복잡도 판정 기준표 및 가중치 .....	31
<표 3.10> 외부조회(EQ)의 복잡도 판정 기준표 및 가중치 .....	32
<표 3.11> 직접스텝수 산정 양식 .....	33
<표 3.12> 공정별 규모당 단가 .....	36
<표 3.13> 개발언어 유형별 보정계수 .....	38
<표 3.14> 개발공정중 개발언어 보정계수 적용대상 공정 .....	39
<표 3.15> 여러 종류의 개발언어를 사용하는 경우 .....	40
<표 3.16> 규모 보정계수 .....	41
<표 3.17> 어플리케이션 유형별 보정계수 .....	43
<표 3.18> 하드웨어 유형별 보정계수 .....	44
<표 3.19> 개발자 생산성 수준 파악 .....	45
<표 3.20> 기술등급별 소프트웨어노임단가표 (2002년기준) .....	45
<표 3.21> 사례1 (데이터 제작 양 비교) .....	46
<표 3.22> 사례2 (데이터 제작비 및 자료입력비 비교) .....	48
<표 3.23> 컨설팅 대가 산정 모형 .....	46

<표 3.24> 소프트웨어 개발 사업대가 산정 절차 .....	6
<표 3.25> 컨설팅 업무 가중치 결정표 .....	6
<표 3.26> 컨설턴트 등급별 자격요건 .....	6
<표 3.27> 컨설턴트 대가 .....	7
<표 3.28> 컨설턴트 대가 산정 예 .....	7
<표 4.1> 예산수립시의 애로사항 극복방안 .....	2
<표 5.1> 기능점수 관련 사업수행계획서 고려사항 .....	7
<표 5.2> 기능점수 관련 분석/설계 산출물 .....	83
<표 5.3> 범위관리 점검항목 .....	9
<표 5.4> 일정관리 점검항목 .....	9
<표 5.5> 형상관리 점검항목 .....	9
<표 5.6> 7가지 비용 및 일정 추진 관련 이슈 .....	9
<표 6.1> 감리분야에 따른 감리 유형 .....	13
<표 6.2> 감리대상과 사업단계에 따른 감리 유형 .....	14
<표 6.3> 감리 분야별 영향 .....	15
<표 6.4> 개발산출물의 추가 검토사항 .....	16

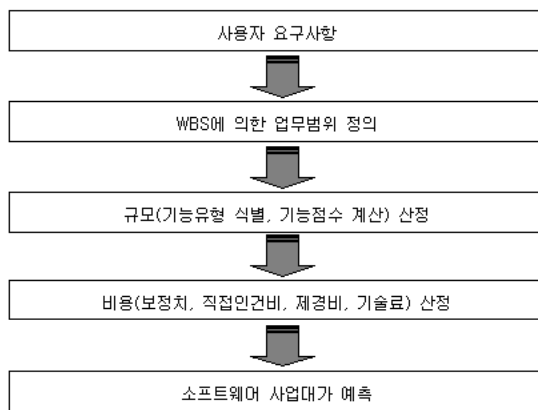
## ■ 그 림 목 차 ■

(그림 1.1) 대가산정 기본구조 .....	1
(그림 2.1) ISO 12207의 관리공정 .....	5
(그림 2.2) 정보화사업관리 단계간의 연계도 .....	7
(그림 3.1) 소프트웨어 개발비 산정 개념모델 .....	2
(그림 3.2) 소프트웨어 개발비 산정 구조 .....	3
(그림 3.3) 직원정보 파일의 RET와 ILF 측정 예 .....	2
(그림 3.4) 어플리케이션 규모 보정계수 함수 .....	2
(그림 3.5) 32개 모든 인자를 아는 경우의 생산성 함수 .....	4
(그림 3.6) 핵심 6개 인자만 아는 경우의 생산성 함수 .....	6
(그림 3.7) 현행 데이터베이스 구축 비용 산정 절차 .....	8
(그림 3.8) 데이터베이스 구축 비용산정 모델 .....	6
(그림 3.9) 컨설팅 대가 기준의 구조 .....	6
(그림 5.1) 소프트웨어 개발공정에서의 기능점수 산정 .....	6
(그림 5.2) 어플리케이션의 경계 .....	9
(그림 5.3) 개발인력 투입계획 .....	8
(그림 5.4) 패키지 커스터마이징 분야 .....	8
(그림 5.5) 변경요청 영향범위 파악 .....	9
(그림 5.6) 변경요청에 따른 일정계획 반영 .....	2
(그림 5.7) 변경영향 분석서(예) .....	9
(그림 6.1) 기능점수 요소별 기여도 분석 .....	10

## 제1장 서론

매년 수많은 정보화 개발 프로젝트가 추진된다. 그러나 현행의 정보화 과정에서 많은 문제점들이 발생하고 있다. 이 문제점들은 프로젝트의 규모 및 일정 예측에 대한 부정확성과 프로젝트 추진과정에서의 요구사항 가변성에 기인하고 있다. 이러한 문제를 최소화하고자 기능점수 방식을 도입하고 있다.

기능점수에 의한 대가산정의 기본구조는 (그림1.1)과 같다. 이 대가산정 방법은 사용자 요구사항에 따른 업무범위를 정의하여 규모와 비용을 산정하는 방식으로 발주자 또는 개발업체의 과거 DB에 기초하여 예측의 정확성을 향상시키며 세분화된 기능점수 목록에 근거하므로 요구사항의 변경관리가 기능점수의 범위내에서 통제될 수 있다.



(그림 1.1) 대가산정 기본구조

기능점수분석(Function Point Analysis)은 다음과 같은 목표를

가지고 있다.

- 성과 측정과 예측에 사용되는 소프트웨어 규모에 대한 일관성 있는 측정수단이 된다,
- 실 사용자가 이해하기 쉽다,
- 시스템에 사용된 기술과 독립적이다,
- 적용하기 쉽다,
- 요구사항 정의서로부터 결정될 수 있다.

현재의 프로그램 본수에 의한 개발규모 산출 방식보다 일관성 있는 수단으로 볼 수 있으나 사무처리용 시스템 위주로 패키지 개발, 웹기반 개발에 대한 한계도 있다.

기능점수 모형은 간이모형과 상세모형으로 나눌 수 있으며 사업계획 및 RFP 작성 및 제안 단계에서는 간이 모형이 적용되며 프로젝트 착수후 상세 분석단계 이후에는 상세모형을 통한 보다 정확한 개발규모를 산출해 낼 수 있다.

- 간이 모형 : ISP 또는 기초요구분석 후 예산 수립시
- 상세 모형 : 상세 요구분석 후, 기본설계 후, 계약 또는 정산시

계약이후 분석 및 설계, 구현, 테스트 단계에서는 상세 모형에 의한 기능점수 산출이 가능하므로 이에 근거한 개발규모 재 산정을 통하여 범위 변경통제와 공수 예측이 이루어질 수 있다

〈표 1.1〉 규모산정에서의 간이/상세 모형 비교

구분	간이 모형	상세 모형
규모산정	단위 업무기능 식별 소프트웨어 기능 식별 복잡도 산정 (생략)  규모 산정	단위 프로세스명 식별 S/W 기능 및 FP 유형 식별 DET, RET/FTR 파악 또는 항목수 * 처리복잡도 일반시스템 특성 파악 (생략) 규모 산정

출처 : 소프트웨어 사업대가기준 개선사업(2001.12), 정보통신부

본 지침(안)에서는 정보시스템 구축 프로젝트에 대한 개발범위와 방법에 따라 개략적인 규모를 예측하고 RFP를 작성한 후 다수의 사업자가 제시하는 제안서를 평가하여 적격자를 선택한 후 프로젝트 계약 및 사업관리를 할 때 기존의 본수 방식이 아닌 기능 점수 방식을 적용할 수 있도록 안내하는 지침을 제공한다.

이를 위해 먼저 정보화 추진절차를 소개한다. 제3장에서는 기능점수 산정지침을 제공한다. 이 지침에서는 기능점수에 의한 소프트웨어 개발비 산정, 데이터베이스 구축비 산정, 그리고 컨설팅 대가를 산정하는 지침을 제공한다. 제4장에서는 예산수립시의 기능점수 활용방안을 다루며, 제5장에서는 계약이후 발주자 기관의 입장에서 무엇이 달라지고 어떠한 부분을 중점 점검하여야 하는지 설명한다. 마지막으로 제6장에서는 정보시스템 감리 관점에서의 중점 점검사항을 다룬다.

## 제2장 정보화 추진절차

### 2.1 정보화추진절차

수주이후의 정보화 프로세스는 발주자 측면에서 크게 사업관리 프로세스와 개발공정 프로세스로 나누어 볼 수 있으며 정보화사업에 관련해서는 “공공부문 정보화사업관리 방법론 개발에 관한 연구(2002.4)”에서 제시된 체계와 절차를 활용하여 일반적인 공공기관에서의 사업관리와 개발공정을 구분하고 기능점수 모형의 도입에 따라 추가적으로 고려할 부분을 파악한다.

사업관리와 개발공정에 관련된 다양한 표준 및 연구가 진행중에 있으며 ISO/IEC 12207 등 국제표준을 비롯하여, PMI의 PMBOK(Project Management Body of Knowledge), 그리고 상용화된 관리 방법론이나 도구 등과 같은 관리방법론과 형상관리도구의 출현, CMM, SPICE 모델, COBIT-III 등 수행조직의 성숙도를 측정하기 위한 여러 가지 모델 등이 참조될 수 있다.

소프트웨어 프로세스에 대한 국제표준인 ISO/IEC 12207은 소프트웨어 개념화부터 폐기까지의 생명주기 상위수준 구조를 설정하고 있다. 소프트웨어를 포함한 시스템, 단독형(stand-alone) 소프트웨어 제품 및 서비스의 획득 동안에, 그리고 소프트웨어 제품의 공급, 개발, 운영 및 유지보수 동안에 적용될 수 있는 공정(process), 활동(activity) 및 세부 업무(task)를 포함한다.

ISO/IEC 12207는 기본공정, 지원공정, 조직공정으로 구성되어 있으며, 관리공정은 조직공정에 속해 있다. 실제로 획득공정에서 소프트웨어 생명주기가 시작되며, 공급공정에서 이를 응하여 개발, 운영 및 유지보수공정을 수행하게 된다. 관리공정을 포함한 조직공정은 조직이 관련 생명주기공정과 인력으로 구성된 하부



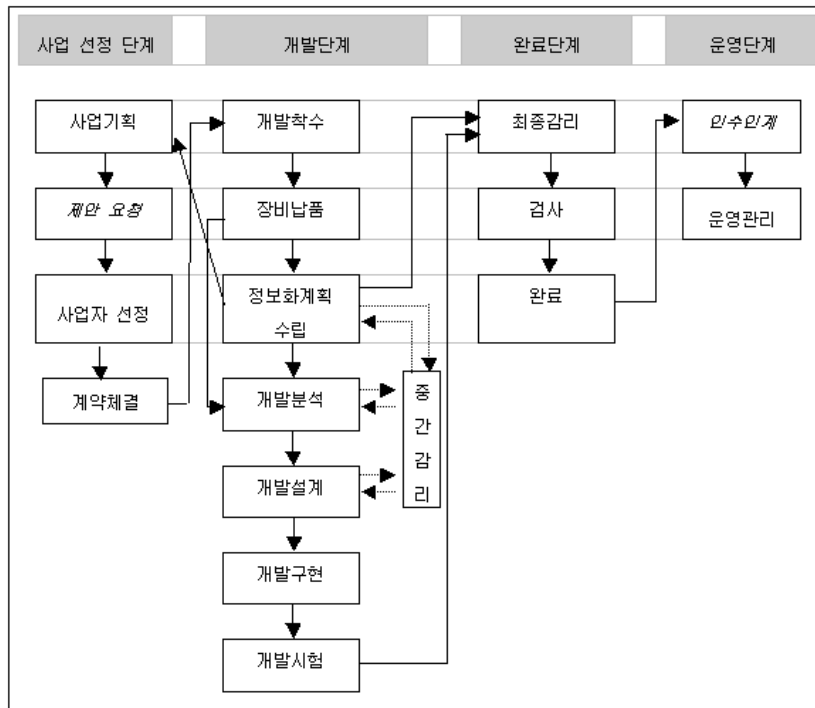
구조를 설정하고 구축하며 지속적으로 이 구조와 공정을 개선하는데 사용된다. 관리공정은 생명주기 공정이 수행되는 동안 프로젝트관리를 포함한 관리의 기본적인 공정을 정의하고 있다. ISO/IEC 관리공정은 계획, 수행, 통제, 검토, 평가 및 완료의 관리주기를 중심으로 작성되었으며, 주기별로 수행한 활동을 정의하고 있다. 그러나 ISO/IEC 12207은 정량적 관리기준을 제공하고 있지는 않으며, 일반적으로 사업관리보다는 개발공정에 초점을 둔 표준으로 참조할 수 있을 것이다.

#### (그림 2.1) ISO 12207의 관리공정

프로젝트 관리 분야는 1969년 설립된 국제적인 단체인 PMI(Project Management Institution)에서 정보통신 분야를 고려하여 1996년에 현재와 같은 PMBOK(A Guide of Project Management Body of Knowledge)로 발표하였으며 프로젝트 관리의 핵심지식체계를 프로젝트 통합, 범위, 일정, 비용, 품질, 인적자원, 의사소통, 위험, 조달관리 등의 9개 지식영역과 37개 프로세스로 제시하였다. 관리 프로세스 능력의 개선을 위해서는

SPICE, CMM 등의 성숙도 모형을 참조하여 활용할 수 있다.

한국전산원의 정보화사업관리방법론에 제시된 정보화사업의 세부적인 단계는 생명주기별로 4개의 관리 영역에 17개의 단계로 구성되어 있으며 이중 “과제선정”과 “소유권 양도”는 정보화 촉진기금에 의한 정보화지원사업에만 적용되는 단계이므로 일반적인 공공기관에서 자체 정보화사업을 추진할 경우에는 불필요한 단계이다. 또한 “정보화계획수립”단계는 정보화지원사업의 일부로 편성되어 있으나 사업기획 단계 이전에 수행하는 것이 일반적이며 이를 기초로 제안요청 및 사업자 선정단계를 추진하는 것이 타당하다.



(그림 2.2) 정보화사업관리 단계간의 연계도

개발공정에 대해서는 가장 보편적인 ISO/12207의 체계에 비교하여 분석, 설계, 구현, 시험 등 집합적으로 구분하고 있으며 정보시스템 감리 활동을 추가적으로 식별하여 구성하였다. 현재 공공부문에서 대부분 정보시스템 감리가 실시되고 있는 현실을 고려하여 감리 측면에서 기능점수 모형 도입에 따른 추가적인 고려사항을 파악한다.

〈표 2.1〉 ISO12207 기본공정과 정보화사업관리단계 비교

공정	활동	정보화사업관리상 수행단계
개발공정	공정구현	(7. 정보화기획 단계 )
	시스템 요구분석	8. 개발분석 단계
	시스템 구조설계	9. 개발설계 단계
	소프트웨어요구분석	8. 개발분석 단계
	소프트웨어구조설계	9. 개발설계 단계
	소프트웨어상세설계	
	소프트웨어 코딩 및 시험	11. 개발구현 단계
	소프트웨어 통합	
	소프트웨어자격시험	
	시스템 통합	12. 개발시험 단계
	시스템 자격시험	
	소프트웨어 설치	
	소프트웨어 수락지원	13. 최종감리 단계 14. 검사 단계 15. 완료 단계
운영공정	공정구현	17. 운영 및 관리
	운영시험	
	시스템 운영	
	사용자 지원	
유지보수 공정	공정구현 문제 및 수정 분석 수정 구현 유지보수 검토/수락 전환 소프트웨어 폐기	계약상의 하자보수와 운영관리에 포함되어 별도의 단계로 구분하지 않음.

## 2.2 기능점수 모형의 활용범위

수주이후 개발단계에서 기능점수 방식에 의한 규모 산정 및 공수 예측에 따라 사업관리 및 개발공정에 있어 보다 체계적인 접근이 가능하여 발주자 측면에서 개발과정에 보다 적극적인 참여가 필요로 하며 프로젝트 목표 일정과 범위에 대한 실현가능성을 높여 줄 것으로 기대된다.

사업관리 측면에서는 범위관리, 일정관리, 형상관리에서 기능점수 목록을 활용하여 프로젝트 편성 및 변경통제를 강화할 수 있으며 개발공정에서는 기존 산출물의 추가 보완을 통하여 기능점수를 상세 모형으로 유지관리할 수 있다. 또한 시험단계와 검수단계에서는 기능점수 목록이 기준선(Baseline)으로서 역할을 하며 프로젝트의 위험을 감소시키는 역할을 할 수 있을 것으로 기대된다. 가장 대표적으로 기능점수 모형이 활용되는 것은 요구사항 관리(Requirement Management)가 크게 향상될 수 있을 것이다.

<표 2.2> 개발공정에서의 기능점수 모형 활용내역

구분	세부 내역	활용내역	비고
프로젝트 관리	범위관리	과업의 범위 및 요구사항의 구체화 기능점수 목록에 근거한 요구사항의 변경통제 수행	요구사항추적표 업무배경도 (Context Diagram)
	일정관리	기능점수 모형에 기초한 투입공수 산출 요구사항 변경에 따른 공수 및 일정 의 영향도 파악	상세공정별 생산성자료
	형상관리	형상항목 및 형상통제 베이스라인 설정 시 단계말 기능점수 목록 활용	
개	분석단계	요구사항 정의시 기능점수 목록과 연계	프로세스정의서

발 공 정			엔티티정의서
	설계단계	내외부 시스템간 인터페이스 파악으로 EIF 도출 프로그램 상세설계시 프로그램과 테이블 관계 추적에 의한 복잡도 판단	프로그램 사양서 테이블정의서 인터페이스정의 서 프로그램 / 테이블 매트릭스
	구현단계	기능시험을 위한 시험요건의 도출과 시험 완료 기준 활용	
	설치단계	기능점수 목록에 대비한 과업범위 이행 여부 확인	

## 제3장 기능점수 산정지침(안)

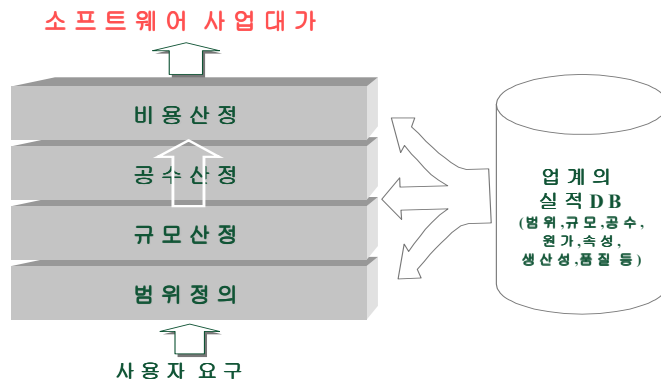
본 장에서는 기능점수에 의한 소프트웨어사업 대가 산정 지침(안)을 제시한다. 이 지침(안)은 소프트웨어 개발비, DB 구축대가, 그리고 컨설팅 사업대가로 이루어진다.

### 3.1 소프트웨어 개발비

#### 3.1.1 개념적 절차

소프트웨어 개발의 개념적인 절차는 아래의 (그림 3.1)에서와 같이 범위정의, 규모산정, 공수산정, 그리고 비용산정으로 구성된다.

범위정의에서는 사용자의 요구를 업계의 실적 DB를 참조하여 분할 가능한 서브시스템과 ISO 12207 (소프트웨어 라이프사이클 모델)의 표준 공정으로 구성되는 프로젝트 작업분류체계(WBS)를 만든다. 규모산정에서는 기능점수(Function Point: : FP)법을 이용하여(여의치 못하면 직접스텝수 산정), 정의된 범위내의 소프트웨어 규모를 산정한다. 대가기준에는 포함되지 않은 공수산정에서는 개발자의 능력을 조사하여 생산성을 파악한 다음에 소요될 공수를 판단한다. 비용산정에서는 발주자는 산정된 규모에 규모당 단가(FP 단가 또는 스텝 단가)를 적용하여 필요 예산을 판단하고, 수주자는 산정된 공수에 자신의 인건비 단가를 곱하여 원가를 산정한다. 단, 수주자가 자신의 원가를 산정하는 방법에 대해서는 별도로 제시하지 않는다.

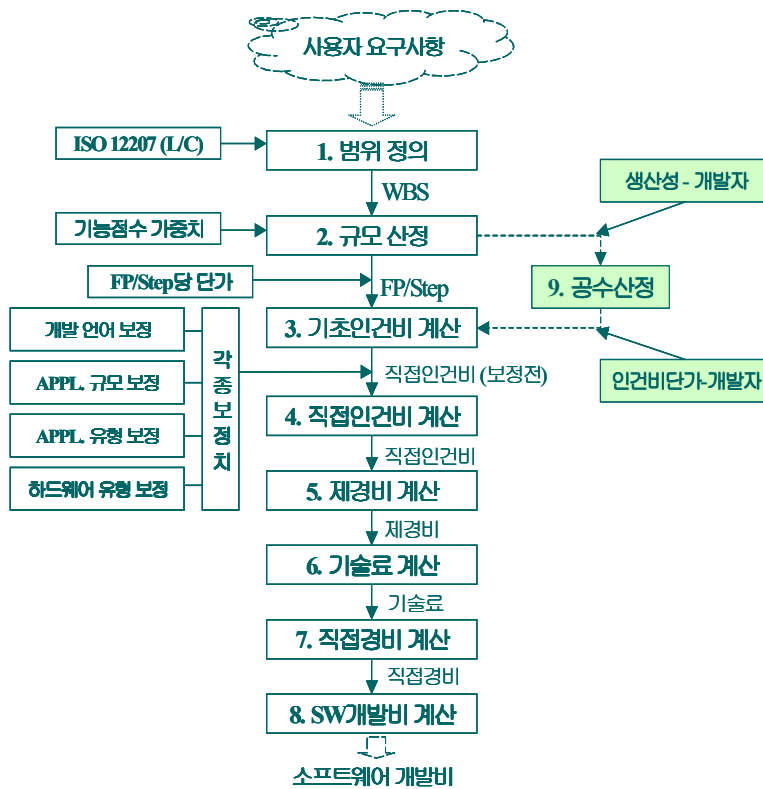


(그림 3.1) 소프트웨어 개발비 산정 개념모델

### 3.1.2 구조적 절차

소프트웨어 개발비 산정절차를 구조적으로 구분하면 (그림 3.2) 와 같이 8단계로 구분할 수 있다.





(그림 3.2) 소프트웨어 개발비 산정 구조

- 1 단계 - 범위정의 - 요구사항에 근거하여 정의
- 2 단계 - 규모산정 - 원칙은 FP로 하나 부득이한 경우는 스텝수로도 산정 가능
- 3 단계 - 기초인건비 산정 =  $\text{규모}(\text{FP}/\text{스텝수}) \times \text{단가}(\text{FP}/\text{스텝수})$
- 4 단계 - 직접인건비 산정 =  $\text{기초인건비} \times \text{보정치 (4종류)}$
- 5 단계 - 제경비 계산 =  $\text{직접인건비의 } 110\%$
- 6 단계 - 기술료 계산 =  $\text{직접인건비와 제경비의 합의 } 20\%$

**7 단계** - 직접경비 계산 - 직접경비 항목의 합

**8 단계** - 소프트웨어 개발비 계산 = 직접인건비 + 제경비 + 기술료 + 직접경비

### 3.1.3 실행 절차 및 산정 지침

앞 절에서 정의한 8단계의 산정 절차와 각 절차별 활동 및 적용 템플리트는 다음 <표 3.1>과 같다. 산정자의 필요에 따라 선택적으로 할 수 있는 공수산정도 포함하였다.

<표 3.1> 소프트웨어 개발비 산정 절차

단계	절차	템플리트
<b>1. 범위정의</b>	① 프로젝트 프로파일 작성	[1. 프로파일]
	② 프로덕트 정의	[2. 범위정의]
	③ 프로세스 정의 (ISO12207)	
<b>2. 규모산정</b>	① 단위 프로세스명 및 FP 유형 식별	[3. 규모산정]
	② 가중치 결정 - RET/FTR 파악 - DET 파악	
	③ 직접스텝수 산정	
<b>3. 기초인건비 산정</b>	① 규모당 단가 적용	[4. 기초인건비 산정]

<b>4. 직접인건비 산정</b>	각종 보정계수 적용 ① 개발언어 보정계수 ② APPL. 규모 보정계수 ③ APPL. 유형 보정계수 ④ 하드웨어 유형 보정계수	[5. 보정계수 적용]
<b>5. 제경비 계산</b>	① 직접인건비의 110%	[6. 비용산정]
<b>6. 기술료 계산</b>	① 직접인건비+제경비의 20~30%	
<b>7. 직접경비 계산</b>	① 직접경비 항목 파악	
<b>8. SW개발비 계산</b>	① 직접인건비+제경비+기술료 +직접경비	
<b>9. 공수산정</b>	① 개발자 생산성 수준 파악	[7. 생산성]
	② 기초소요공수 산정	[8. 공수산정]
	③ 직접소요공수 산정	
	④ 원가산정	[9. 원가산정]

#### 가. 제 1 단계 - 범위 정의

범위정의는 다음과 같이 작성된다.

- 범위정의는 사용자 요구사항을 근거로 작성한다.
- PMBOK<sup>1)</sup>의 WBS 작성법에 따른다.
- 우선 제품관점의 서브 시스템 및 주요 업무 기능 등의 컴포넌트를 구분한다.
- 구분된 컴포넌트를 구현하는데 필요한 공정을 식별한다.
- 필요한 공정은 ISO 12207의 엔지니어링 공정표준을 기준으로 식별하고, 추가로 필요한 공정을 식별한다.
- 제품의 특성과 프로젝트의 유형을 고려하여 수행공정을 식별한다.

1) PMBOK : Project Management Body of Knowledge

### ① 프로젝트 프로파일 작성

프로젝트 프로파일 작성의 목적은 산정내역을 이해하는데 필요한 프로젝트 개요 파악하고 산정 데이터베이스 구축에 필요한 구분 정보 파악하는 데 있다.

<표 3.2> 프로젝트 프로파일 작성 양식

가. 프로젝트 명 :					
나. 도메인 :					
	① 공공		④ 유통/서비스		⑦ 유틸리티
	② 금융		⑤ 국방		⑧ 교통
	③ 제조		⑥ 정보통신		⑨ 기타
다. 프로젝트 기간 :		yyyy-mm-dd	부터	yyyy-mm-dd	까지 ##### 개월
라. 고객 명 :					
마. 사업자 명 :					
바. 산정자					
	소 속 :				
	직 급 :				
	이 름 :				
바. 산정일자 :		yyyy-mm-dd			

적용할 때에는 도메인에 산정 대상 프로젝트의 해당 도메인을 아래 9가지 중 하나를 선택하고 기타일 경우는 ⑨를 선택하여 그 옆에 구체적으로 기술한다. 프로젝트 기간의 경우 계약이 아닌 실제기간을 기재하고 고객명은 발주자를 의미한다.

### ② 프로덕트 정의

프로덕트의 목적은 프로젝트 작업분류체계(WBS) 정의의 일환으로 여기서는 제품의 관점이고 시스템의 규모가 크면 관리가

어렵기 때문에 관리 가능한 서브시스템으로 구분하여 규모를 파악하기 위함이다.

<표 3.3> 프로젝트 프로덕트 정의 양식

① 어플리케이션 명	② 프로젝트 유형	③ 추가적인 요구사항													④ 요구사항		
		531 공정 유형	532 시스템 구분	533 시스템 구분	534 SW 요구 분석	535 SW 구조 설계	536 SW 상세 설계	537 SW 코딩 시험	538 SW 통합	539 SW 자격 시험	5310 시스템 통합	5311 시스템 자격 시험	5312 SW 설치	5313 SW 수락 지원	프로 토타 이핑	자료 변환	기타
인사	I	X	N/A	N/A	X	X	X	X	X	X	X	X	X	X	N/A	N/A	N/A
재무	R	X	N/A	X	N/A	X	X	X	X	X	X	X	X	X	N/A	5%	N/A
영업	K	X	X	X	X	N/A	N/A	N/A	X	X	X	X	X	X	N/A	3%	N/A
홈페이지	N	X	X	X	X	X	X	X	X	X	X	X	X	X	5%	N/A	N/A

주) ISO/IEC 12207의 지원활동(문서화, 현상관리, 품질보증, 검증/확인, 합동검토, 감사 등)과 관리활동(프로젝트 관리 등)

<표3.3>에서 타원으로 표시한 부분의 적용은 본 프로젝트에서 개발할 서브시스템(어플리케이션) 명을 기재하며 특히 담당조직이나 사람이 다른 경우를 고려하여 구분하여 작성한다. 프로젝트 유형은 다음의 5가지 중 하나를 선택하여 ①에서 구분한 어플리케이션별로 식별한다.

- 신규개발(N): 새롭게 시스템을 개발할 경우로 ISO 12207상의 13개 개발공정을 대부분 수행함
- 기능개선(I): 기존 시스템의 일부 소프트웨어 기능을 추가, 수정, 삭제하는 경우로 유지보수와 유사함
- 재구축/재개발(R): 기존 시스템의 상당 부분을 새로 구축/개발하여 시스템의 구조를 변경하는 경우
- 플랫폼교체(P) - 기존 시스템의 업무기능은 그대로 두고 기존 시스템의 플랫폼만 바꾸는 경우

- 패키지적용(K) - 패키지를 도입하여 시스템 구축에 필요한 부분만 변경 또는 개발하여 적용하는 경우

### ③ 프로세스 정의 (ISO 12207에 준함)

프로세스 정의의 목적은 프로젝트 WBS 정의의 일환으로 여기서는 개발활동의 관점이며 프로젝트 수행기간이 길면 관리가 어렵기 때문에 주요 산출물과 적용기술을 중심으로 개발활동을 여러 공정으로 구분하여 수행하는 점을 고려하여, 여기서는 ISO/IEC 12207의 표준 개발공정을 준용한다.

<표 3.4> 프로젝트 프로세스 정의 양식

① 어플리케이션 명	② 프로젝트 유형	531	532	533	534	535	536	537	538	539	5310	5311	5312	5313	④ 추가적인 요구사항		
		공정구현	시스템분석	시스템구분	SW요구분석	SW구조설계	SW상세설계	SW코딩및시험	SW통합	SW자격시험	시스템통합	시스템자격시험	SW설치	SW수락지원	추가적인 요구사항	자료변환	기타
		ENG 1	ENG 1.1	ENG 1.2	ENG 1.3	ENG 1.4	ENG 1.5	ENG 1.6	ENG 1.7	ENG 1.8	ENG 1.9	ENG 1.10	ENG 1.11	ENG 1.12	CUS 2	프로토타입	자료변환
인사	I	X	N/A	N/A	X	X	X	X	X	X	X	X	X	X	N/A	N/A	N/A
재무	R	X	N/A	X	N/A	X	X	X	X	X	X	X	X	X	N/A	5%	N/A
영업	K	X	X	X	X	N/A	N/A	N/A	X	X	X	X	X	X	N/A	3%	N/A
홈페이지	N	X	X	X	X	X	X	X	X	X	X	X	X	X	5%	N/A	N/A

주) ISO/IEC 12207의 지원활동(문서화, 현상관리, 품질보증 검증/확인, 합동검토, 감사 등)과 관리활동(프로젝트 관리 등).

위 <표 3.4>에서 타원으로 표시한 부분에 적용은 어플리케이션 별로 프로젝트 유형별 특성을 감안하여 수행하는 공정을 ISO/IEC 12207의 표준공정 중에서 식별한다. (식별이 곤란하면 SPICE ENG/CUS 공정 또는 각 셀의 메모 참조)

- X: 해당공정이 필요한 경우

- N/A: 해당공정이 불필요한 경우

추가적인 요구활동은 고객이 필요하다고 요구할 경우만 식별 하되 어플리케이션별로 구분할 수 있도록 한다. 하지만 비율은 기본공정 전체 투입에 대한 비율로 표시한다. 프로토타이핑은 요구수렴을 위해 핵심기능의 로직을 구현한 경우로 해당 어플리케이션의 전체 규모중 몇 %를 구현했는지를 명시하며 자료변환은 컨버전으로 현행시스템 또는 수작업에서 신시스템으로 변환해야 할 데이터가 있는 경우에만 해당된다. 기타 운영지원 등과 같이 표준공정으로 식별하기 어려우면서 별도의 노력을 인정해 주는 활동을 의미한다.

## 나. 2 단계 - 규모 산정

규모정의는 다음과 같이 작성된다.

- 규모정의는 기능점수(FP)를 원칙으로 하나 스텝수(LOC)가 합리적 이라고 판단되는 경우 스텝수로 산정해도 무방하다.
- 규모산정은 반드시 사용자 요구기능을 근거로 산정한다.(기술적, 품질적 특성은 무시)
- 기능점수는 IFPUG의 조정전 기능점수법을 사용함을 원칙으로 한다.
- '범위정의'에서 구분한 컴포넌트 단위로 산정함을 원칙으로 한다.
- 가정사항은 반드시 문서화하도록 한다.

### ① 단위 프로세스명 및 FP 유형 식별

단위 프로세스명 및 FP 유형 식별의 목적은 사용자 요구를 지원하기 위한 소프트웨어의 기능점수 유형과 수량을 파악하기 위

하여 서브시스템별로 세부 업무기능을 최대한 세분하여 단위 프로세스로 식별하고자 하며 식별된 단위 프로세스의 기능점수 유형을 식별해야 한다.

적용은 <표 3.5>에 녹색으로 표시한 부분이다.

단위프로세스(Elementary Process) 식별기준은 다음의 기준을 모두 충족해야 단위프로세스로 식별될 수 있다:

- 식별대상 프로세스가 사용자에게 의미 있는 가장 작은 단위의 액티비티임
- 식별대상 프로세스가 자체로서 완비되어 어플리케이션 비즈니스를 일관된 상태에 있음

데이터기능 유형(Data Function Type) 식별기준은 다음의 기준을 모두 충족해야 내부논리파일(ILF: Internal Logical File)로 식별될 수 있다.

- 데이터 또는 제어정보 그룹이 논리적이고 사용자가 식별할 수 있음
- 데이터 그룹이 특정 어플리케이션 경계내의 단위프로세스를 통하여 유지 관리됨

다음의 기준을 모두 충족해야 외부연계파일(EIF: External Interface File)로 식별될 수 있다:

- 데이터 또는 제어정보 그룹이 논리적이고 사용자가 식별할 수 있음
- 참조되는 데이터 그룹이 특정 어플리케이션 경계 밖에 있음
- 참조되는 데이터 그룹은 특정 어플리케이션 내에서는 유지관리 되지 않음
- 참조되는 데이터 그룹은 다른 어플리케이션에서 내부논리파일(ILF)로 유지 관리됨



트랜잭션기능 유형(Transaction Function Type) 식별기준은 다음의 기준을 모두 충족해야 외부입력(EI: External Input)으로 식별될 수 있다:

- 데이터 또는 제어정보가 어플리케이션 경계 밖에서 들어옴
- 경계 밖에서 들어온 정보가 데이터이면 최소한 하나 이상의 내부논리파일을 유지관리 해야 하고, 제어정보일 경우는 시스템의 행위를 변경해야 함
- 식별된 단위프로세스는 다음 3가지 중 하나에 해당되어야 함

프로세싱 로직이 어플리케이션 내 다른 외부입력(EI)의 그것과는 구분되게 유일해야 하며, 식별된 데이터 요소 집합이 어플리케이션 내 다른 외부입력의 그것과는 구분되어야 한다. 참조되는 내부논리파일 또는 외부연계파일이 어플리케이션 내 다른 외부입력의 그것과는 구분되어야 한다. 다음의 기준을 모두 충족해야 외부출력(EO: External Output)으로 식별될 수 있다:

- 데이터 또는 제어정보를 어플리케이션 경계 밖으로 내보냄
- 식별된 단위프로세스는 다음 3가지 중 하나에 해당되어야 함

프로세싱 로직이 어플리케이션 내 다른 외부출력(EO)의 그것과는 구분되게 유일해야 하며 식별된 데이터 요소 집합이 어플리케이션 내 다른 외부출력의 그것과는 구분되어야 한다. 또한 참조되는 내부논리파일 또는 외부연계파일이 어플리케이션 내 다른 외부출력의 그것과는 구분되어야 한다. 위 조건을 충족했다라도 다음 4가지 기준 중 하나는 충족해야 외부출력으로 식별될 수 있다:

- 단위프로세스의 프로세싱 로직은 하나 이상의 수학기식이나 계산식

을 포함해야 함

- 단위프로세스의 프로세싱 로직은 파생 데이터를 생성해야 함
- 단위프로세스의 프로세싱 로직은 최소한 하나 이상의 내부논리파일을 유지관리해야 함
- 단위프로세스의 프로세싱 로직은 시스템의 행위를 변경해야 함

다음의 기준을 모두 충족해야 외부조회(EQ: External Inquiry)로 식별될 수 있다:

- 데이터 또는 제어정보를 어플리케이션 경계 밖으로 내보냄
- 식별된 단위프로세스는 다음 3가지 중 하나에 해당되어야 함

프로세싱 로직이 어플리케이션 내 다른 외부출력(EO)의 그것과는 구분되게 유일해야 하며, 식별된 데이터 요소 집합이 어플리케이션 내 다른 외부출력의 그것과는 구분되어야 한다. 또한 참조되는 내부논리파일 또는 외부연계파일이 어플리케이션 내 다른 외부출력의 그것과는 구분되어야 한다. 위의 조건을 모두 충족했다더라도 다음 5가지 기준 모두를 충족해야 외부조회로 식별될 수 있다:

- 단위프로세스의 프로세싱 로직은 내부논리파일이나 외부연계파일로부터 데이터나 제어정보를 가져와야 함
- 단위프로세스의 프로세싱 로직은 수학적식이나 계산식을 포함해서는 안됨
- 단위프로세스의 프로세싱 로직은 파생 데이터를 생성해서는 안됨
- 단위프로세스의 프로세싱 로직은 내부논리파일을 유지관리해서는 안됨
- 단위프로세스의 프로세싱 로직은 시스템의 행위를 변경해서는 안됨

<표 3.5> 단위프로세스명 또는 데이터기능유형 식별 양식

어플리케이션 명	단위 프로세스명 또는 데이터기능 유형	FP유형	RET/FTR	DET	복잡도	가중치
인사	직원정보관리	EI	2	12	A	4
인사	직원정보관리	EI	2	12	A	4
인사	직원정보관리	EI	2	12	A	4
인사	직원정보관리	EO	2	10	A	5
인사	직원정보관리	EQ	1	5	S	3
인사	보직관리	EI	3	7	H	6
인사	보직관리	EO	3	6	H	7
인사	보직관리	EQ	3	4	A	4
인사	보직관리	EQ	1	4	S	3
인사	근무지조회	EQ	2	5	S	3
인사	근무지출력	EO	2	7	A	4
인사	직원면담기록	EI	2	6	A	4
인사	직원면담기록	EI	2	6	A	4
인사	직원면담기록	EI	2	6	A	4
인사	직원면담기록	EQ	2	6	A	4
인사	직원기록대장관리	EI	1	5	S	3
인사	직원기록대장관리	EO	2	6	S	4
인사	부양가족 변동자료작성	EQ	1	8	S	3
인사	고과대상자 자동통보	EO	1	4	S	4
인사	매주말 직원변동상황 자동 리포트	EO	1	8	S	4
인사	수표발행금액과 내역	EO	2	10	A	5

	수표발행 파일 기록					
인사	타시스템 전달 트랜잭션 파일을 작성	EO	1	6	S	4
인사	화면필드 Help메시지	EQ	1	3	S	3
인사	화면 Help메시지	EQ	1	3	S	3
인사	직원자료변환(구->신)	EI	2	15	A	4
인사	진급대상자 발체	EI	2	4	S	3
인사	정현원 통계	EO	2	4	S	4
인사	부서별 직원통계	EO	2	5	A	5
인사	직원정보	ILF	2	15	S	7
인사	직무정보	ILF	1	5	S	7
인사	직원보직	ILF	2	6	S	7
인사	직원면담	ILF	2	5	S	7
인사	출력옵션	ILF	1	5	S	7
인사	근무지정보	EIF	1	4	S	5
도움말	화면헬프	EIF	1	4	S	5
도움말	필드헬프	EIF	1	4	S	5
합계						0

<표 3.6> 기능유형별 가중치 식별 양식

어플리케이션 명	단위 프로세스명 또는 데이터기능 유형	FP유형	RET/FTR	DET	복잡도	가중치
인사	직원정보관리	EI	2	12	A	4
인사	직원정보관리	EI	2	12	A	4
인사	직원정보관리	EI	2	12	A	4
인사	직원정보관리	EO	2	10	A	5

인사	직원정보관리	EQ	1	5	S	3
인사	보직관리	EI	3	7	H	6
인사	보직관리	EO	3	6	H	7
인사	보직관리	EQ	3	4	A	4
인사	보직관리	EQ	1	4	S	3
인사	근무지조회	EQ	2	5	S	3
인사	근무지출력	EO	2	7	A	4
인사	직원면담기록	EI	2	6	A	4
인사	직원면담기록	EI	2	6	A	4
인사	직원면담기록	EI	2	6	A	4
인사	직원면담기록	EQ	2	6	A	4
인사	직원기록대장관리	EI	1	5	S	3
인사	직원기록대장관리	EO	2	6	S	4
인사	부양가족 변동자료작성	EQ	1	8	S	3
인사	고과대상자 자동통보	EO	1	4	S	4
인사	매주말 직원변동상황 자동 리포트	EO	1	8	S	4
인사	수표발행 금액과 내역 수표발행파일 기록	EO	2	10	A	5
인사	타시스템 전달 트랜잭션 파일을 작성	EO	1	6	S	4
인사	화면필드 Help메시지	EQ	1	3	S	3
인사	화면 Help메시지	EQ	1	3	S	3
인사	직원자료변환(구->신)	EI	2	15	A	4
인사	진급대상자 발체	EI	2	4	S	3
인사	정현원 통계	EO	2	4	S	4
인사	부서별 직원통계	EO	2	5	A	5

인사	직원정보	ILF	2	15	S	7
인사	직무정보	ILF	1	5	S	7
인사	직원보직	ILF	2	6	S	7
인사	직원면담	ILF	2	5	S	7
인사	출력옵션	ILF	1	5	S	7
인사	근무지정보	EIF	1	4	S	5
도움말	화면헬프	EIF	1	4	S	5
도움말	필드헬프	EIF	1	4	S	5
합계						0

## ② 가중치 결정

기능점수 유형별 가중치는 기능의 복잡도로 결정되고 복잡도는 3가지가 있다. 각 기능점수 유형별 복잡도 판단 기준은 다음과 같다.

가중치는 기능점수 유형별 기능점수를 파악하고 기능점수를 파악하고자 기능의 복잡도를 측정하는 방법을 제시하기 위함이다.

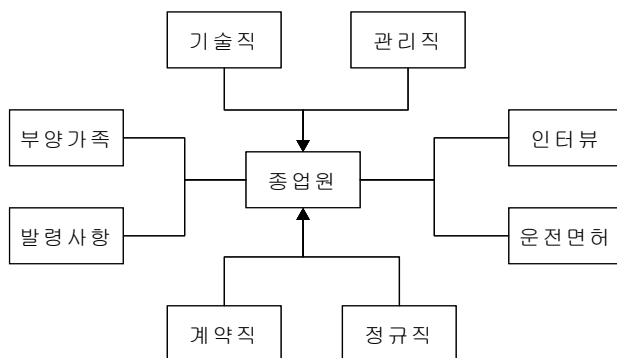
적용은 <표 3.6>에서 하늘색으로 표시한 부분으로 용어는 다음과 같이 정의한다:

- 레코드요소유형(RET: Record Element Type) : 데이터기능유형에서만 식별되는 데이터의 서브그룹으로 선택형(Optional)과 필수형(Mandatory)이 있음
- 참조파일유형(FTR: File Type Referenced) : 트랜잭션기능유형에서만 식별되는 참조파일유형으로 다음과 같이 식별됨
- 데이터요소유형(DET: Data Element Type) : 데이터기능유형이나 트

랜잭션기능유형 모두에서 복잡도를 측정할 때 식별하는 데이터 요소로 사용자가 인지할 수 있고 반복되지 않는 필드

- 복잡도(Complexity) : RET/FTR수와 DET수로 가중치를 파악하기 위해 5가지 기능유형별(ILF/EIF/ET/대/EQ)로 구분한 3가지 등급
- 가중치(Weight) : 각 기능유형별 3가지 등급의 복잡도에 따라 정해진 점수를 의미함

측정의 데이터기능유형(ILF/EIF)의 복잡도 측정기준은 데이터 기능유형의 가중치는 레코드요소유형(RET)의 개수와 데이터항목 유형(DET)의 개수에 따른 복잡도로 결정하며, 레코드요소유형(RET)의 개수 판단은 내부논리파일이나 외부연계파일의 모든 선택적 또는 필수적 데이터 서브그룹에 대해서는 각각을 하나의 RET로 식별하거나 만일 데이터 서브그룹이 없다면 ILF나 EIF 자체를 하나의 RET로 식별한다. (그림 3.3)은 ILF와 RET 개수 판단 예이다.



(그림 3.3) 직원정보 파일의 RET와 ILF 측정 예

위 그림에서 모든 직원은 반드시 기술직이나 관리직 중 하나

를 또 계약직이나 정규직 중 하나를 선택해야 하기 때문에 직원정보는 5개의 RET를 갖는 1개의 ILF와 1개의 RET를 갖는 4개의 ILF로 구성되어 있다.

데이터항목유형(DET)의 개수 판단은 사용자가 인식가능하고 반복되지 않는 필드로 단위프로세스의 수행을 통하여 ILF나 EIF로부터 참조되거나 유지관리 되는 필드를 하나의 DET로 카운트 하며, 두개의 어플리케이션이 같은 ILF/EIF를 유지관리 또는/및 참조할 때 각자가 다른 용도에서 같은 DET들을 사용한다면 어플리케이션별로 용도에 따른 DET 수를 중복 카운트 한다. ("A" 어플리케이션은 주소를 시/도, 시/군/구, 읍/면/동, 번지를 유지관리 목적으로 사용하고, "B" 어플리케이션은 주소를 봉투에 출력하기 위하여 같은 EIF로부터 같은 DET를 사용했다면, "A"는 4개의 DET로 "B"는 '주소'라는 1개의 DET로 카운트 함) 또한 서로 다른 ILF/EIF들끼리 관계를 맺기 위하여 사용된 DET는 각 ILF/EIF에서 하나의 DET로 카운트 한다. (예, 직원번호가 기본 사항에 있으나 경력사항, 가족사항, 학력사항 등에 Foreign Key로 사용되었다면, 각 ILF에서 직원번호를 각자 하나의 DET로 카운트함)

RET와 DET의 개수에 따른 복잡도와 가중치 판단표는 <표 3.7>과 같다.

<표 3.7> 데이터기능유형(ILF/EIF)의 복잡도 판정 기준표 및 가중치

	DET : 1 ~ 19 개	DET : 20 ~ 50 개	DET : 51 개 이상
RET : 1 개	Low	Low	Average
RET : 2 ~ 5 개	Low	Average	High
RET : 6 개 이상	Average	High	High



복잡도 수준	가중치
Low	7
Average	10
High	15

복잡도 수준	가중치
Low	5
Average	7
High	10

트랜잭션기능유형(EI/EO/EQ)의 복잡도 측정기준은 트랜잭션 기능유형의 가중치는 참조파일유형(FTR)수와 데이터항목유형(DET)수에 따른 복잡도로 결정되며, 트랜잭션기능유형에는 외부입력(EI)과 외부출력(EO) 및 외부조회(EQ)로 구분되고, 각각의 복잡도 측정 및 가중치 판단은 다음과 같다.

외부입력(EI)의 경우 FTR 측정기준은 유지관리되는 ILF마다 FTR로 카운트하며, 외부입력 프로세스에 의해서 조회되는 ILF, EIF마다 하나의 FTR로 카운트 한다. 동시에 수정, 조회되는 ILF는 한번만 FTR로 카운트 한다.

DET 측정기준은 어플리케이션 경계 내에 존재하거나 경계 안으로 들어온 데이터로 외부 입력기능 수행에 필요하고 사용자가 인식 가능하며 반복되지 않는 각각의 필드를 하나의 DET로 카운트 하며, 시스템 내부에서 조회되거나 발생하는 필드로 단위 프로세스를 통해 어플리케이션 경계를 벗어나지 않는 필드는 카운트 하지 않는다.

다음은 하나의 DET로 카운트 하는 경우이다.

- 물리적으로는 여러 필드지만 사용자에게는 의미상 하나로 요구되는 것
- 프로세스 수행 중 발생하는 에러 혹은 완료 후 확인하는 메시지 필드

- 기술적인 이유로 ILF 내에 존재하는 필드는 DET로 카운트하지 않음
- 기능키는 한 화면당 여러 종류가 있어도 하나의 DET로 카운트 함

<표 3.8>은 FTR과 DET의 개수에 따른 복잡도와 가중치 판단 표이다.

<표 3.8> 외부입력(EI)의 복잡도 판정 기준표 및 가중치

	DET : 1 ~ 4	DET : 5 ~ 15	DET : 16 이상
FTR : 0 ~ 1개	Low	Low	Average
FTR : 2개	Low	Average	High
FTR : 3개 이상	Average	High	High

복잡도 수준	가중치
Low	3
Average	4
High	6

외부출력(EO)의 경우 FTR 측정기준은 단위 프로세스 수행 중에 조회되는 각각의 ILF, EIF를 하나의 FTR로 카운트 하며, DET 측정기준은 사용자가 인식가능하고 반복되지 않는 필드로 어플리케이션 경계 내부로 들어오고 단위 프로세스에 의해서 언제, 무엇을, 어떻게 생성 또는 조회되는지를 규정하기 위해 필요한 필드를 하나의 DET로 카운트 한다. 사용자가 인식가능하고 반복되지 않는 필드로 어플리케이션 경계 내부에 존재하는 필드를 하나의 DET로 카운트 한다.

다음은 하나의 DET로 카운트하는 경우이다.

- 프로세스 수행 중에 발생하는 에러 혹은 프로세스 완료 후에 확인하는 메시지 필드
- 조회 조건을 선택하는 필드

- 기능키는 한 화면당 여러 종류가 있어도 하나의 DET로 카운트 함

<표 3.9>는 FTR과 DET의 개수에 따른 복잡도와 가중치 판단 표이다.

<표 3.9> 외부출력(EO)의 복잡도 판정 기준표 및 가중치

	DET : 1 ~ 5	DET : 6 ~ 19	DET : 20개 이상
FTR : 0 ~ 1개	Low	Low	Average
FTR : 2 ~ 3개	Low	Average	High
FTR : 4 개 이상	Average	High	High

복잡도 수준	가중치
Low	4
Average	5
High	7

외부조회(EQ)의 경우 FTR 측정기준은 단위 프로세스 수행 중에 조회되는 각각의 ILF, EIF를 하나의 FTR로 카운트 하며 DET 측정기준은 사용자가 인식가능하고 반복되지 않는 필드로 어플리케이션 경계 내부로 들어오고 단위 프로세스에 의해서 언제, 무엇을, 어떻게 생성 또는 조회되는지를 규정하기 위해 필요한 필드를 하나의 DET로 카운트 하고 사용자가 인식가능하고 반복되지 않는 필드로 어플리케이션 경계 내부에 존재하는 필드를 하나의 DET로 카운트 한다.

다음은 하나의 DET로 카운트 하는 경우이다.

- 프로세스 수행 중에 발생하는 에러 혹은 프로세스 완료 후에 확인하는 메시지 필드
- 조회 조건을 선택하는 필드
- 기능키는 한 화면당 여러 종류가 있어도 하나의 DET로 카운트 함

<표3.10>은 FTR과 DET의 개수에 따른 복잡도와 가중치 판단표이다.

<표 3.10> 외부조회(EQ)의 복잡도 판정 기준표 및 가중치

	DET : 0 ~ 5 개	DET : 6 ~ 19 개	DET : 20 개 이상
0 ~ 1 FTR	Low	Low	Average
2 ~ 3 FTR	Low	Average	High
4 or more FTR	Average	High	High

복잡도 수준	가중치
Low	3
Average	4
High	6

### ③ 직접스텝수 산정

기능점수(FP)로 규모측정이 어려운 반면 직접 스텝수로 산정하는 것이 용이한 경우에 한하여 적용하는 것을 목적으로 하며 일반적으로 IFPUG CPM 4.1로 측정하면 규모가 정확히 산정되기 어렵다고 판단이 되는 실시간 처리, 알고리즘이 많은 과학기술계산 등의 어플리케이션 경우에 적용한다.

적용은 <표 3.11>과 같다.

&lt;표 3.11&gt; 직접스텝수 산정 양식

어플리케이션명	단위 프로세스 명	언어	스텝수
교통신호통제시스템	빨간신호에 대기중인 차량의 인식	C	350
교통신호통제시스템	방향지시등을 켜고 대기중인 차량의 인식	C	380
교통신호통제시스템	보행자버튼의 활성화 상태 점검	C	400
교통신호통제시스템	기관차통제시스템과의 상황정보 교환 및 유지관리	C	1,340
교통신호통제시스템	모든 신호등을 통제하는 컨트롤박스 관리	C	1,654
교통신호통제시스템	비상용 차량 인식 및 관련장비의 비상상태 활성화	ASM	2,430
교통신호통제시스템	교통신호통제장치에 정상 또는 비상 상태 전달	ASM	1,330
교통신호통제시스템	장비의 비정상 상태 감지 및 관련 장비에 상태 전달	ASM	2,300
합 계			

단위프로세스(Elementary Process) 식별기준은 직접 스텝수를 산정하기에 가장 적합한 단위의 프로세스를 식별하며 사용자의 관점에서 사용자 요구의 식별이 가능하도록 한다. 개발언어별 스텝수를 산정해야 하므로 언어의 구분이 가능하게 식별한다.

개발언어 식별기준은 기능의 구현에 사용된 프로그램 언어를 식별한다.

스텝수 식별기준은 각 언어의 소스라인 식별 기준을 따르며 명령어와 명령의 대상이 함께 나타나 있게 코딩되어 있는 소스라인을 하나의 라인으로 식별한다. 여기서 주석문은 식별대상에서 제외되며, 프로그램 환경선언, 데이터선언, 라이브러리 등을 복사하여 사용할 경우에는 실행명령 스텝수만을 계산하도록 한

다. 다음은 활용 예이다.

(다음 예의 경우는 7개 스텝으로 카운트 함)

```
#INCLUDE <STDIO.H>      ①
#INLCUDE <DOS.H>        ②
MAIN()
{INT A, B, D, E, X, Y, PAY, BILL;  ③
  :
  :
  IF (A==B) Y = X; ④
  ELSE Z = X;      ⑤
  BILL = BILL + D; ⑥
  PAY = PAY + E;   ⑦
```

### 다. 3 단계 - 기초인건비 산정

기초인건비 산정은 다음과 같이 작성된다.

- 제1단계 '범위정의'에서 식별된 컴포넌트 단위로 제2단계 '규모산정'에서 측정된 기능점수 또는 스텝수에 기준 기능점수당 단가 또는 스텝당 단가를 곱하고 각 해당공정 및 컴포넌트의 값을 더하여 인건비를 구한다.
- 추가공정이 식별되어 있으면 해당 추가공정의 비중을 감안하여 추가공정에 해당되는 인건비를 추가하여 기본공정에서 구한 인건비에 더하여 기초인건비를 구한다.

직접인건비를 산정하기 위해 규모에 규모당 단가를 곱하여 기초인건비를 산정한다.

'2단계'에서 산정된 어플리케이션 규모에 다음과 같이 규모당 단가를 적용하여 기초인건비를 산정하며 추가공정은 제외한다:

$$\text{기초인건비} = \text{규모(FP)} * \text{FP당 단가}$$

### ① 규모당 단가 적용

개발자의 생산성에 영향을 미치는 어플리케이션 규모와 형태 및 하드웨어 유형 등을 고려하지 않고 업계의 평균생산성만으로 정한 규모당 단가를 적용하여 인건비를 구하며 규모당 단가는 규모산정을 기능점수(FP)로 한 경우와 스텝수(LOC)로 한경우로 구분이 되므로 기능점수일 경우는 기능점수당 단가를, 스텝수일 경우는 스텝당 단가를 각각 적용한다.

기능점수(FP)당 단가 적용은 <표 3.12>의 기능점수당 단가 참조한다. <표 3.12>의 13개 공정별 기능점수(FP)당 단가(원)를 2 단계에서 기능점수로 산정한 어플리케이션의 규모에 곱하여 기초인건비를 구한다. 이 때 주의할 점은 13개의 모든 공정에 무조건 적용하는 것이 아니고 1 단계에서 식별한 어플리케이션별 해당 공정에만 적용하여 계산한다. 1 단계에서 수행하지 않겠다고 식별한 공정은 계산에서 제외된다. <표 3.12>의 공정별 단가에는 개발활동 이외의 프로젝트 관리, 행정지원 등의 관리활동과 품질 보증, 문서작업, 형상관리 등의 지원활동에 필요한 비용도 포함되어 있는 것으로 간주하여 책정되었다.(평균 15% 정도) 기본공정 이외에 추가적인 요구공정(<표 3.4> 참조)에 소요되는 비용은 1 단계에서 식별한 추가활동 비중(기본공정을 100%로 볼 때 추가활동에 소요될 노력의 정도로 표시함)을 기본공정 합계액에 곱하여 산정한다.

&lt;표 3.12&gt; 공정별 규모당 단가

순번	공정명 (ISO 12207)	기능점수(FP)당 단가(원)	스텝당 단가(원)
1	공정구현	14,116	253.2
2	시스템 요구분석	12,533	224.9
3	시스템 구조설계	12,993	233.1
4	소프트웨어 요구분석	18,349	329.2
5	소프트웨어 구조설계	18,058	324.0
6	소프트웨어 상세설계	20,515	368.1
7	소프트웨어 코딩 및 시험	35,250	632.4
8	소프트웨어 통합	15,362	275.6
9	소프트웨어 자격시험	14,353	257.5
10	시스템 통합	14,678	263.3
11	시스템 자격시험	12,385	222.2
12	소프트웨어 설치	11,387	204.3
13	소프트웨어 수락지원	9,794	175.7
합계		209,773	3,763.5

스텝당 단가 적용은 <표 3.12>의 스텝당 단가를 적용하는데 <표 3.12>의 13개 공정별 스텝(LOC)당 단가(원)를 2 단계에서 직접스텝수로 산정한 어플리케이션의 규모에 곱하여 기초인건비를 구한다.

이 때 주의할 점은 13개의 모든 공정에 무조건 적용하는 것이 아니고 1 단계에서 식별한 어플리케이션별 해당 공정에만 적용하여 계산한다. 1 단계에서 수행하지 않겠다고 식별한 공정은 계산에서 제외된다. <표 3.12>의 공정별 단가에는 개발활동 이외의 프로젝트 관리, 행정지원 등의 관리활동과 품질보증, 문서작업, 형상관리 등의 지원활동에 필요한 비용도 포함되어 있는 것으로



간주하여 책정되었다.(평균 15% 정도) 기본공정 이외에 추가적인 요구공정(<표 3.4> 참조)에 소요되는 비용은 1 단계에서 식별한 추가활동 비중(기본공정을 100%로 볼 때 추가활동에 소요될 노력의 정도로 표시함)을 기본공정 합계액에 곱하여 산정한다.

#### 라. 4 단계 - 직접인건비 산정

직접인건비는 다음과 같이 산정된다.

- 단순히 규모에 규모당 단가를 곱하여 구한 기초인건비에 프로젝트 특성과 제품(어플리케이션)의 특성을 감안하여 최종 직접인건비를 구한다.
- 프로젝트 특성은 개발언어보정, 어플리케이션 규모보정이고 제품 특성은 어플리케이션 유형보정 및 하드웨어 유형보정이다.
- 개발언어보정은 프로그램언어가 구축에 직접적인 영향을 미치는 5개 개발관련 공정에만 적용된다.
- 규모보정은 프로젝트 단위보다는 응집도가 높은 어플리케이션 단위로 적용된다.

프로젝트 수행에 관련된 제반 요소인 사람, 기술, 환경, 제품 등의 특성을 종합한 여러 가지 보정계수를 감안하여 프로젝트 수행에 직접적으로 소요되는 인건비를 산정하고자 함이다.

‘3단계’에서 산정된 기초인건비에 다음의 4가지 보정계수를 곱하여 산정하고, 이렇게 산정된 금액에 추가공정별(추가적인 요구활동)로 식별된 비율을 곱하여 구한 추가공정비용을 합하여 직접인건비를 산정한다.

$$\text{직접인건비} = \{\text{기초인건비} * \text{개발언어 보정계수}(5\text{개 공정}) * (\text{규모보정계수} * \text{어플리케이션 유형 보정계수} * \text{하드}$$

웨어 유형 보정계수}} \* (1 + 추가공정 비율 합계)

### ① 개발언어 보정계수

개발언어의 생산성 정도에 따라 소요공수에 많은 차이를 보이고 있음을 감안한다. 1세대 언어와 4세대 언어의 생산성은 경우에 따라서는 10배 이상을 보이기도 한다. 특히 같은 어플리케이션 내에서도 다양한 개발언어가 사용됨을 감안하여 적절한 생산성 보정계수가 적용될 수 있도록 했다.

아래 <표 3.13>, <표 3.14>, <표 3.15>를 보면 ‘1 단계’에서 식별한 어플리케이션 구분별로 보정계수를 적용하며 개발언어 유형별 보정계수는 다음 <표 3.13>과 같이 5개 유형으로 분류하여 적용한다.

<표 3.13> 개발언어 유형별 보정계수

개발 언어 유형	보정 계수
4세대 언어(4GL, Javascript)	0.65
RPG	0.70
고급언어 (COBOL, FORTRAN, PL/1, PASCAL, BASIC 등)	1.00
C, C++, Java, Perl	1.30
Assembly	2.00

<표 3.14> 13개 개발공정중 개발언어 보정계수 적용대상 공정

순번	공정명	보정계수 적용범위
1	공정구현	해당 무
2	시스템 요구분석	해당 무
3	시스템 구조설계	해당 무
4	소프트웨어 요구분석	해당 무
5	소프트웨어 구조설계	해당 무
6	소프트웨어 상세설계	해당 무
<b>7</b>	<b>소프트웨어 코딩 및 시험</b>	<b>해당</b>
<b>8</b>	<b>소프트웨어 통합</b>	<b>해당</b>
<b>9</b>	<b>소프트웨어 자격시험</b>	<b>해당</b>
<b>10</b>	<b>시스템 통합</b>	<b>해당</b>
<b>11</b>	<b>시스템 자격시험</b>	<b>해당</b>
12	소프트웨어 설치	해당 무
13	소프트웨어 수락지원	해당 무

개발언어 보정계수는 모든 공정에 적용하지 않고 개발과 직접적으로 관련이 있는 소프트웨어 코딩 및 시험, 소프트웨어 통합, 소프트웨어 자격시험, 시스템 통합, 시스템 자격시험 등 5개 공정에만 적용한다.

하나의 어플리케이션 내에서 여러 종류의 개발언어를 사용하는 경우는 해당 언어 수준별 적용율을 파악하여 적용한다.

<표 3.15> 같은 어플리케이션에서 여러 종류의 개발언어를 사용하는 경우

개발 언어 유형	보정계수	적용율	적용 보정계수
4세대 언어(4GL, Javascript)	0.65	50%	0.38
RPG	0.70	0%	0.00
C, C++, Java, Perl	1.30	25%	0.33
고급언어(COBOL, FORTRAN, PL/1, PASCAL, BASIC 등)	1.00	25%	0.25
Assembly	2.00	0%	0.00
언어보정계수 종합		100%	0.90

## ② 어플리케이션 규모 보정계수

프로젝트 규모가 클수록 증가하는 투입인력으로 커뮤니케이션 채널이 늘어나 생산성이 떨어지기(Frederick Brooks, The Mythical Man Month) 때문에 규모보정가 증가함에 따른 보정이 필요하다.

적용은 <표 3.16> 및 (그림 3.4)에서 ‘1 단계’에서 식별한 어플리케이션 구분별로 보정계수를 적용하며 규모를 기능점수로 산정했으면 다음의 함수식을 적용하여 규모보정계수를 식별한다:

$$\text{규모 보정치} = 0.108 * \text{Log } e (\text{규모}) + 0.2229$$

규모를 스텝수로 산정했으면 다음의 함수식을 적용하여 규모보정계수를 식별한다:

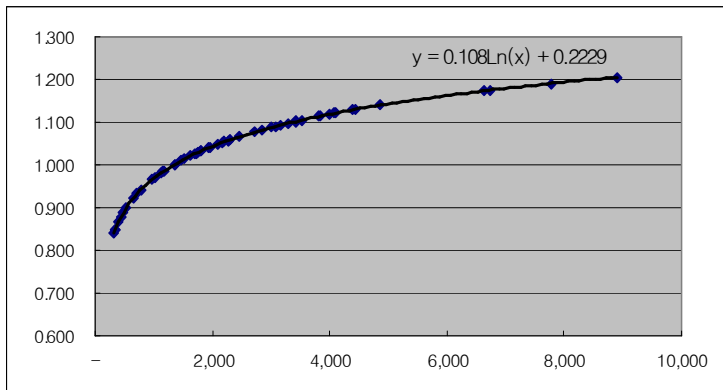
$$\text{규모보정계수} = 0.108 * \text{Log } e (\text{스텝수}/56) + 0.2229$$

300 FP 미만은 아무리 작아도 0.75 미만의 보정계수를 적용하지 않으며 규모보정은 규모의 증가에 따른 인터페이스 노력이 많이 소요되는 것을 고려하고자 함이므로, 원칙적으로 적용 대상 어플리케이션을 구성하는 서브시스템간의 응집도가 높아야 한다. 만일 어플리케이션간의 응집도가 높으면 어플리케이션별로 규모 보정치를 적용하는 것보다 이들을 통합한 규모로 보정치를 적용하는 것이 타당하다.

<표 3.16> 규모 보정계수

어플리케이션 규모 (FP)	어플리케이션 규모 (스텝수)	규모보정 계수
300미만	24,000 미만	0.750
300	24,000	0.839
500	40,000	0.894
1,000	80,000	0.969
2,000	160,000	1.044
3,000	240,000	1.088
5,000	400,000	1.143
10,000	800,000	1.218
20,000	1,600,000	1.292
30,000	2,400,000	1.336
50,000	4,000,000	1.391
100,000	8,000,000	1.466
200,000	16,000,000	1.541

주1) 300 FP를 출발점으로 하여 그 이하일 경우는 일률적으로 0.75를 적용한다.



(그림 3.4) 어플리케이션 규모 보정계수 함수

### ③ 어플리케이션 유형 보정계수

같은 규모의 어플리케이션이라도 그 특성별 유형에 따라 개발자의 생산성이 달라짐을 감안하여 이를 반영하며 데이터나 정보처리가 강한 어플리케이션 보다 통제 및 통신이 강한 어플리케이션이나 복잡한 계산이 많은 알고리즘 위주의 어플리케이션이 훨씬 개발에 많은 노력이 소요되는 점을 반영한다.

적용은 아래 <표 3.17>과 같이 ‘1 단계’에서 식별한 어플리케이션 구분별로 보정계수를 적용하고 대상 어플리케이션이 여러 어플리케이션 유형 중 어디에 해당되는 지를 예시를 참조하여 구분한다. 대상 어플리케이션이 복합적이어서 여러 유형에 해당된다고 판단이 되면, ‘1 단계’로 돌아가서 어플리케이션 구분을 더 세분화할 필요가 있는지 검토한다. 특별히 어플리케이션 유형의 적용은 보정계수가 크기 때문에 신중하게 적용해야 하고, 애매할 경우는 어플리케이션을 세분하는 방안을 적극 검토하도록 한다.

〈표 3.17〉 어플리케이션 유형별 보정계수

어플리케이션 유형	예 시	보정 계수
사무처리형 (MIS)	인사, 회계, 급여, 영업 등 경영 관리용 및 특정 분야의 업무용 소프트웨어 등	1.0
지식기반형	지식 기반으로 수행되는 모든 어플리케이션	1.3
과학/수학/의사 결정지원형	과학계산, 시뮬레이션, 스프레드시트, 통계, OR, CAE 등	1.5
지능정보형	그래픽, 영상처리, 음성처리, 멀티미디어 응용분 야, 교육·오락용 등	2.0
멀티미디어형	자연어처리, 인공지능, 전문가시스템	2.0
시스템용	운영체제, 언어처리 프로그램, DBMS, 인간-기계 인터페이스, 윈도시스템, CASE, 유틸리티 등	2.4
통신/네트워크 용	통신프로토콜, 에뮬레이션, 교환기소프트웨어 등	3.0
공정제어/실시 간(실장)형	생산지원, CAM, CIM, 기기제어, 로봇분야 등	3.2
지휘통제형	군, 경찰 등 군장비·인력의 지휘통제를 요하는 소 프트웨어	4.0

#### ④ 하드웨어 유형 보정계수

개발 어플리케이션이 운용될 하드웨어 아키텍처의 복잡성 정도에 개발 노력이 많은 차이를 보이고 있음을 감안한다. 특히 다중 터어 구조를 가진 클라이언트/서버형 하드웨어 구조를 가진 경우는 메인프레임만이나 개인용 컴퓨터만으로 구성된 경우와는 노력의 차이를 상당부분 인정해 주고 있음을 감안한다.

작성은 아래 〈표 3.18〉과 같이 ‘1 단계’에서 식별한 어플리케이션 구분별로 보정계수를 적용하도록 하나 만일 동일 프로젝트

에서 구축하는 모든 어플리케이션이 같은 하드웨어 구조를 공유하면 프로젝트단위로 식별해도 무방하다. 그러나 어플리케이션별로 운용환경이 다르면 이를 면밀히 검토하여 하드웨어 유형 보정계수의 적용이 남용되지 않도록 해야 한다.

<표 3.18> 하드웨어 유형별 보정계수

하드웨어 유형	보정계수
CLIENT/SERVER	1.1
MAINFRAME	1.0
PC	0.8

#### 마. 5 단계 - 제경비 계산

제경비는 다음과 같이 계산된다.

- 제경비는 직접인건비에 무조건 110%를 적용하도록 고정하였다.
- 과거기준에 있던 납기단축에 따른 보정은 적용의 어려움과 현실적이지 못하여 삭제하였다.

프로젝트를 간접적으로 지원하는데 소요될 간접비, 판매비, 일반관리비 등을 감안한다.

적용은 ‘4단계’에서 산정된 직접인건비의 110%를 고정율로 적용한다.

$$\text{제경비} = \text{직접인건비} * 110\%$$

#### 바. 6 단계 - 기술료 계산



기술료는 다음과 같이 계산된다.

- 기술료는 20%를 기본으로 적용한다.
- 다만, 일반적인 품질이 아닌 특별히 품질보증이 필요하고 그 적용에 대한 확실한 근거가 있다고 인정되는 경우만 10%를 추가로 인정해준다.

프로젝트 수행에 적용된 기술 개발에 소요된 개발자의 투자를 인정할 뿐 아니라, 국가차원에서 소프트웨어 산업의 발전을 촉진하는데 활용할 수 있는 기술개발 투자를 인정하고자 한다.

적용은 ‘4단계’에서 산정된 직접인건비와 ‘5단계’에서 계산된 제경비를 합한 금액의 20%를 기본으로 인정하고, 만일 고객이 특별한 품질을 요구하여 상호 합의가 이루어졌다면 기술료를 최고 10%까지 추가로 인정할 수 있다:

$$\text{기술료} = (\text{직접인건비} + \text{제경비}) * (20\% \sim 30\%)$$

#### 사. 7 단계 - 직접경비 계산

직접경비는 다음과 같이 계산된다.

- 컴퓨터 사용료, 소프트웨어 개발도구 사용료
- 지급이자, 전문가비용, 여비
- 특수자료비, 인쇄, 청사진비
- 자료조사비, 기자재시험비
- 위탁비와 현장운영비
- 모형제작비
- 기타

프로젝트 수행에 직접적으로 소요되는 여비, 인쇄비 등의 직접성 경비를 프로젝트 비용에 감안한다.

적용은 해당 프로젝트에 특별히 필요로 하는 컴퓨터시스템 사용료는 소프트웨어 사업 추진시 사용되는 컴퓨터기기(호스트컴퓨터, 단말기 및 OA기기 등)를 발주자가 제공하지 않아 자체 컴퓨터기기나 제3자의 컴퓨터기기를 사용하여야 할 경우 그 규모와 종류에 따라 시스템 사용료가 달라질 수 있으나 당해 컴퓨터기기 보유업체의 기준이나 시스템 사용료 산정기준에 따라 산출한다. 일반적으로, 장비사용료(또는 시스템사용료)의 산정방법은 장비를 임차할 경우에는 그 임차료, 개발회사 소유일 경우에는 법인세법에 의한 감가상각비 등을 적용함이 타당하다. 당해 프로젝트에 특별히 필요로 하는 소프트웨어 Tool사용료는 발주자의 요구에 의해 사용해야 하는 소프트웨어패키지, CASE Tool 등의 구입 또는 사용료 등이며 선투자 후정산 사업으로 추진되는 사업의 경우 지급이자는 제경비 항목의 정상적인 지급이자와는 성격이 상이하므로 직접경비에 별도 계상한다.

발주자의 요구에 의한 특정기술 도입과 관련된 전문가 비용은 사업의 수행을 위하여 관련전문가(예: 공인회계사)를 활용할 경우 직접경비로 계상한다.

- 발주자의 요구' 는 수주자의 제안내용을 발주자가 채택한 경우를 포함함
- '전문가'는 특정기술도입과 관련하여 국가공인자격증(예: 공인회계사) 또는 국제공인자격증을 소지한 자 또는 특정 외국기술 적용을 위해 투입되는 외국인 기술자를 의미함

멀티미디어 데이터제작시 발주자가 특별히 요구하는 성우, 배우 등 특정인물을 활용하는 경우 관련전문가 활용으로 직접경비

로 계상한다.

- 전문가 비용은 직접경비로 계상함이 타당하며 그 산출기준은 관련 법규 및 규정(예: 공인회계사 보수기준)을 참조하여 계상함. 단 적절한 관련법규 및 규정이 없는 경우는 수주자의 제안금액을 기준으로 발주자와 협의하여 산출함

당해 업무수행에 직접 필요한 여비는 지방에 체류하며 소프트웨어 사업 업무를 수행할 경우, 체류비용은 직접경비의 '여비'항목으로 계상함. 여비에는 운임(철도, 선박, 항공, 자동차), 현지교통비, 숙박비 등이 포함한다.

특수자료비(특허, 노-하우등의 사용료)는 당해 프로젝트 수행에 필요한 특허 사용료, 노-하우 사용료를 말하며 제출문서의 인쇄·청사진비는 당해 프로젝트 수행시 공정별 산출물 및 보고서의 인쇄비 및 복사비등을 이다. 자료조사비는 당해 프로젝트 수행시 소요되는 문헌, 전문도서등의 구입과 이에 소요되는 비용이다. 현장운영비는 당해 프로젝트 수행에 필요한 보조요원의 급여와 수주자가 현장에 사무실을 설치하여 운영할 경우의 현장운영경비이다. 모형제작비는 발주자의 요구사항을 명확히 하기 위해 본 프로젝트에 들어가기 전에 모형을 제작할 때 소요되는 비용이다.

직접경비를 계상시에는 정확한 내역을 제시해야 하고 경비집행의 기준을 계약상 명시해야 하며 또한 직접경비는 사후 정산에 대비하여 집행근거 및 증빙이 반드시 있어야 한다.

#### 아. 8 단계 - 소프트웨어 개발비 산정

소프트웨어 개발비는 다음과 같이 계산된다.

- 소프트웨어 개발비는 개발비 산정기준에서 궁극적으로 구하고자 하는 값이다.
- 4가지 값을 합산하여 계산한다 : 소프트웨어 개발비 = 직접인건비 + 제경비 + 기술료 + 직접경비

소프트웨어 개발에 소요될 총 비용을 알고자 하며 적용은 ‘4단계’의 직접인건비, ‘5단계’의 제경비, ‘6단계’의 기술료, ‘7단계’의 직접경비를 합산하여 구한다:

**소프트웨어 개발비** = 직접인건비 + 제경비 + 기술료 + 직접경비

#### 자. 9단계 - 공수 산정(대가기준에는 미포함)

공수는 다음과 같이 계산된다.

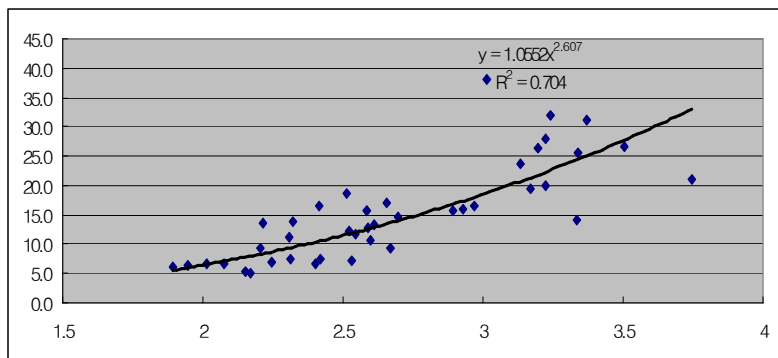
- 공수산정은 대가기준에는 포함되어 있지 않으나 개발자의 원가계산에 필요한 부분이기 때문에 포함시킨 것으로 소프트웨어 개발비 계산에는 포함되지 않는 절차이다.
- 공수산정은 개발자의 생산성을 산정된 규모에 적용하여 구한다.
- 생산능력인자를 가지고 구한 생산능력지수를 생산함수에 대입하여 생산성을 구한다.
- 인건비단가는 소프트웨어 노임단가를 기준으로 하거나 개발자의 인건비 단가를 기준으로 한다.
- 생산성으로 구한 소요공수에 인건비 단가를 곱하여 기초인건비를 구하고 나머지는 다른 절차와 같은 방법으로 구한다.

산정된 규모의 일을 수행하려면 개발자가 자신의 생산성 수준에서 어느 정도의 공수를 투입해야 할지를 알 수 있도록 해주어야 한다. 적용은 ‘2단계’에서 산정된 규모를 개발자의 생산성으로 나누어 소요공수를 산정한다.

### ① 개발자 생산성 수준 파악

소요공수를 파악하고자 개발자 자신의 생산성을 파악하고자 하며 적용은 <표 3.19>에서 '1 단계'에서 식별한 어플리케이션 구분별로 개발팀의 생산성 수준을 파악하며 사람, 기술, 프로세스 구분별 총 32개 인자에 대한 개발팀의 능력을 5개 수준으로 평가하면 각각에 해당 가중치를 곱한 값의 합계 (가중치의 합)을 다음 식에 대입하여 생산성을 구한다.(그림 3.5 참조)

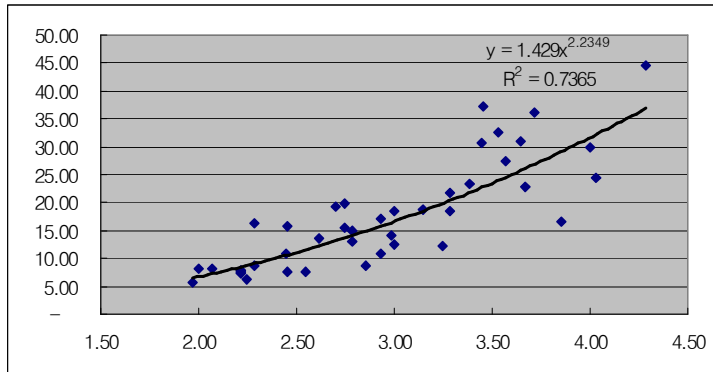
$$\text{생산성 (FP/MM)} = 1.0552 * (\text{총 가중치의 합} / 1000)^{2.607}$$



(그림 3.5) 32개 모든 인자를 아는 경우의 생산성 함수

사람, 기술, 프로세스 구분별 총 32개 인자 중 핵심인자 6개 (<표 3.19>의 색칠한 인자)인 관리능력(리더쉽, 의사소통, 문제해결), 분석/설계/개발 능력, 품질 보증/관리 능력, 요구분석/설계 검토 참여도, 개발 도구 성능, 사용자 요구의 명확도 및 안정성 등만 알아도 개발팀의 능력을 다음 식에 대입하여 생산성을 구할 수 있다. (그림 3.6) 참조

**생산성 (FP/MM) = 1.429 \* (총 가중치의 합 / 350)<sup>2.2349</sup>**



(그림 3.6) 핵심 6개 인자만 아는 경우의 생산성 함수

&lt;표 3.19&gt; 개발자 생산성 수준 파악

능력 요소	능력 인자	능력 인자별 세부 항목	평가 수준					가중치
			탁월	우수	보통	미흡	미달	
사람 (People)	PM	관리능력(리더쉽, 의사소통, 문제해결)			X			100
		유사 프로젝트 관리 경험		X				60
		프로젝트관리 도구 활용 능력			X			40
	개발자	유사 시스템 개발 경험			X			30
		분석/설계/개발 능력			X			60
		분석/설계/개발 도구 활용 능력			X			45
		개발 플랫폼 경험		X				15
	품질관리자	유사 프로젝트 개발/품질관리 경험			X			15
		품질 보증/관리 능력			X			25
		품질관리 도구 활용 능력			X			10
	사용자	정보시스템 프로젝트 경험			X			10
		유사 시스템 사용 경험		X				20
		요구분석/설계검토 참여도		X				40
		인수시험 참여도		X				30
기술 (Technology)	개발 도구	분석/설계 도구 성능			X			45
		개발 도구 성능			X			75
		테스트/디버깅 도구 성능			X			30
	지원 도구	프로젝트 관리 도구 성능			X			30
		품질관리 도구 성능			X			16
		형상관리 도구 성능				X		12
	개발 플랫폼	개발 플랫폼 성능			X			45
		도구(개발, 지원)의 플랫폼 적합성			X			27
		라이브러리(소스/문서)의 효과(율)성				X		9

프로세스 (Process)	요구수렴 절차	사용자 요구의 명확도 및 안정성				X		50
		사용자 요구분석 방법의 효과성 및 숙련도				X		30
		프로토타이핑 방법의 효과성				X		20
	작업수행 절차	시스템 개발방법론의 효과성 및 숙련도				X		30
		품질보증과 테스트 방법의 효과성 및 숙련도				X		18
		문서화 숙련도 및 납품절차의 효율성				X		12
	측정·개선 절차	측정 기준/방법의 구체성				X		12
		측정 실시 및 데이터 관리				X		20
		성과분석 및 피드백의 원활				X		8
생산성 수준 (전체 요소)			17.19					
생산성 수준 (6대 요소)			17.46					

## ② 기초 소요공수 산정

직접소요공수를 산정하기 위해 규모에 개발자 생산성을 적용하여 기초소요공수를 산정한다. 적용은 ‘2단계’에서 산정된 어플리케이션 규모에 ‘① 개발자 생산성 수준’을 적용하여 기초소요공수를 산정하며 추가공정은 제외한다.

$$\text{기초소요공수} = \text{규모(FP)} / \text{개발자 생산성 수준 (FP/MM)}$$

## ③ 직접 소요공수 산정

프로젝트 수행에 관련된 제반 요소인 사람, 기술, 환경, 제품 등의 특성을 종합한 여러 가지 보정계수를 감안하여 프로젝트 수행에 직접적으로 소요되는 공수를 산정한다. 적용은 ‘② 기초



소요공수 산정'에서 산정된 기초소요공수에 '4단계 직접인건비 산정'에서 구한 4가지 보정계수를 곱하여 산정하고, 이렇게 산정된 공수에 추가공정별(추가적인 요구활동)로 식별된 비율을 곱하여 구한 추가공정별 공수를 합하여 직접 소요공수를 산정한다.

$$\begin{aligned} \text{직접소요공수} = & \{ \text{기초소요공수} * \text{개발언어 보정계수(5개 공정)} * \\ & (\text{규모보정계수} * \text{어플리케이션 유형 보정계수} * \\ & \text{하드웨어 유형 보정계수}) \} * (1 + \text{추가공정 비율 합계}) \end{aligned}$$

#### ④ 원가 산정

개발자의 생산성에 기반하여 개발자 자신의 원가를 파악하며 사업대가 기준으로 제안할 시 자신의 원가와 비교하여 어느 정도의 손익이 예상되는지를 알고자 한다. 고객의 예가를 알고 있을 때 그에 맞추기 위해서 개발자가 목표로 삼아야 할 생산성 수준을 알고자 한다.

적용은 '③ 직접 소요공수 산정'에서 산정된 직접소요공수에 소프트웨어 노임단가 <표 3.20>를 적용하거나 개발자 자신의 인건비 단가를 적용하여 직접인건비를 산정한다.

〈표 3.20〉 기술등급별 소프트웨어 노임단가기준표 (2002년도 기준)

기술 등급	일단가(8시간)	월단가(25일)	투입율	기대치
기술사	180,281	4,507,025	2%	90,141
특급기술자	166,514	4,162,850	3%	124,886
고급기술자	132,341	3,308,525	20%	661,705
중급기술자	104,809	2,620,225	35%	917,079
초급기술자	79,524	1,988,100	10%	198,810
고급기능사	70,850	1,771,250	15%	265,688
중급기능사	65,879	1,646,975	10%	164,698
초급기능사	49,883	1,247,075	5%	62,354
평균인건비	106,260	2,656,503	100%	2,485,359

주1) '투입율'은 가상치이고 '기대치'는 '투입율'에 따른 기대치로 예시임

### 3.2 데이터베이스 구축 대가 산정

급속히 늘어나는 정보 수요를 충족시키기 위해 데이터베이스 제공업자 및 정부는 최근 몇 년 동안 데이터베이스 산업육성에 힘을 기울였다. 그 일환으로 데이터베이스진흥정책이 수립되어 그 중요성 인식과 보급에 괄목할 만한 성장을 거두게 되었다. 그러나, 데이터베이스 산업의 급격한 성장에도 불구하고 데이터베이스에 대한 가치 인식 미비, 장기적인 안목에서의 과감하고 적극적인 기술 투자 미흡, 전문 기술인력의 만성적인 부족, 국내 시장 협소, 데이터베이스 구축업체의 영세성 등 질적인 수준은 데이터베이스 산업의 가시적인 성장에 미치지 못하고 있다.

특히, 데이터베이스 구축을 발주 또는 수주 할 경우 객관적이고 합리적인 데이터베이스 구축 대가 산정 기준과 연구 결과가 부족하여 데이터베이스 개발자들이 노력한 만큼의 정당한 가치를 인정받지 못하게 되는 어려움을 겪고 있다. 이러한 개발자의 어려움은 구축업체의 채산성 및 재무구조를 악화시키고, 결국 데이터베이스 품질과 연결되어 악순환을 거듭하게 되는 것이다.

이에 1994년 한국데이터베이스진흥센터에서 『데이터베이스 구축비용 산정기준 및 서비스 이용 요금 체계 설정을 위한 기초조사』 연구를 실시하였으며, 정보 산업의 네트워크화, 오픈시스템화, 멀티미디어화, 다운사이징화 등의 추세에 따라 SI체제로의 정보시스템의 구축 프로젝트가 일반화되면서, 97년 정보통신부 고시 제1997-57호로 『소프트웨어 사업 대가 기준』에 데이터베이스 구축 대가 산정기준이 포함되어 고시되었다. 그러나 정보기술의 급속한 발전, 사회 각 부문의 폭발적인 정보화 요구의 증가로 데이터베이스 개발 사업의 복잡화, 대규모화가 이루어지고 있는 환경에서 기존 『소프트웨어 사업 대가 기준』으로는 성공적

인 데이터베이스 구축, 운영을 실행하기 어려운 상황이다.

### 3.2.1 범위

데이터베이스 구축 대가기준의 적용범위는 국가, 지방자치단체, 정부투자기관 및 기타 공공기관이 데이터베이스 구축 개발용역 사업을 개발업자에게 위탁하기 위해 데이터베이스 구축비용을 산정 하고자 하는 경우, 또는 기타 기관에서 DB 구축에 대한 합리적 대가 산정을 할 경우 이 기준을 적용할 수 있다.

데이터베이스 대가기준 제안을 위하여 사용되는 용어는 다음과 같이 정의하며, 표현되는 단어는 일반적으로 사용되는 의미와 상이(相異)할 수 있다.

- 데이터베이스 구축 : 사회적으로 산재해 있는 방대한 원시자료를 수집·분류·분석하여 이용자에게 유용한 의미 있는 정보로 설계하여 수집한 자료를 가공·제작·제공하는 일련의 작업
- 데이터베이스 설계 : 사용자의 요구사항으로부터 그들에게 필요한 정보를 제공할 수 있도록 데이터베이스 구조를 개발하는 과정이며, ER다이아그램, 각종 명세서 및 지침서 등이다.
- 데이터제작 : 수집된 원시자료를 일련의 체계화된 작업절차에 따라 이용자에게 유용하고 의미 있는 정보형태로 전환하여 표현하는 작업

데이터베이스 구축 대가 기준 적용에 있어 데이터베이스 설계에서는 공수 구분을 위하여 유형을 구분하며, 데이터 제작에서는 공정 및 공수 구분을 위하여 미디어별로 구분하였다. 공수산정에 있어 데이터베이스 설계를 기획, 설계, 편집으로 데이터 제작에 있어 준비, 제작, 검사로 공정 범위를 정의하였다. 또한 각 공정별 세부 공정, 공정별 업무범위를 정의하였고 데이터 제작의

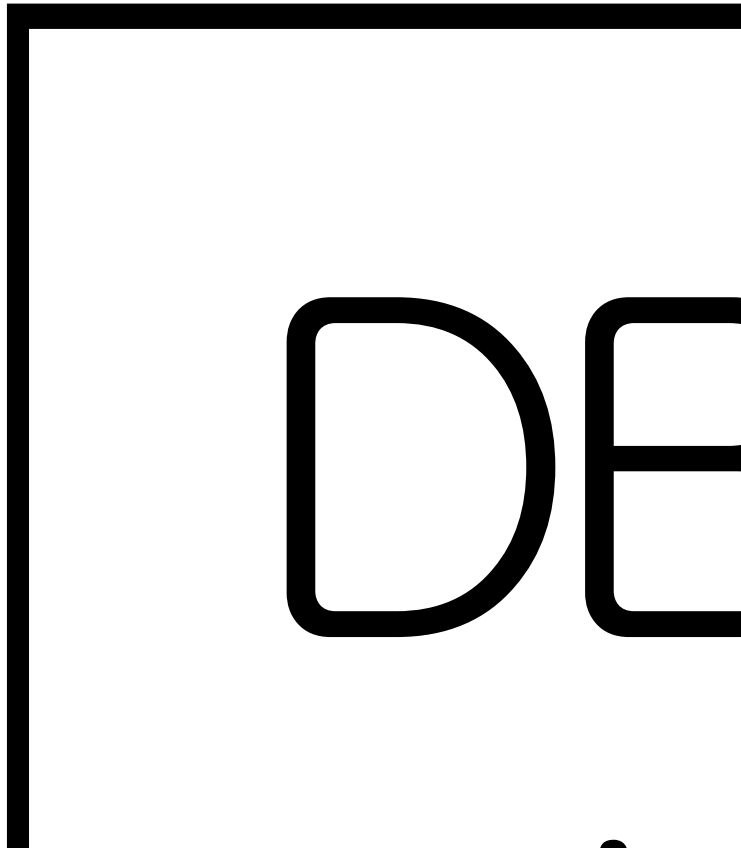
경우 미디어별 공정을 구분하여 세분화 하였다. 따라서 공정 적용에 있어 연구에서 정의한 공정을 기준으로 비용을 산정할 수 있다.

### 3.2.2 사례분석

국내 정보통신부 고시 제1997-57호 『소프트웨어 사업 대가 기준』에서의 데이터베이스 구축비 산정 기준과 데이터베이스 구축 비용산정과 관련된 국내외 이론에 대해 살펴본다.

#### 가. 구축비 산정 기준의 구성

정보 산업의 네트워크화, 오픈시스템화, 멀티미디어화, 다운사이징화 등의 추세에 따라 SI체제로의 정보시스템의 구축 프로젝트가 일반화되면서, 97년 정보통신부 고시 제1997-57호로 『소프트웨어 사업 대가 기준』에 데이터베이스 구축비용 산정기준이 포함되어 고시되었다.



(그림 3.7) 현행 데이터베이스 구축 비용 산정 절차

이렇게 고시된 현행 대가기준은 이후 부분적으로 몇 번의 수정을 거쳐 현행 고시되어 적용되고 있었다.

데이터베이스 구축비 현행 산정 기준은 (그림 3.7)과 같이 데이터베이스 설계비와 직접경비로 구분되며, 직접경비에는 원시자료 수집비, 데이터제작비, 메뉴화면 제작비, 기타경비가 포함된다.

- 데이터베이스 설계비: 데이터베이스 설계공정 수행 시는 직접인건비, 제경비, 기술료 및 직접경비의 합에 의하여 용역비를 산정한다. 단, 제경비 및 기술료 산정은 현행기준 제10조와 같이 제경비는 직접인건비에 110%를 기술료는 직접인건비와 제경비의 합에 20%로 산정한다. 데이터베이스 설계비는 직접인건비로 아래와 같이 데이터베이스 설계의 공정별·기술등급별 소요공수와 규모별 투입인력을 계산하여 산정된 미디어별 기초인건비를 기준으로 데이터 제작 양과 보정계수인 규모별 보정계수, 작업난이도별 보정계수를 곱하여 산정한다.

$$\text{직접인건비} = \text{미디어별 기초인건비} \times \text{데이터 제작 양} \times \text{규모 보정계수} \times \text{작업난이도별 보정계수}$$

- 데이터제작비: 데이터베이스 구축에 쓰이는 데이터는 문자, 이미지, 그래픽, 애니메이션, 동영상, 음성·음악·음향으로 구분하며 데이터 제작공정은 직접인건비, 제경비 및 기술료를 포함한 총 경비를 일괄로 산정한다. 데이터제작비는 직접경비로 비용산정은 난이도 평가기준 및 등급판정 기준에 따라 품질을 단순, 보통, 복잡으로 구분하며, 난이도 등급에 따라 미디어별, 규모별로 산정한다.

$$\text{직접경비} = \text{난이도별, 미디어별 기초인건비} \times \text{데이터 제작 양}$$

- 원시자료수집비: 수집자료에 대한 저작권, 품질 등에 대한 기준이 모호하여 일률적인 단가를 정하여 대가를 산정하는 것이 불가능하므로, 각 사례별로 데이터 수집의 시장관행이나 경험에 따라 데이터베이스의 주요 메뉴별 구축 데이터 총량을 기준으로 계약한다.
- 메뉴화면제작비: 메뉴화면제작비는 기초화면 제작비와 안내정보 제작비로 별도 산정한다.

#### 나. 현행 대가기준의 문제점

현행 대가기준의 문제점을 요약하면 과도한 비용 산정, 산출기

준의 비 적정성, 산출과정의 복잡성, 산출기준의 모호성 등이다. 이러한 문제점을 구체적으로 살펴보면 다음과 같다.

- 데이터 제작 양이 증가할수록 데이터베이스 설계비 및 간접비가 기하급수적으로 상승
- 간접비가 직접비의 2배 이상 나오는 구조
- 제작 데이터 양이 데이터베이스 설계비 규모에 직접적인 영향 요소로 작용
- 복잡한 파라메타로 인하여 사용방법에 어려움
- 소프트웨어 설계에서 데이터베이스 설계를 포함하고 있으나, 데이터베이스 구축비용 산정시 적용에 어려움

또한 현재 가장 많은 데이터베이스 구축 비중인 문자 데이터의 경우 현행 데이터베이스 구축 대가기준의 데이터 제작 기준과 자료입력비 기준이 공존하고 있다. 때문에 기준을 적용하는 방법에 따라 두 가지의 비용이 산정 되는 모순을 가지고 있다.

#### 다. 사례분석

현행 데이터베이스 구축 대가기준에 대해 사례분석을 통하여 문제점을 파악하고 개선모델 방향을 제시한다.

데이터베이스 구축 사업대가 기준에서의 데이터 양의 변화에 따른 비교와 현행 데이터 제작의 경우 두 개의 기준인 데이터 제작과 자료입력비 기준에 대한 비교를 하였다.

데이터 양에 변화에 따른 비교를 보면 아래 표에서 데이터제작 규모가 커짐에 따라(25,000KByte → 50,000KByte → 75,000KByte) 직접인건비(데이터베이스 설계비)가 비례이상으로



증가하는 것을 볼 수 있다. 또한 직접인건비의 증가는 곧 제경비 및 기술료의 증가요인으로 전체적인 비용상승의 요인이 된다.

데이터 양이 비례로 증가할 때 데이터 제작비가 비례로 증가해야 하지만 설계비가 비례 이상으로 증가하는 것은 현실적이지 못하다. 따라서 설계비를 데이터 양에 의한 현행 비용산정 방법에는 문제가 있다.

<표 3.21> 사례1 (데이터 제작 양 비교)

(단위 :원)

DB 구축비	데이터제작 양			비교
	①25,000KByte	②50,000KByte	③75,000KByte	
직접 인건비	28,251,998	73,513,394	125,194,880	①② : ▲ 2 배 ①③ : ▲ 3 배
제경비	31,077,198	80,864,733	137,714,368	①② : ▲2.60배 ①③ : ▲4.33배
기술료	11,865,839	30,875,625	52,581,850	①② : ▲2.60배 ①③ : ▲4.33배
데이터 제작비	107,274,600	214,549,200	321,823,800	①② : ▲ 2 배 ①③ : ▲ 3 배
합 계	178,469,635	399,802,952	637,339,898	①② : ▲2.24배 ①③ : ▲3.57배

주1) 비교대상 : 문자데이터

주2) 비교조건 : 단순비교(동일 공정, 동일 난이도 등 모든 조건 동등)

주3) 2001소프트웨어 사업대가기준 해설 2001. 2 참조

데이터 제작비와 자료입력비 기준과의 비교를 하기 위하여 두 기준을 모두 적용할 수 있는 문자(한글) 데이터 입력으로 하였다. 데이터 제작의 7개의 공정 중 첫 번째 자료입력 한 공정만을 적용하였으며 난이도는 데이터 제작과 자료입력비 모두 보통으

로 하여 비교하였다.

비교 결과 데이터 제작비가 자료입력비에 비해 170% 높게 산정된다. 따라서 비용을 적용하는 기관 및 담당자에 따라 비용이 다르게 산정 되는 모순이 있는 것을 알 수 있다.

<표 3.22> 사례2 (데이터 제작비 및 자료입력비 비교)

(단위 :원)

	기준 비교		비교
	① 데이터 제작비	② 자료입력비	한글 50,000KByte
데이터제작비	52,134,950		
인건비		16,384,576	
제경비		11,469,203	
이윤		2,785,378	
합 계	52,134,950	30,639,157	①② : ▲1.70배

주1) 비교대상 : 문자데이터(한글)

주2) 비교조건

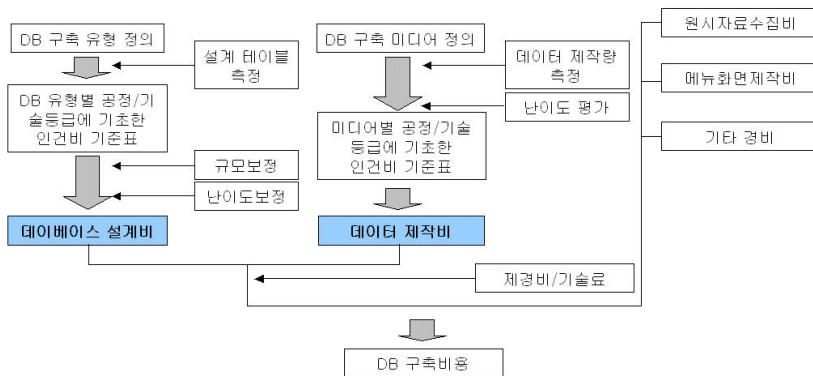
·데이터 제작 : 7개 공정 중 자료입력만 적용, 난이도 보통

·자료 입력비 : 스트로크기준 A4 27,500장, 난이도 보통

주3) 2001소프트웨어 사업대가기준 해설 2001. 2 참조

### 3.2.3 대가산정의 절차

개선된 데이터베이스 비용산정은 (그림 3.8)과 같이 데이터베이스 설계비와 데이터 제작비를 기초인건비로 하여 제경비 및 기술료 그리고 원시자료수집비, 메뉴화면제작비, 기타경비 등으로 산정된다.



(그림 3.8) 데이터베이스 구축 비용산정 모델

데이터베이스 설계비는 구축하는 데이터베이스 유형을 정의하고 설계될 데이터베이스의 테이블을 측정한다. 데이터베이스 설계 유형과 테이블 개수로 공정 및 기술 등급에 기초한 인건비 기준표에서 기초인건비를 산정하고 난이도보정계수를 곱하여 데이터베이스 설계비를 산정한다.

- 데이터베이스 구축 유형 정의
- 데이터베이스 설계 테이블 측정
- 데이터베이스 설계 기초인건비를 산정하고, 난이도 보정계수를 곱하여 데이터베이스 설계비를 산정한다.

데이터 제작비는 데이터베이스 구축하는 미디어를 정의하고 제작할 데이터의 양을 측정한다. 제작할 데이터별 난이도 평가기준과 가중치에 의해 난이도를 평가하고 공정 및 기술 등급에 기초한 인건비 기준표에 의해 기초인건비를 산정한다.

- 데이터 제작 미디어 정의
- 데이터 제작량 측정

- 데이터 제작 난이도 평가
- 미디어별 기초인건비를 산정한다.

산정된 데이터베이스 설계비와 데이터 제작비는 제경비와 기술료를 산정하는 지표가 된다. 제경비는 데이터베이스 설계비와 데이터 제작비에 최대 110%를 곱하여 산정할 수 있으며, 기술료는 데이터베이스 설계비에 제경비를 더한 값에 20%를 곱하여 산정한다. 또한 필요한 경우 원시자료수집비 및 메뉴화면제작비, 기타경비를 산정하여 데이터베이스 구축비용을 산정할 수 있다.

데이터베이스 설계비와 데이터제작비를 산정하고 제경비와 기술료를 협의하여 데이터베이스 구축비를 산정한다.

### 3.3 컨설팅 사업비

#### 3.3.1 개념 및 구조적 절차

컨설팅 사업비는 (그림 3.9)와 같이 사용자 요구에 따른 범위정의, 공수산정, 비용산정을 통해 산정된다. 범위정의에서는 컨설팅 업무의 가중치를 결정한다. 공수산정에서는 투입 컨설턴트의 등급을 결정하고 등급별 투입공수를 결정한다. 그리고 컨설턴트 대가를 구하여 가중치를 반영하고 직접경비를 합산하여 비용이 산정된다.

(그림 3.9) 컨설팅 대가 기준의 구조

〈표 3.23〉 컨설팅 대가 산정 모형

절차	방법	비고
범위 정의	·컨설팅 업무 가중치 결정	가중치 결정표 운용
공수 산정	·투입 컨설턴트 등급 결정 ·등급별 투입 공수 결정	별도의 컨설턴트 등급 및 대가 운용
비용 산정	·컨설턴트 대가 합계 ( ∑ 등급별 투입공수 × 등급별 대가) ·컨설팅비 계산 컨설턴트 대가 합계×가중치 + 직접경비	

컨설팅사업대가 산정 구조는 다음과 같이 범위 정의, 규모 산정, 비용 산정, 컨설팅비 계산의 4 단계로 이루어진다:

**1단계 범위정의** : 컨설팅업무 가중치 및 컨설팅 공정 결정

**2단계 규모산정** : 컨설팅 공정별 컨설턴트 등급 및 투입 공수 산정

**3단계 비용산정** : 비용산정 직접경비 산정

**4단계 컨설팅비 산정** : 컨설턴트 대가 × 가중치 + 직접경비

### 3.3.2 실행 절차

컨설팅 사업대가 산정 구조는 <표 3.24>같은 실행단계를 거친다.

<표 3.24> 소프트웨어 개발 사업대가 산정 절차

단계	방법	참조 표/템플리트
1. 범위 정의	① 컨설팅 업무 가중치 계산	컨설팅 업무 가중치 결정표
2. 규모 산정 (공수산정)	① 투입 컨설턴트 등급 결정	컨설턴트 등급별 자격요건
	② 투입 공수 산정	
3. 비용 산정	① 컨설턴트 대가 산정	컨설턴트 등급별 대가 산정
	② 컨설팅업무 가중치 곱함	
	③ 직접경비 산정	
	④ 총 비용 산정	

#### 가. 컨설팅 업무 가중치 결정

대상 업무의 특성 및 난이도에 따라 <표 3.25>의 가중치 결정표에 의해 가중치를 결정한다.

- 각 항목 별로 난이도에 따라 0%, 10%, 20% 중 하나를 택한다. 단, 각 항목별로 해당 사항이 없는 경우는 가산율을 0%로 한다.
- 각 항목별 적용 가산율을 더하여 가산율 합계를 구한다.
- 다음 식에 의거 가중치를 계산한다.

$$\text{가중치} = 1 + (\text{가산율 합계} / 100)$$

예1) 각 항목별 가산율 누계가 50% 일 때 : 1.5

$$\text{가중치} = 1 + 50/100 = 1.5$$

예2) 각 항목별 가산율 누계가 0% 일 때 : 1.0

$$\text{가중치} = 1 + 0/100 = 1.0$$

예3) 각 항목별 가산을 누계가 100% 일 때 : 2.0

$$\text{가중치} = 1 + 100/100 = 2.0$$

<표 3.25> 컨설팅 업무 가중치 결정표

항목	판단 척도						적용 가산율 (%)
	보통 가산율 : 0%		다소 어려움 가산율 : 10%		어려움 가산율 : 20%		
대상 업무의 예산 규모	50억원 미만	0%	50억 ~ 200억원	+10%	200억 이상	+20%	
타 업무와의 유착성	낮음	0%	보통	+10%	높음	+20%	
유사/참조 모델 유무	있음	0%	유사 모델 있음	+10%	없음	+20%	
사전 조사 필요 정도	낮음	0%	보통	+10%	높음	+20%	
신기술 요구 정도	낮음	0%	보통	+10%	높음	+20%	
업무의 지속성	지속적	0%	2회 이상 간헐적	+10%	일회성	+20%	
가중치 = 1 + (각 항목별 가산율(%)을 모두 더한 값) / 100							

#### 나. 규모 산정

- ① 해당 사업에서의 역할에 따라 투입할 컨설턴트를 결정하여 각 컨설턴트에 대해 <표 3.26>의 컨설턴트 등급별 자격요건에 따라 수석 컨설턴트, 책임컨설턴트, 전임컨설턴트, 컨설턴트, 보조컨설턴트의 6 가지 등급 중 해당 등급을 결정한다.
- ② 컨설턴트 등급별 투입 공수를 산정한다.



<표 3.26> 컨설턴트 등급별 자격요건

컨설턴트 등급	자격 요건	
	기술사/박사 급	기사/석사급 이상
수석 컨설턴트	해당분야 경력 10년 이상	해당분야 경력 14년 이상
책임 컨설턴트	해당분야 경력 6년 이상	해당분야 경력 10년 이상
전임 컨설턴트	해당분야 경력 2년 이상	해당분야 경력 6년 이상
컨설턴트	기술사/박사급 이상	해당분야 경력 4년 이상
보조 컨설턴트	학사급 이상으로서 해당분야 경력 1 년 이상	

- 단, 1. 박사급은 석사급 경력 4년 이상 및 학사급 경력 6년 이상과 등가되고, 석사급은 학사급 경력 2년 이상 및 전문학사급 경력 4년 이상과 등가된다.
2. 학사 취득 후 2년 경력자는 석사와 동등한 자격을 적용한다.
3. 전문학사 취득 후 2년 경력자는 학사와 동등한 자격을 적용한다
4. 산업기사 취득 후 2년 경력자는 기사와 동등한 자격을 적용한다
5. 그 밖에 명기되지 사항은 소프트웨어기술자등급별 자격요건에 의거하여 판단한다.
6. 특수분야의 경우 기술특성 등에 따라 계약 당사자의 상호 합의에 따라 자격요건을 일부 조정할 수 있다

#### 다. 비용 산정

##### ① 컨설턴트 대가 계산

- <표 3.27> 컨설턴트 대가를 이용하여 컨설턴트 등급별 컨설턴트 대가를 구한다
- 등급별 컨설턴트 대가 합계 = 등급별 총 투입공수 × 등급별 대가
- 등급별 컨설턴트대가 합계를 모두 더하여 컨설턴트대가 합계를 구한다.

&lt;표 3.27&gt; 컨설턴트 대가

컨설턴트 등급	월별 대가
수석 컨설턴트	20,651,000 원
책임 컨설턴트	16,521,000 원
전임 컨설턴트	12,391,000 원
컨설턴트	10,326,000 원
보조 컨설턴트	8,260,000 원

&lt;표 3.28&gt; 컨설턴트 대가 산정 예

#	투입 컨설턴트		투입 공수 (1)	컨설턴트 대가 (2)	(1) × (2)
	이름	컨설턴트 등급			
1	김갑철	수석 컨설턴트	6 인월	20,651,000원	123,906,000원
2	이을동	책임 컨설턴트	6 인월	16,521,000원	99,126,000원
3	박삼수	책임 컨설턴트	6 인월	16,521,000원	99,126,000원
4	안병옥	전임 컨설턴트	4 인월	12,391,000원	49,564,000원
5	정순희	전임 컨설턴트	4 인월	12,391,000원	49,564,000원
컨설턴트 대가 합계					<b>421,286,001원</b>

## ② 컨설팅업무 가중치를 곱해준다

- ①에서 구한 컨설턴트 대가 합계에 컨설팅업무 가중치를 곱한다: 등급별 컨설턴트 대가 합계 × 등급별 대가

③ 직접 경비 산정

- 컨설팅 사업 수행에 필요한 직접경비를 산정한다.

④ 총 비용 산정

- ②항의 결과에 직접 경비를 합산한 것이 컨설팅비가 된다.

**컨설팅 비용** = 컨설턴트 대가 합계 × 가중치 + 직접경비

## 제4장 예산수립시의 기능점수 활용

예산수립 단계에서 기능 점수를 활용하는데는 다음과 같은 애로사항이 있다:

- 기능점수 유형 구분의 어려움
- 기능점수 측정 경계선을 명확히 하기 어려움
- 기능점수 측정을 위한 사용자 요구사항 식별의 어려움
- 기능점수 측정에 필요한 정보의 부족
- 기능점수 산정결과와 실제 측정치와의 차이

이러한 예산수립 단계에서의 애로사항은 <표 4.1>과 같이 극복이 가능하다.

<표 4.1> 예산수립시의 애로사항 극복방안

기능점수 활용시의 애로사항	애로사항 극복방안
<ul style="list-style-type: none"> <li>• 기능점수 유형 구분의 어려움</li> </ul>	<ul style="list-style-type: none"> <li>• 조직의 IT 예산 효율화(ITA 활용) 차원의 검토를 방아 사업 계획을 수립한 후 예산 선정</li> </ul>
<ul style="list-style-type: none"> <li>• 기점수 측정 경계선을 명확히 하기 어려움</li> </ul>	<ul style="list-style-type: none"> <li>• 조직의 IT 전략과 비전에 따른 어플리케이션 포트폴리오 구성을 이해한 후 경계선 설정</li> </ul>
<ul style="list-style-type: none"> <li>• 기능점수 측정을 위한 사용자 요구사항 식별의 어려</li> </ul>	<ul style="list-style-type: none"> <li>• 실사용자를 찾아 가능한 구체적인 요구수렴</li> <li>• 유사 사례를 찾아 요구사항 벤치마킹</li> </ul>
<ul style="list-style-type: none"> <li>• 기능점수 측정에 필요한 정보의 부족</li> </ul>	<ul style="list-style-type: none"> <li>• 유사사례를 활용. 가능한 분야를 중심으로 측정한 후 이를 기반으로 추정</li> </ul>

<ul style="list-style-type: none"> <li>기능점수 선정결과와실제 측정치와의 차이</li> </ul>	<ul style="list-style-type: none"> <li>"Law of IS Cost Underestimating"(Gartner 2002)를 적용 산정치 2배를 곱하여 예산을 산정함</li> </ul>
---	--

기능점수를 활용하여 예산을 수립하기 위해서는 다음과 같은 사항을 고려해야 한다.

- 최대한 유사사례를 활용하여 산정한다.
- 불확실한 정보나 가정사항에 대해서는 철저한 문서화로 사후 변경에 대비한다.
- 산정에 필요한 최소한의 정보도 미흡 경우(예, 측정유형이 불투명, 경계가 모호, 요구사항이 상당수준 불확실 등)에는 측정을 보류하고 선행작업을 실시한다.
- 산정치를 확정적으로 제시하기 보다는 정확도 수준과 함께 범위 값으로 제시한다.
- 재산정이 필요한 경우를 구체적으로 명시하여 재산정의 가능성을 열어 놓는다.

정보화예산 수립 단계별 절차는 다음과 같다.

단계1 : 범위정의 - 예산 수립 대상 사업 도출  
단계2 : 규모산정 - 기능별 기능점수 도출  
단계3 : 규모보정  
- 불명확한 요구된 사항에 관련된 위험 반영 수준  
- 과거자료와의 비교 검토  
단계4 : 비용산정

- 기초인건비
- 직접인건비(보정계수 적용)
- 제경비/기술료/직접경비 등

예산 수립단계에서 기능점수를 활용하기 위해서는 기능점수 측정에 필요충분한 정보가 제공되기까지는 확정적 규모제시를 보류할 수 있는 여지가 필요(단계적 예산)하고, 정보화 예산수립의 합리화를 위한 제도 연구가 필요하다.

## 제5장 계약이후 발주기관의 점검 사항

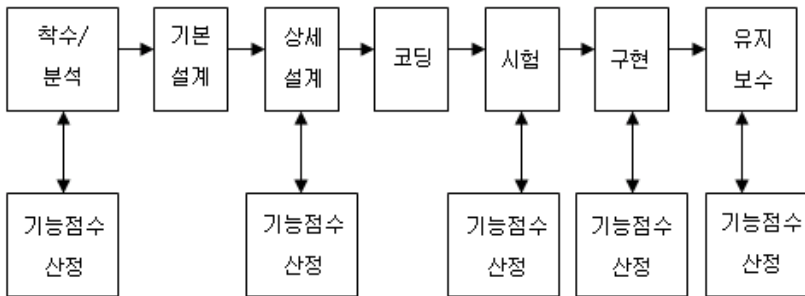
### 5.1 개발공정 단계

기능점수 산정 프로세스는 어플리케이션의 개발 단계에 따라 다를 수 있다. 개발의 초기단계에서는 산정시 참고할 만한 정보가 많지 않으며, 개발이 진행됨에 따라 더 많은 정보가 활용 가능해진다. 따라서 개발 초기에는 구두로 전달되는 사용자 요구사항이 그 대상이 될 수 있으며, 개발 단계에서는 다음과 같은 문서들이 기능점수 산정에 활용될 수 있다.

- 프로젝트 제안서
- 요구사항 명세서
- 상위 시스템 개념도
- 엔티티관계도(ERD)
- 기능 명세서
- 시스템 명세서
- 논리 데이터 모델
- 데이터베이스 레이아웃
- 프로세스 모델
- 프로그램/모듈 명세서
- 화면 설명서
- 파일 레이아웃
- 보고서 레이아웃
- 사용자 지침서
- 교육 교재

- 시스템 도움말

개발 프로젝트 기능점수 산정(Development Project Function Point Count)은 개발된 어플리케이션이 처음으로 설치될 때 최종사용자에게 제공하는 기능을 산정하는 것이다. 개발 프로젝트 기능점수 산정은 최초의 어플리케이션 기능점수수(Application Function Point Count)와 데이터 전환에 필요한 기능을 포함한다. 개발 기능점수는 개발 프로세스가 진행됨에 따라 변경되어야 한다. 이것은 이전에 파악된 기능들을 검증하고 새로 추가된 기능들을 파악하려는 것으로 보통 “범위 확장(scope creep)”때문에 필요하다. 기능점수 산정은 다음 그림과 같은 단계에서 수행할 수 있다.



(그림 5.1) 소프트웨어 개발공정에서의 기능점수 산정

유지보수 프로젝트 기능점수 산정은 기존의 응용에 대한 변경을 산정하는 것으로 새로운 기능의 추가, 기존 기능의 삭제, 기존 기능의 수정 등에 의해 사용자에게 제공되는 기능들을 합친 것이다. 전환 기능 또한 유지보수 프로젝트에 포함될 수 있다. 유지보수 프로젝트가 완료된 이후에는 어플리케이션의 기능에



있어서의 변경을 적절히 반영하여 어플리케이션 기능점수 (Application Function Point)를 변경해야 한다.

### 5.1.1 착수 단계

계약이후 처음 공정은 공정구현 단계로서 계약이후 15일 이내에 사업수행계획서(프로젝트계획서)를 제출하며 발주자는 프로젝트가 주어진 과업내용을 목표일정내에 품질을 확보할 수있도록 이의 타당성을 검토한다.

현행의 사업수행계획서에는 다음과 같은 항목이 포함된다.

<표 5.1> 기능점수 관련 사업수행계획서 고려사항

사업수행계획서 목차	기능점수 관련 고려사항
1. 사업명	
2. 사업기간	
3. 사업목적	
4. 사업범위	어플리케이션 경계 반영
5. 사업추진체계	최종사용자의 식별 기능점수 검증 담당 지정
6. 사업추진절차 및 접근방법	기능점수 합의 마일스톤 정의 변경관리 방안 및 기준 정의
7. 산출물계획	기능점수 관련 기존 산출물 조정 및 추가
8. 사업추진일정	기능점수산출시 활용된 WBS의 반영
9. 공정별 인력투입계획	기능점수, 개발생산성에 근거한 투입공수와 월별 투입계획
10. 품질관리계획	요구사항관리방안 추가 고려
11. 보고계획	주간/월간 보고시 기능점수 관련프로젝트 진척현황 항목 추가 필요
12. 위기관리 및 보안대책	전제조건 파악 및 대비책

13. 교육계획	기능점수 관련 교육대상(수/발주자모두) 및 일정
14. 주관기관 협조요청사항	

개발규모 및 비용산출 방식이 본수방식에서 기능점수 방식으로 전환될 경우 상기 사업수행계획서의 내용이나 양식에 일부 조정이 필요가 있다.

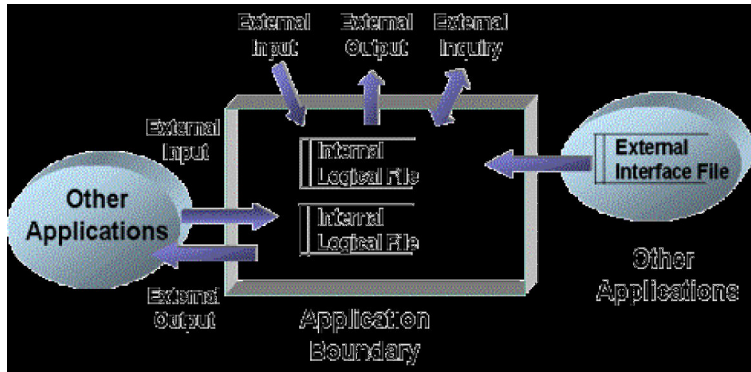
기본적으로 제안서 및 과업내용서의 산출내역서에 기능점수에 근거한 개발규모 및 비용이 제시되어 있으므로 이를 기초로 실제 수행가능한 일정과 인력, 공정편성, 산출물 등이 사업수행계획서로 작성되어야 한다. 작성된 기능점수 산출내역서에 대하여 검증하는 방법에 있어서는 3절의 “기능점수 모형에 대한 점검사항”을 참조하여 기능점수를 산출하는 과정과 지침, 근거자료에 대한 점검한다.

개발팀에서 제출한 사업수행계획서를 검토함에 있어 기능점수로 인한 추가적인 점검항목을 포함하여 다음 항목들이 점검될 필요가 있다.

- 제안요청서, 제안서, 과업내용서를 종합적으로 비교하여 최종 합의된 과업내용이 기능점수 모형의 업무 기능 및 데이터 기능으로 누락되지 않았는지 확인한다. 자체적으로 보유하거나 타 공공기관의 유사 사례가 있을 경우 과거 기록에 근거한 기능점수와 허용 오차범위(예, +/- 5%) 내에 있는지 확인하는 것이 용이한 방법이며 개발자가 제시한 통계자료도 참조할 수 있다.
- 개발대상 업무의 상세수준을 작업분할구조(WBS) 수준으로 상세화하고 명확하게 정의하였는지 작업분할 구조와 기능점수 목록을 비교하여 점검한다.
- 원시자료 구축에 대해서는 대상과 물량을 명확히 하여 데이터 전환

기능 점수에 반영되었는지 확인한다.

- 기능점수 산정시 산정 대상 프로젝트나 어플리케이션과 다른 외부의 어플리케이션 및 사용자 도메인을 구분하는 “어플리케이션 경계”를 확인한다.



(그림 5.2) 어플리케이션의 경계

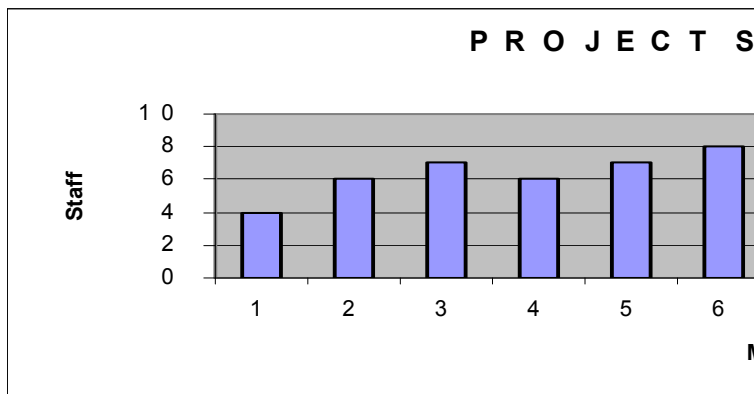
참조 : IFPUG의 어플리케이션 경계 구분 규칙

- 어플리케이션의 경계는 사용자의 관점에 기초해야 한다. 사용자는 어플리케이션의 범위를 정의할 수 있어야 한다.
- 관련된 어플리케이션들 사이의 경계는 기술적 구현이 아닌 업무 기능의 분리에 의해 구분되어야 한다.
- 어플리케이션의 경계는 어플리케이션이 유지보수됨에 따라 변경될 수 있다. 기능의 추가에 따라 경계가 확장될 수 있으며 삭제된 기능은 경계를 축소시킬 수 있다. 변경된 기능은 어플리케이션의 기능점수의 크기를 변경시킨다. 이 규칙은 유지보수 프로젝트의 범위가 어플리케이션의 경계를 결정짓는 것은 아니라는 것을 의미한다.
- 개발 프로젝트와 유지보수 프로젝트는 하나 이상의 어플리케이션을 포함할 수 있다. 이런 경우에는 모든 어플리케이션 경계들이 파악되고 기능점수도 별도로 산정되어야 한다.

- 기존 사업의 연속 사업으로 수행하는 경우 어플리케이션 경계를 명

확히 하여야 하며 기존 사업에서 개발된 응용 프로그램 및 DB를 연동하거나 재사용 범위에 대해서 사전에 파악하고 기능점수에 중복되지 않도록 반영하여야 한다.

- 타 시스템과 연동되어야 하는 경우 연동부분이 계획에 반영되었는지 확인한다.
- 제공될 기능(Functionality)의 최종 사용자가 누구인지 명확히 한다. 사용자의 대표성을 가지는 조직이 추진조직에 반영되어 요구사항을 제공/검증하고 최종 인수 책임자로서 역할이 명시되어야 한다. 이는 기능점수 산정시의 업무기능, 데이터 기능을 판단하는 기준이 된다.
- 예측된 개발규모에 따른 투입공수와 일정과의 연관성을 점검한다. 대개 개발 목표일정은 RFP에 의하여 부여되고 계약서에 명시된 일정은 특정한 사유없이 변경되지 않으므로 투입공수와 기간별 투입인력에 대하여 개발생산성 자료에 기초하여 점검한다.



(그림 5.3) 개발인력 투입계획

- 개발일정계획은 기능점수 목록을 산출하기 위한 작업분할구조(WBS)와의 일관성을 검토하여 불일치를 시정한다. 단계별 마일드스톤에서의 검토일정과 감리일정 등의 반영여부도 확인한다.
- 사업 추진상 전제조건이 있다면 이를 명시하고 이의 대비책이 구체화되어 정리되었는지 확인한다. 전제조건이 기능점수 산정의 정확성

에 영향을 미치는 요인인지 판단하여 기능점수 산정치 결과와 비교한다. 법/제도적인 요인이나 조직변경 등이 예상될 경우에는 사업일정이나 기능점수 예측에도 고려되어야 한다.

ERP, 전자결재시스템, KMS, CMS 등 다양한 패키지가 활용되고 있으며 패키지를 도입하여 커스터마이징을 수행하는 경우에는 기능점수 산정방식이 다르므로 일반적인 개발산출물을 적용하기 어렵고 일부 산출물은 불필요하므로 초기부터 산출물 조정작업이 필요하다.

패키지 기능		추가 기능
그대로 사용되는 패키지 요구기능 (그룹 1)	수정이 필요한 패키지 요구기능 (그룹 2)	추가 요구기능 (그룹 4)
불필요한 패키지 기능 (그룹 3)		

(그림 5.4) 패키지 커스터마이징 분야

사용자 요구사항에 따라 패키지의 기능중 그대로 사용하는 기능(그룹1)과 수정이 필요한 기능(그룹2)는 기능점수에 반영되어야 하며 필요하지 않은 기능(그룹3)은 제외되어야 한다. 이러한 내역을 정리한 산출물이 작성되어야 하며 “패키지 갭 보고서”라는 산출물이 작성되도록 사업수행계획서에 반영할 필요가 있다.

### 5.1.2 분석 및 설계 단계

분석 및 설계 단계는 ISO12207의 시스템 요구분석, 시스템 구

조설계, 소프트웨어요구분석, 소프트웨어구조설계, 소프트웨어상세설계 등을 포함하여 프로젝트 규모에 따라 세분화되기는 하나 일반적으로 분석단계와 설계단계로 나뉘어 단계별 승인 작업이 이루어진다.

사업초기 예산편성이나 제안, 계약시점에는 기능점수 산정방식이 간이모형에 의하여 이루어질 수 밖에 없으나 개발공정의 분석, 설계 단계에서는 요구사항을 확정하고 세부 기능분해 결과에 의한 복잡도 계산이 가능하므로 상세 모형으로 전환하여 산정할 수 있다. 적어도 설계단계 말에는 상세모형으로 산출한 기능점수수를 검증하고 확정함으로서 구현 및 전개 단계의 기준선으로 활용할 수 있게 된다.

기능점수 산정과 관련되는 분석 및 설계 단계의 산출물은 다음과 같이 기능점수 산출요소별로 나누어 볼 수 있으며 기능점수 관련하여 산출물의 내용을 일부 보완할 필요가 있다. 트랜잭션과 데이터의 상호 연관도를 표시하는 산출물은 일반적으로 공식 산출물이 아닌 경우가 있으나 기능점수 방식에 의한 개발규모 및 비용이 산출되었다면 이를 공식화할 필요가 있다.

〈표 5.2〉 기능점수 관련 분석/설계 산출물

기능점수 요소	분석 산출물	설계 산출물
어플리케이션 경계	업무배경도 (Context Diagram)	신규 업무흐름도
트랜잭션 기능	업무기능분해도 데이터흐름도(DFD) 프로세스정의서	화면 설명서 보고서/서식 설명서 프로그램사양서
데이터 기능	엔티티관계도(ERD) 엔티티정의서	논리 DB 다이어그램 테이블정의서
복잡도 산정		프로그램/테이블 매트릭스 (CRUD Matrix)
추가	인터페이스 정의서	인터페이스 설계서 분산설계 사용자접근권한설계

분석/설계 단계말 개발자가 제출한 분석/설계 산출물을 검토함에 있어 기능점수로 인한 추가적인 점검항목을 포함하여 다음 항목들이 점검될 필요가 있다.

- 과업내용서 및 사업수행계획서 작성시의 기능점수 예상치와 분석/설계 단계말 기능점수 산정치의 결과를 비교하여 변경여부를 파악하고 사용자 요구사항과의 추적성을 통하여 변경내역의 타당성을 검토한다.
- 요구사항정의서를 도출하기 위하여 실시한 면담계획 및 결과를 검토하여 초기 정의한 최종사용자(업무별, 직급별, 위치별 등)가 누락되지 않고 포함되어 있는지 확인한다.
- 요구사항 정의서의 내용중 시스템화 대상(업무기능, 데이터기능)이 식별되어야 하며 이중 시스템화 대상을 중심으로 기능점수에 반영

되었는지 비교한다.

- 현행시스템의 응용시스템과 데이터베이스 현황을 상세하게 파악하고 전환 대상을 결정하였는지 확인한다. 변환 대상 데이터에 대해서는 변환 매핑 자료가 작성되고 변환 프로그램이 개발되어야 하므로 데이터 전환기능 점수와 비교하여 일치여부를 점검한다. 구현단계의 변환절차서, 시험전략(접근방법), 신시스템 검증 작업에 영향을 준다.
- 분석단계의 프로세스 모델과 데이터 모델에 대하여 요구사항 반영 여부와 상호 비교 검증을 통한 내부 일관성을 검토하고 기능점수와 비교를 통한 누락여부를 점검한다.
- 초기 설정한 일정 및 투입공수 대비 실적을 비교하여 예측의 타당성을 검토하고 실적자료를 기초로 차후 단계의 일정 및 투입공수 예측에 반영되었는지 확인한다.
- 외부 시스템과의 연계가 있는 경우에는 내외부 인터페이스 정의서에 상호 교환정보 및 주기 등에 대한 내용이 명확히 파악되었는지 확인한다.
- 운영을 지원하는 프로세스가 설계되어 문서화되어 있는지 확인한다. 응용시스템을 운영하기 위한 기능은 일반 사용자로부터 파악하지 못하므로 운영자 측면에서 고려하여야 한다.
  - 입출력 제어를 위한 프로세스 설계
  - 오프라인 일괄처리 프로세스를 위한 작업 제어 설계
  - 백업, 재시도, 복구를 위한 프로세스 설계
  - 보안, 감사/통제를 위한 프로세스 설계
- 시스템 기반구조 설계시 지역적 분산을 고려한 분산분석과 설계가 수행되었는지 확인한다. 분산방식으로 설계가 결정된 경우 응용시스템 및 데이터베이스의 분산설계를 반영하고 있는지 점검한다.
- 데이터베이스 설계 과정에서 비정규화(Denormalization)작업이 수행된 경우 조정내역을 파악하여 데이터기능의 복잡도 산정시 RET(Record Element Type:레코드요소 유형) 파악에 적용되었는지 확인한다.



- 화면설명서/보고서설명서, 프로그램 사양서, 테이블정의서 등을 프로그램/테이블 매트릭스를 기초로 상호 추적검증하고 기능점수의 복잡도 산정에 반영되어 있는지 확인한다. 여기서는 전체 기능점수를 대상으로 하기 어려우므로 표본추출에 의하여 비교작업을 수행한다.

### 5.1.3 구현 및 설치 단계

구현 및 설치단계는 ISO 12207의 소프트웨어 코딩 및 시험, 소프트웨어 통합, 소프트웨어자격시험, 시스템 통합, 시스템 자격시험, 소프트웨어 설치, 소프트웨어 수락지원 등의 공정이 포함되며 실제 프로그램 코딩 및 데이터베이스 구현 작업을 수행하고 시험 및 설치, 인수 활동이 진행된다.

코딩 및 시험단계에서는 코딩이외에 시험전략의 수립과 시험모델의 구성 및 시험실시, 시험결과 정리 등이 핵심적인 활동이며 초기 과업내용, 사용자 요구사항을 검증하는 중요한 단계이다. 기능점수 방식의 도입에 따라 세부 기능요건에 따른 견적과 개발계획의 상세화가 가능하며 본 시험단계에서 이의 적합성을 확인하게 된다.

구현 단계말 개발자가 제출한 코딩 및 시험 산출물과 실제 구현된 응용시스템 및 데이터베이스를 검토함에 있어 기능점수로 인한 추가적인 점검항목을 포함하여 다음 항목들이 점검될 필요가 있다.

- 코딩 및 단위 테스트 결과가 상세설계 단계의 사양에 따라 구현되었는지 시험결과보고서를 점검하고 변동내역은 사유를 관리하여야 한다.
- 통합시험을 통하여 사용자 요구사항의 충족여부를 검증하고 품질목표의 달성여부를 확인한다. 통합시험 시나리오에는 기능점수 산정을

위한 트랜잭션 기능과 데이터 기능이 모두 포함되어 시험결과를 통한 과업내용 이행여부도 확인하도록 한다.

- 기존 시스템이나 업무에서 발생한 데이터를 신규 시스템으로 전환하는 작업이 계획대로 수행되었는지 확인하고 이 과정에서의 데이터 전환기능이 기능점수 목록과 비교하여 일치하는지 점검한다.
- 시험보고서에는 시험모델에 따른 테스트 케이스별로 시험결과를 정리하여 시험성공, 결함율등의 품질 자료가 포함되어 있는지 확인한다.
- 시험결과에 따라 변경요청 사항을 처리하는 절차가 있는지 확인하고 변경여부 및 재시험여부를 점검한다.
- 사용자 시험에는 초기 설정한 최종사용자(조직)가 참여하여 시험을 수행하였는지 확인한다.
- 단위시험 또는 통합시험 완료를 기준으로 형상통제를 위한 기준선(Baseline)을 설정하고 공식적인 변경요청 절차에 의하여 변경통제가 이루어지는지 확인한다.

최종 완료보고 단계에서는 개발자에 의하여 인수기준에 따른 검수 요청이 있으며 이에 대한 요청일자 포함하여 검수작업이 15일 이내에 실시되어야 한다. 검수작업은 발주기관의 검사담당자가 선정되어 수행되며 검사목록을 준비하여 이행여부를 확인하는 방식으로 진행된다.

- 검사 목록을 사전에 준비하였고 검사를 위한 기초자료가 확보되었는지 점검한다. 과업내용 및 사용자 요구사항의 이행여부를 점검하기 위하여 요구사항추적표를 제출하도록 하고 이를 기능점수 목록과 비교하여 확인한다.
  - 사업범위 및 사용자 요구사항의 점검 (범위 및 요구사항 적용 추적표 활용)
  - 산출물 작성 (사업수행계획서 정의내역) 및 제출점검

- 원시자료구축의 경우 건수 및 물량 등의 점검
- 장비 및 설비의 납품 내역 점검
- 사업내용의 변경 등으로 인하여 정산요인이 있는 경우, 다음과 같은 기준에 의해서 정산이 수행되었는가 확인한다. 기능점수 방식을 채택한 경우에는 변경내역을 기능점수로 환산하여 초기 설정된 단가를 적용하여 정산내역을 산출한다.
  - 계약서, 준공검사확인서, 및 변경요청목록을 근거로 정산하였는가
  - 개발용역비의 경우 산출물 변경, 직접경비의 사용, 품질보증활동에 따라 적절히 정산되었는가
  - 직접경비의 경우 계약금액에 근거하여 실비정산을 수행하였는가
- 정산결과 감액이 발생한 경우 감액금액 산정은 타당한가 확인한다.
  - 개발용역비는 소프트웨어사업비 대가기준에 따라 개발규모를 산정하는 데 적용한 방식 (개발본수 또는 기능점수)등을 근거로 감액 금액을 산정하였는가
  - 장비구입비는 낙찰율을 고려하여 감액 금액을 산정하였는가

운영 및 유지보수 단계를 위한 방안 및 절차서를 작성하고 인수인계 작업이 수행되어야 한다.

프로젝트 종료시 완료보고서를 제출하고 있으나 여기에 기능점수 관련 산출물을 별도로 정리하도록 하고 내부적인 기능점수 데이터베이스로 관리할 필요가 있다.

- 프로젝트정의서
- 트랜잭션기능정의서
- 데이터기능정의서
- 기능점수 산출내역서 (보정전/후)
- 개발공정 착수시의 예측자료

- 개발과정의 변경이력 (요구사항추적표)

## 5.2 프로젝트 관리 영역

프로젝트관리의 각 활동에 대한 검토항목은 PMBOK와 SW-CMM 의 "소프트웨어 프로젝트 계획수립"과 "소프트웨어 프로젝트 추적 및 감독"을 참조하고 “정보시스템감리기준”의 일반 관리 점검표를 활용하여 점검사항을 도출하였다.

### 5.2.1 범위관리

범위관리는 프로젝트를 성공적으로 완수하기 위해, 프로젝트의 모든 요구사항을 제대로 하는지를 보증하는 절차로서 어떤 일이 프로젝트에 포함되고 어떤 일이 프로젝트에 포함되지 않는지를 정의하고 조정하는 일이다. 범위관리 활동에는 범위계획, 범위정의, 범위 검증, 변경통제 등이 있으며 범위에는 프로젝트 범위(Project Scope)와 프로덕트 범위(Product Scope)가 있다.

범위관리는 기능점수 방식의 도입에 따라 보다 개선될 수 있는 프로젝트관리 활동으로서 기능점수수를 산정하는 과정에서 작성되는 문서를 활용할 수 있으며 보다 계량화된 관리가 가능하다.

<표 5.3> 범위관리 점검항목

- 프로젝트 범위는 사용자 요구사항을 반영하여 설정되었는가?
- 프로젝트 범위 및 주요 산출물에 대하여 사용자와 개발자가 합의하였는가?
- 프로젝트 범위를 관리하기 위한 범위관리계획이 수립되었는가?
- 작업분할구조(WBS : Work Breakdown Structure)는 소요자원, 비용, 일정을 산정하고 관리할 수 있도록 충분히 분할되었는가?
- 범위관리계획에 따라 범위가 변경되고 있는가?
- 범위 변경사항이 관련 문서에 반영되었는가?
- 완료된 프로젝트의 범위에 대하여 사용자측의 공식적인 승인이 있었는가?

범위를 정의하는데 사용되는 작업분할구조(WBS, Work Breakdown Structure)를 보다 구체적으로 작성함으로써 범위의 명확화와 함께 개발규모 예측, 개발일정 및 투입공수 산정의 기초자료로 활용될 수 있다.

범위는 작업분할구성도(WBS)와 주요 산출물로 나눌 수 있으며 정확한 범위 정의는 비용, 일정, 자원 산정의 정확도를 높이고 성과측정과 조정의 기준선이 되며 명백한 책임할당을 가능하게 한다.

작업분할구성도(WBS)는 서비스 및 기능분류 체계(Physical Breakdown Structure)와 개발활동 분류체계(Functional Breakdown Structure)로 구분되며 기본적으로 PBS와 FBS를 조합하여 단계별, 세부 업무별 작업분류를 구성하고 여러 시스템이 동일한 개발경로를 사용하는 경우 개발경로를 시스템별로 정의한다.

- 프로젝트에 관련된 모든 작업을 포함해야 한다.

- 프로젝트 계획단계부터 현장적용 및 업무전환 단계까지의 모든 작업을 분류하되 고객 및 사업자의 조직 및 여건을 고려한다.
- 작업분류 구조 및 요소작업은 관리 가능한 단위로 구분해야 한다.
- 작업분류별 적용기준, 적용대상, 적용범위 등을 명확히 정의해야 한다.
- 작업분류는 계층적으로 구분(PBS와 FBS의 조합으로 구분)해야 한다.

Change in Scope	FP Size	Additional Effort (staff-months)	Additional Cost (\$000)	Additional Schedule (calendar months)
New Vendor Function	100	7	100	2.0
Graphical Displays	20	1.2	20	.4
Banking System	20	1.2	20	.4
Mandatory Changes	10	.6	10	.2
Total	150	10	\$150	3.0

(그림 5.5) 변경요청 영향범위 파악

새로운 요구사항에 따라 작업 영역이 변경됨에 따라, CIO는 기능점수를 단위 작업 척도로 이용하여 사용자와 변경된 작업 영역의 정확한 특성과 이 변경에 관련된 비용에 관해 의사 소통할 수 있다. 범위가 변경되면 변경요청서에 이한 공식적인 변경 절차를 따라야 하며 범위 변경에 따른 일정, 공수, 비용, 품질 등의 영향을 파악하고 향후 계획에 반영하여야 한다.

### 5.2.2 일정관리

일정(공정)관리는 프로젝트를 정해진 기간 내에 완료할 수 있도록 작업을 분할하고 자원 및 비용을 할당하여 프로젝트를 합리적으로 관리하는 것으로 범위관리에서 도출된 WBS를 기초로 공정을 편성하고 일정계획을 수립한다.

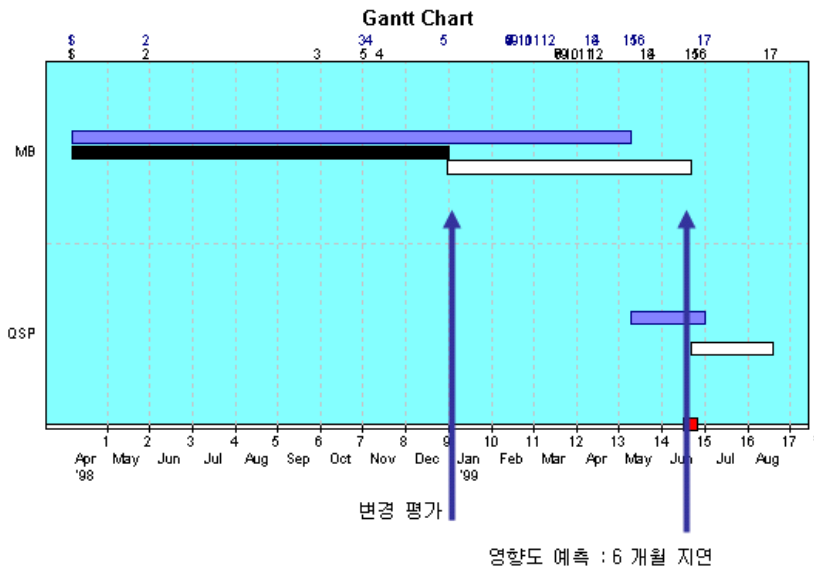
#### <표 5.4> 일정관리 점검항목

- 프로젝트를 수행하기 위하여 필요한 활동들이 모두 정의되었고,
- 정의된 활동들은 관리 가능한가?
- 주요 활동간의 의존관계가 파악되었고, 주요 활동에 대한 완료기준이 설정되었는가?
- 활동에 소요되는 기간은 활용가능한 자원을 고려하여 산정되었는가?
- 프로젝트는 시작일과 완료일이 명시된 활동계획에 따라 진행되었으며, 일정에 대한 진척도가 파악되고 있는가?
- 일정변경에 대한 영향을 분석하고 대책을 수립하여 관련자의
- 승인을 받았는가?
- 일정변경사항이 관련문서에 반영되었는가?

공정 편성시 분야별, 시스템별, 단계별로 전체 프로젝트 범위가 포함되어야 하며 정기적인 공정 진척현황 보고체계가 마련되어야 한다. 공정편성시에는 공정간의 선후행 관계가 정의되고 핵심공정을 파악하여야 하며 각 공정별 일정을 부여하는 경우에 개발생산성 및 투입가능 인력과 목표 일정과의 관계를 고려하여 편성한다. 필요시 특정 공정을 수행하기 위한 투입인력의 기술요건(skill)을 지정하고 해당 인력을 확보하거나 교육훈련을 통하여 충족시켜야 한다.

공정의 지연상황에 대하여는 지연사유, 타 공정에 미치는 영

향, 향후대책 등을 분석하여 공정 진행상황을 보고하도록 한다.



(그림 5.6) 변경요청에 따른 일정계획 반영

기능점수 측면에서 공정진척관리를 위하여 최소한 2주 단위의 진척보고 자료가 제출되어야 한다. 진척보고자료는 기본계획에 대한 차이분석(통계적 형태)에 활용되며 다음과 같은 내용으로 구성된다.

- 조직구성 : 프로젝트에 참여하는 인력
- 경과된 핵심 마일드스톤 : 프로그램 설계 완료, 코딩 완료 등
- 개발 진척단계 : 설계, 코딩, 단위시험, 통합시험, 검증
- 프로그램 모듈 규모 : 코딩완료시
- 총 비용 : 형상통제하에서현행화된 비용(공수)



- 소프트웨어 결함
- 통합 및 인수시험의 계획 및 실적

### 5.2.3 형상관리

형상관리는 프로젝트의 요구사항으로부터 최종 제품에 이르기까지의 형상을 관리하여 최종 제품이 고객의 요구사항을 정확히 반영하도록 하는 것이다.

일반적으로 형상항목으로는 개발산출물과 소프트웨어(프로그램 소스)를 대상으로 하고 있으며 일부 데이터베이스도 포함하는 경우가 있다. 이들 형상항목은 라이브러리에 등록하여 도구 또는 수작업에 의하여 관리된다. 기준선(Baseline)은 각 단계에서 공식적으로 발주자에 의해 검토되고 승인된 형상 항목들의 집합으로서 각 개발공정의 단계말이 되며 이 기준선에 따라 기능점수를 산정하여 발주자의 검토를 받는 것이 유용하다.

<표 5.5> 형상관리 점검항목

- |   |
|---|
| <ul style="list-style-type: none"> <li>• 형상항목과 버전 식별을 위한 기준 및 절차가 문서화되었으며 이에 따라 형상이 식별되고 기준선이 설정되었는가?</li> <li>• 형상항목에 대한 변경 및 통제가 체계적으로 관리되어 변경에 대한 추적이 가능한가?</li> <li>• 형상항목의 기능적이고 물리적인 완전성을 보장하기 위하여 형상평가가 수행되었는가?</li> </ul> |
|---|

형상관리에서 중요한 것은 프로젝트 추진과정에서 발생하는 변경을 합리적으로 관리하고자 하는 것으로 다음과 같은 목적을 가진다.

- 합리적인 변경을 통해서 시간과 자원의 효율적인 활용
- 무분별한 변경요청의 합리적인 통제
- 변경을 통한 프로젝트의 영향 분석 및 결과의 통제
- 변경이력을 통해서 계획 외 투입된 자원들의 집계자료로 활용

변경요청에 따른 형상변경관리 절차는 요청, 분석, 계획, 처리, 확인의 단계를 거치며 “변경요청 분석”시 변경으로 인한 프로젝트 범위, 일정, 자금에 미치는 영향을 분석한 후, 변경의 영향이 팀 내부로 한정되지 않고 프로젝트전체에 영향을 미치거나 여러 팀에 영향을 미치는 경우에는 변경영향분석서를 작성하고, 형상관리위원회를 소집하여 검토 및 승인을 받아야 한다. 변경에 따른 비용에의 영향은 기능점수 목록을 근거로 기능점수를 산정하고 초기 단가를 적용하여 산출할 수 있으며 추적할 수 있도록 근거자료를 관리하여야 한다.

## 변경영향 분석서

변경요청번호		제목			
현재 개발단계		담당SE		분석 일	
<i>프로젝트 범위에 미치는 영향</i>					
개요					
영향 받는 시작 개발단계					
변경규모	추가/삭제	대상 항목			
<i>프로젝트 일정에 미치는 영향</i>					
개요					
변경노력	추가/삭제	투입 Effort (Hours)			
변경 Staff	매월	추가/삭제	팀 규모	Total 시간 투입	
변경 기간	프로세스 또는 Task	시작일		종료일	
		이전	신규	이전	신규
<i>프로젝트 비용에 미치는 영향</i>					
개요					

(그림 5.7) 변경영향 분석서(예)

## 5.3 기능점수 모형에 대한 점검사항

### 5.3.1 비용 및 일정추정 관련 이슈

관리적 측면에서 소프트웨어 개발 프로젝트에 대하여 비용과 일정에 대한 추정치를 제시하였을 때 이의 신뢰성을 점검하고 프로젝트를 추진하여야 하며 신뢰도 점검과 관련하여 다음표와 같은 7가지 이슈를 제시할 수 있다.

<표 5.6> 7가지 비용 및 일정 추진 관련 이슈

1. 비용과 일정 추정의 목적이 명확하고 정확한가 ?
2. 태스크가 적절한 규모로 책정되었는가 ?
3. 예측 비용과 일정은 타 프로젝트의 예시된 성과와 일관성이 있는가 ?
4. 예측치에 영향을 주는 요인을 식별하여 설명하고 있는가 ?
5. 추정 프로세스의 무결성을 확보하도록 절차를 따르는가 ?
6. 조직의 과거 실적이 추정치를 신뢰할 수 있도록 뒷받침할 수 있는가 ?
7. 추정치를 준비한 이후 상황이 변화되고 있는가 ?

출처 : CMU/SEI-95-SR-004 A Manager's Checklist for Validating Software Cost and Schedule Estimate, Robert E. Park

이러한 이슈별로 세부 체크리스트를 가지고 점검함으로써 작성된 비용 및 일정 추정치에 의존하여 프로젝트를 추진할 것인가를 결정하는 참조자료로 활용할 수 있을 것이다. 이는 개발자나 감리인 등 프로젝트 이해관계자 모두 활용할 수 있는 체크리스트이며 착수단계부터 프로젝트 완료 단계의 전체 공정에서 지속적으로 참조할 수 있다.

### 5.3.2 이슈별 체크리스트

#### 이슈 1 : 비용과 일정 추정의 목적이 명확하고 정확한가 ?

##### 신뢰성 근거

- 추정의 목적을 명문화하고 있다
- 추정에서 적용하고 있는 생명주기가 명확히 정의되어 있다
- 추정에 포함되거나 제외된 태스크와 활동이 명확히 구별된다.
- 추정에 포함된 태스크와 활동이 추정의 목적과 일관성을 가진다.

#### 이슈 2 : 태스크가 적절한 규모로 책정되었는가 ?

##### 신뢰성 근거

- 소프트웨어 성과물의 규모를 추정하고 기술하는데 체계화된 절차가 사용되고 있다
- 재사용의 한도를 추정하고 기술하는데 체계화된 절차가 사용되고 있다
- 개발 규모와 재사용 한도를 추정하는 절차가 문서화되어 있다
- 개발 규모와 재사용 설명서는 적용된 규모 및 재사용 측정치에 무엇이 포함되었는지(제외되었는지) 구분한다.
- 재사용 측정치는 수정될 코드와 시스템에 그대로 통합될 코드를 분별한다
- 규모와 재사용을 설명하는데 사용된 정의, 측정치와 규칙은 비용과 일정을 추정하는데 사용된 모델의 요구사항(및 세부조정)과 일관성이 있다

- 규모 예측은 타 소프트웨어 제품 또는 구성품의 측정 규모와 연관되어 검증되었다.
- 규모 예측 프로세스는 완성된 제품의 측정 규모에 대비하여 예측 능력을 테스트함으로서 검증되었다.

### **이슈 3 : 예측 비용과 일정은 타 프로젝트의 예시된 성과와 일관성이 있는가?**

#### 신뢰성 근거

- 조직은 예측치를 완료된 작업의 실제 비용과 일정과 연계하기 위한 체계적인 절차를 가지고 있다.
  - 절차가 문서화되어 있다
  - 절차가 적용되었다
- 적용된 비용과 일정 모델은 관련 과거 데이터에 맞추어 조정되고 있다.
- ( 일부 모델은 이전 경험으로부터 조정되기 위한 일관된 규칙을 제공할 필요가 있다 )
- 비용과 일정 모델은 소프트웨어 제품과 프로젝트간의 차이에 대하여 정규화하는 방식으로 예시된 조직적 성과를 계량화한다.(단순하고 비정규화된 공수별 소스코드라인 조정은 추정을 위한 기초가 아니다 )
- 비용과 일정 모델을 과거 데이터에 맞출 때 달성된 일관성을 측정하여 보고하고 있다.
- 비용과 일정 모델의 파라미터에 적용된 값은 모델을 과거 프로젝트에 맞추는 값과 비교할 때 유효하게 보인다.
- 비용과 일정 모델의 조정은 추정치 준비에 사용된 모델과 동일한 버전으로 이루어 진다.

- 재사용을 계산하는데 사용된 방법은 무료가 아니다.(추정치는 효과적인 재사용과 관련된 인터페이스 설계, 수정, 통합, 테스트, 문서화와 같은 활동을 계산한다 )
- 과거 프로젝트로 부터의 조정은 응용기술의 차이를 고려한다.(예를 들면, 전통적인 메인프레임 응용을 구축하는 프로젝트의 데이터는 클라이언트-서버 구축을 추정하는 기초로 사용된다면 조정이 필요하다. 일부 비용 모델은 이에 대한 기능을 제공한다 )
- 과거 프로젝트로 부터의 세부조정은 소프트웨어 기술 개선에 있어 관측된 장기 추세를 고려한다.(일부 비용 모델은 내부적으로 시도하지만 최선의 방법은 조정된 조직의 성과에 보정되고 측정된 추세에 기초한다 )
- 과거 프로젝트로 부터의 조정은 새로운 소프트웨어 기술이나 프로세스 도입 효과를 고려한다.(신 기술이나 프로세스 도입은 초기에 조직의 생산성을 감소시킬 수 있다 )
- 워크 플로우 개요도는 프로젝트가 조직의 과거 성과를 특징지우는 데 사용된 프로젝트들과 유사한지(다른지) 평가하는데 사용되고 있다.

#### **이슈 4 : 예측치에 영향을 주는 요인을 식별하여 설명하고 있는가?**

##### 신뢰성 근거

- 가정을 식별하고 설명한다. 템플릿이나 포맷 같은 체계적인 프로세스가 핵심 요소를 간과하지 않도록 사용된다
- 파라미터 값의 불확실성을 식별하고 계량화한다.
- 위험분석을 실시하고 비용이나 일정에 영향을 주는 위험을 식별하고 문서화한다.(언급된 항목에는 발생가능성, 파라미터값에의 영향, 원가 파급효과, 스케줄 파급효과, 타 조직과의 연계 등 이슈를 포함한다 )

## 이슈 5 : 추정 프로세스의 무결성을 확보하도록 절차를 따르는가 ?

### 신뢰성 근거

- 관리자는 원가를 추정하기 전에 모든 서술식 파라미터 값을 검토하고 승인하였다.
- 원하는 원가나 스케줄을 충족하도록 파라미터 값을 조정한 내역을 문서화하고 있다
- 지정된 스케줄을 부여하고 있다면 추정치는 다음 추정치를 동반하고 있다.
  - 정상 스케줄
  - 지정 스케줄을 맞추기 위하여 필요한 추가 비용
- 원하는 원가와 일정을 맞추기 위한 파라미터 값의 조정은 값을 현실화하기 위한 관리 활동을 동반한다.
- 하나 이상의 원가 모델이나 추정 접근방법을 사용하고 있고 결과의 차이를 분석하여 설명하고 있다
- 연관성이 있으나 다른 프로젝트나 훈련에 관련된 인력을 추정 준비에 포함시켰다.
- 적어도 한명의 추정팀 멤버는 훈련된 추정전문가 이고 적용된 원가 모델 훈련을 받았다.
- 조직 운영에 독립적인 추정전문가가 파라미터 값과 추정 방법론의 합리성에 동의하고 있다.
- 작업을 수행할 팀이 추정치를 달성가능한 목표로 수용한다.
- 합의서를 비용과 일정에 영향을 주는 타 협력조직과 협의하고 사인하였다.



## 이슈 6 : 조직의 과거 실적이 추정치를 신뢰할 수 있도록 뒷받침할 수 있는가 ?

### 신뢰성 근거

- 추정 조직은 과거 프로젝트의 정보(이력 데이터베이스)를 체계화하여 유지하는 방법을 가지고 있다.
- 데이터베이스는 완료된 프로젝트에 대하여 유효한 일련의 자료를 보유하고 있다.
- 투입공수, 비용, 일정, 규모에 포함된(제외된) 요소와 재사용 요소를 데이터베이스에서 명확히 구분할 수 있다.(투입공수, 비용, 규모에 대한 SEI 체크리스트 참조 )
- 일정의 마일드스톤(시작과 종료일)은 착수, 종료 조건으로 기술되어 있어 마일드스톤 사이의 작업을 명확히 구분할 수 있다.
- 완료된 프로젝트에 대한 기록에는 미지급 오버타임이 사용되었는가를 알 수 있다.
- 미지급된 오버타임이 있다면 계량화하여 기록된 데이터가 향후 공수 예측에 유효한 기초자료를 제공한다.
- 추정에 사용된 비용 모델은 과거 데이터를 기록하는 프레임워크에 일관되게 적용된다. (모든 프로젝트에 비교가능한 조건과 파라미터가 사용되어 기록된 데이터가 추정 모델에 적합하다는 것을 보장한다 )
- 과거 데이터베이스의 데이터에 대하여 비일관성을 검증하고 불일치에 대하여 교정하고 설명한다.(추정에 적용된 동일 비용 모델에 가장 필요하다 )
- 조직 차원에서 진행중인 프로젝트의 공수와 비용 데이터를 수집하는 체계적인 프로세스를 가지고 있다.
- 개발 조직은 프로젝트 종료시 사후 검토를 한다
  - 기록된 데이터가 유효하도록

- 비용과 일정에 영향을 준 이벤트는 개발자가 생생히 기억하고 있을 때 기록하여 설명하도록
- 종료된 프로젝트에 대한 기록은 다음을 포함한다.
  - 적용된 생명주기 모델과 함께 기록된 비용과 일정이 포괄하는 부분
  - 실제(측정된) 규모, 비용 및 일정
  - 실제 참여 인력 프로파일
  - 종료시 추정치와 함께 실제 비용과 일정에 추정치를 매핑한 비용 모델의 파라미터 값
  - WBS(작업분할구조) 또는 기록된 비용에 포함된 태스크의 기술
  - 소프트웨어 프로세스를 도식화하는 워크 플로 체계
  - 비 인건비 비용
  - 관리 비용
  - 프로젝트에서 제공되는 중요 산출물 (소프트웨어 및 문서) 요약이나 목록
  - 비용이나 일정에 영향을 준 비 일상적인 이슈
- 조직의 워크 플로 체계 개선을 통하여 소프트웨어 프로세스의 이해와 측정에 발전이 있다.

## 이슈 7 : 추정치를 준비한 이후 상황이 변화되고 있는가 ?

### 신뢰성 근거

- 최근의 이벤트, 변경 요구사항, 관리 활동(관리부재) 등으로 추정치가 무효화되지 않는다.
- 자원을 할당하고 스케줄을 확정하며 승인을 얻는데 추정치가 기초로 사용되고 있다.
- 프로젝트 추적과 감독에 추정치가 실직적인 기준선이 되고 있다.

## 제6장 정보시스템 감리에서의 점검사항

### 6.1 정보시스템 감리의 영역

#### 6.1.1 정보시스템감리에의 영향

정보시스템 감리는 시점에 따라 개발감리와 운영감리로 나뉘며 감리목적에 따라 이행감리와 업무감리로 구분된다. 현재 대부분 시스템 개발에 대한 이행감리 위주로 실시되고 있다. 또한 감리 분야별로 구분하면 기술감리 위주로 시행되고 있으며 사전원가 계산이나 정산등의 비용감리도 일부 실시되고 있다. 기능점수의 도입으로 기술감리에서는 프로젝트 관리 분야에서 범위 및 일정관리 등을 강화하여 점검할 수 있을 것이다.

<표 6.1> 감리분야에 따른 감리 유형

구분	내 용
기술감리	<ul style="list-style-type: none"> <li>• 기술 측면에서의 정보기술 타당성 검토</li> <li>• 정보기술 자원의 품질 검토</li> <li>• 정보기술 자원의 내부통제 심사</li> </ul>
비용감리	<ul style="list-style-type: none"> <li>• 정보시스템 사업의 계약내용과 비용간의 타당성, 실행의 적정성 검토</li> <li>• 실시시기에 따라 사전원가계산과 정산으로 구분</li> </ul>
성과감리 (성과평가)	<ul style="list-style-type: none"> <li>• 기술감리 정보를 이용한 정보기술관리 측면에서의 평가</li> <li>• 정보기술 활용의 최종 성과를 평가</li> <li>• 사업의 투자에 대한 사회 ; 경제적 평가를 수행</li> <li>• 정책결정을 위한 신뢰성 있는 정보 제공</li> </ul>

감리 대상과 사업단계별 감리 유형으로 구분할 경우 사업감리(개발감리)와 사전감리에서 감리의 역할이 증가하게 된다. 특히 프로젝트 착수시점에 실시하는 사전감리에서는 프로젝트 편성의 타당성을 검증하기 위하여 과업내용서 및 사업수행계획서를 검토하는 과정에서 기능점수 산출 내역서를 참조할 수 있어 보다 상세한 점검이 가능하며 기능점수 산출내역 자체에 대한 감리도 수행할 수 있을 것이다.

<표 6.2> 감리대상과 사업단계에 따른 감리 유형

구분	내 용		
감리대상에 따른 구분	사업 감리	목적	개발사업의 성공적 수행을 목적으로 하며, 개발사업의 진행 단계에 따라 실시
		종류	<ul style="list-style-type: none"> <li>- 정보시스템 중장기계획 감리</li> <li>- 응용시스템 분석/설계 감리</li> <li>- 응용시스템 구현 감리</li> <li>- 시스템통합사업 감리</li> </ul>
	운영 감리	목적	컴퓨터시스템의 설비조직, 업무의 운영관리, 오류대책 등을 점검하는 감리로서 주기적 또는 특별한 사안 발생시에 실시
		종류	<ul style="list-style-type: none"> <li>- 시스템 통제준거성 감리</li> <li>- 시스템 안전성 감리</li> <li>- 시스템 효율성 감리</li> <li>- 시스템 효과성 감리 등</li> </ul>
사업단계에 따른 구분	사전 감리	신규사업이 착수되기 이전에 계획상의 요구사항 정의, 시스템의 기본구조, 개발방침, 세부작업 정의, 일정 ; 조직 ; 인력 ; 적용기술 및 기기, 예산 등을 검토	
	진행 감리	시스템 개발과정에서 진행의 적정성을 확인	
	사후 감리	종료된 사업에 대하여 요구사항의 충족도, 투입비용과 기간의 적정성, 품질 및 성능, 사용자 교육, 문서화	

		등을 평가
--	--	-------

기능점수 모형의 도입에 따라 개발감리를 수행함에 있어 감리 절차나 방식에 있어서 크게 변화되지는 않을 것이나 프로젝트 관리 분야의 감리가 강화될 수 있는 기회가 된다. 단계말 감리시 개발팀에서 산출한 기능점수의 결과에 대하여 검증하는 감리가 추가될 수 있을 것이며 사전 비용산출 및 정산 등을 포함하는 비용감리를 독립적인 감리인이 수행하는 것이 활성화될 수 있다.

<표 6.3> 감리 분야별 영향

감리 분야	영향도	비 고
프로젝트관리	○	범위, 일정관리
표준 및 품질보증활동	○	형상관리
응용시스템	△	요구사항 추적
데이터베이스	△	데이터 기능
시스템 아키텍처 및 보안	-	
시험계획 및 활동	△	시험사례와 기능점수

### 6.1.2 정보시스템 개발감리에의 고려사항

개발공정에서 감리인의 점검항목은 “정보시스템 감리기준”에 제시되어 있고 세부적인 감리지침을 활용하고 있으며 기능점수의 도입으로 인한 점검항목은 발주자의 점검 포인트와 유사하다. 각 단계별 감리시 산출물과 응용시스템, 데이터베이스의 준거성

과 실증 실사를 수행하는 감리의 측면에서는 기능점수와의 관계를 고려하여 비교 검증하는 작업이 추가될 것이다.

<표 6.4> 개발산출물의 추가 검토사항

개발 산출물	추가 점검 항목	비 고
사업수행계획서	과업범위, 일정편성 내역이 기능점수를 산출하기 위한 어플리케이션 경계, WBS 등과 비교하여 일치/누락 여부를 검증	
요구사항정의서	요구사항 추적표에 의하여 기능과 데이터 요건으로 연계여부 확인	최종사용자와의 확인여부 점검
프로세스정의서	트랜잭션기능목록과 비교하여 일관성여부 점검	
엔티티정의서	데이터기능 목록과 비교하여 일관성여부 점검	
화면설명서 보고서설명서	트랜잭션 기능의 복잡도 산출 타당성을 표본 검증	상세모형적용
인터페이스설계서	외부 시스템과의 입출력 기능에 대하여 트랜잭션 기능을 점검	
테이블설계서	데이터기능의 복잡도 산출 타당성을 표본 검증	상세모형
변환매핑	데이터 변환기능과의 비교	
프로그램사양서	화면, 보고서 이외의 배치 프로그램, 공통기능에 대한 기능점수 확인	
프로그램/테이블 매트릭스	트랜잭션기능의 복잡도 산정시 프로그램 사양서와 함께 비교하여 활용	
시험사례	기능점수와의 비교를 통한 충분성 확인	

## 6.2 기능점수(Function Point)에 대한 감리

### 6.2.1 기능점수 감리의 개요

기능점수 방식의 도입이 확산되고 보다 중요한 의사결정 요소가 되면 이에 대한 독립적 감리(감사)의 필요성이 발생하게 된다. 여기서 감리란 정량화된 정보와 설정된 기준간의 연관 정도를 결정하여 보고하기 위하여 자격있고 독립적인 감리인이 증거를 축적하고 평가하는 절차이다. 감리의 독립성은 필수적인 요소이나 동일 조직내에서도 독립적인 부서에 위치하면 감리를 수행할 수 있다.

기능점수와 관련된 기준으로 IFPUG 4.1 산정지침이 기본적인 기준이 될 수 있으며 소프트웨어사업대가기준(해설서)도 가능하다. 이러한 지침이나 기준에의 적합성을 점검하는 것을 준거성 감리(Compliance Audit)라고 한다.

감리증거(Evidence)은 기능점수가 기준을 준수하고 있는가를 결정하기 위하여 감리인이 사용하는 정보라고 할 수 있으며 기능점수 자체, 시스템 문서, 개발자/사용자 면담, 실제 기능점수를 산정한 담당자 인터뷰 등 다양한 형태를 가진다. 실제 감리를 수행함에 있어서는 기능점수 전체를 전수검사를 할 수 없기 때문에 표본 추출을 실시하며 예상 오류의 수와 기능점수 산정절차의 효과성에 대한 감리인의 기대에 따라 표본의 수가 달라지게 된다.

감리증거는 감리에 적합해야 하며 추가적인 테스트(실증)나 검토 감리영역을 찾기 위해서는 감리인의 기술이 필요하다. 감리증거는 신뢰할 수 있어야 하며 요청한 자료중 제출되지 않은 증거는 최종 감리보고서에 명시하여야 한다.

모든 감리증거는 다음에 기초하여 평가하여야 한다.

- 가치평가(Valuation) : 기능점수가 당연히 포함해야할 항목을 산정하고 포함되지 말아야 할 트랜잭션이나 파일을 산정하지 않아야 한다.
- 완전성(Completeness) : 모든 트랜잭션이나 파일이 최종 기능점수에 포함되어야 한다.
- 분류(Classification) : 모든 트랜잭션이나 파일이 적절하게 분류되어야 한다. 외부 입력이나 외부 인터페이스 파일이 정확하게 분류되었는가를 확인하는 것은 매우 중요한 일이다.
- 등급(Rating) : 트랜잭션이나 파일이 적절하게 상,중,하로 등급이 부여되어야 한다. 이를 위해서는 데이터 요소나 파일참조를 세부적으로 점검하여야 한다.
- 기계적인 정확성(Mechanical Accuracy) : 문서간에 계산이나 참조자료를 표본 점검하는 것이다. 산술적 정확성이라고 할 수 있으며 자동화도구가 사용되지 않은 경우에 중요하다.
- 분석적 검증(Analytical Analysis) : EI, EO, EQ, ILF, EIF 등의 비율이나 GSC 를 유사한 타 응용에 비교하여 유효성을 검증하는 것이다. 감리 초기에 실시하게 되면 보다 집중적으로 점검할 영역을 찾는 데 도움이 된다.
- 감리절차의 최종 단계를 보고서로서 발견사항을 사용자와 의사소통하는 수단이다. 보통 3가지 유형으로 나뉘며 표준 부적합, 조건부 의견, 반대 의견 등이 있다

## 6.2.2 기능점수 결과에 대한 감사 절차

기능점수 산정결과를 감사하는 20 단계의 절차(Procedure)가 있으며 상세하게 문서화되지 않더라도 감리를 수행할 수 있는 지침으로 자체적인 지침을 개발하거나 기능점수를 감리하는데 기여한다.



1. 기능 점수를 산정하는 태스크가 전체 프로젝트 계획에 포함되어 있는가 ?

프로젝트 팀이 수행하는 모든 활동은 프로젝트 계획의 항목이 되어야 하며 태스크를 수행할 수 있는 적절한 시간이 할당되어야 한다.

2. 기능점수를 산정하는 사람은 훈련을 받았는가? 자격을 가지고 있는가 ?

공식적인 훈련과정이 필요한 것은 아니나 IFPUG 4.1 산정규칙에 익숙해져야 한다. IFPUG 인증시험을 통과한 사람이 산정하는 것이 바람직하며 정확한 산정을 보장하지는 않지만 최소한의 능력을 확보하는 것이다.

3. IFPUG 4.1 산정 실전매뉴얼을 따르고 있는가 ?

4. 기능점수 산정시 현행화된 프로젝트 산출물을 사용하였는가 ? 그렇지 않다면 얼마나 오래된 문서를 참조한 것인가 ?

5. 프로젝트 팀이 기능점수 산정에 참여하였는가 ?

프로젝트 팀은 사용자에게 제공되는 기능에 대하여 가장 정통한 사람이며 프로젝트에 대한 최상의 정보원이다. 기능점수 산정과정에 프로젝트 팀이 배제되고 산정 담당자가 일부 문서에 의하여 독자적으로 기능점수를 산정할 경우에는 프로젝트 팀이 기능점수의 정확성에 대하여 이의를 제기하게 된다.

6. 내부적으로 작성된 기능점수 산정지침을 따르고 있는가 ?

7. 사용자 관점에서 응용 시스템을 산정하였는가 ?

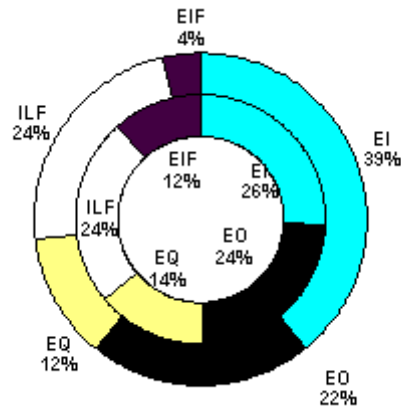
8. 물리적 관점이 아닌 논리적인 관점에서 시스템을 산정하였는가 ?

9. 기능점수에서 설정된 경계는 타 메트릭(일정 보고, 결함추적 등)의 경계와 일치하는가 ? 그렇지 않다면 이유는 ?

10. 기능점수 산정이 보강을 위한 것이라면 경계는 응용시스템의 경계와 같은가 ? 그렇지 않다면 이유는 ?

11. 경계가 변경되었는가 ? 그렇다면 이유는 ?

12. 기능점수 산정에 도구가 사용되었는가 또는 수작업으로 산정하였는가 ?
13. 수작업으로 산정하였다면 산술적 검증이 필요하게 된다.
14. 개별적인 기능점수 요소(ILF, EIF, EI, EO, EQ)의 비율이 산업평균과 일치하는가 ? 그렇지 않다면 타당한 이유는 ? 여러 응용시스템을 감리하면 트랜잭션과 파일의 비율은 유사하다.



(그림 6.1) 기능점수 요소별 기여도 분석

15. 프로젝트팀이 트랜잭션(EI, EO, EQ)과 파일(ILF, EIF)의 저장소를 검토하였는가 ? 기능점수 산정에서 가장 큰 오류는 누락시키는 오류이다. 응용팀이 기능점수의 완전성과 정확성을 점검하는 것이 중요하다.
16. 총 보정요소(Value Adjustment Factors)는 타 프로젝트와 일치하는가 ? 총 보정요소는 검토된 응용시스템의 평균 보정요소와  $\pm 5$  퍼센터 내에 존재한다. 이 범위 밖에 존재하면 기능점수 산정결과에 문서로 설명되어야 한다. 예를들면 평균 VAF가 1.05이며 VAF는 1.0과 1.1 사이에 존재해야 한다.
17. 14개의 일반 시스템 특성(General System Characteristics)은 타 프로젝트의 범주내에 존재하는가? 각 GSC는 평균 GSC의 1점 내

에 존재하는가 ? 예를들어 특정 GSC가 2.0으로 판정되었다면 GSC는 1,2,3중의 하나가 될 것이다. 이 범위 밖에 존재하면 기능점수 산정결과에 문서로 설명되어야 한다.

18. 기능점수 산정과 관련된 모든 전제조건(Assumptions)은 문서화되었는가? 전제조건은 문서화되어 필요시 향후 추가적으로 검토될 수 있다.
19. 이러한 전제조건은 타 프로젝트와 일관성이 있는가 ?
20. 기능점수 산정에 영향을 준 전제조건은 중앙의 기능점수 산정그룹에 보고되었는가 ? 모든 전제조건은 중앙 그룹에 의하여 검토되어야 한다.
21. 산정결과는 독립된 공인 기능점수 전문가(Certified Function Point Specialist)가 검토하였는가 ?

## 참 고 문 헌

- [1] 한국전산원, 알기 쉬운 공공정보화 추진방법, 정보화정책 이슈 02-정책-02, 2002. 5
- [2] 한국전산원, 공공부문 정보화사업 관리 방법론, 2001. 9
- [3] 한국전산원, 소프트웨어 사업대가기준 교육교재 개발, NCA IV-RER-02044, 2002. 12
- [4] 한국소프트웨어산업협회, 소프트웨어대가기준
- [5] Robert E. Park, CMU/SEI-95-SR-004 “A Manager's Checklist for Validating Software Cost and Schedule Estimate”
- [6] Longstreet David, “Auditing Function Point Counts”, The Voice, A Publication of the International Function Point Users Group, 1996,
- [7] Jean-Marc Desharnais & Pam Morris, “Post Measurement Validation Procedure for Function Point Counts”, 1996
- [8] “Software Purchasing Using The SEI-Core Measures”, QSM

## 기능점수방식으로의 전환지침(안)

---

2002년 12월 인쇄

2002년 12월 발행

발행인 : 서삼영

등록번호 :

발행처 : 한국전산원

서울시 중구 무교동 77번지 NCA B/D

Tel. (02) 2131-0114

인쇄처 : 대도문화사

Tel. (02) 2273-5496

---