

논문 2016-1-2

# 개발비 감정 시 SW사업 대가산정 가이드 적용 문제점

권기태\*

## Problem of SW Cost Estimation Guideline for the Appraisal of Software Development Cost

Ki-Tae Kwon\*

### 요 약

컴퓨터 프로그램 개발비용 산정 감정은 소프트웨어 개발에 있어서 그 공정상의 개발비용 등을 소프트웨어 공학적인 측면에서 판단하여 산정하는 것으로 되어 있다. 컴퓨터 프로그램 완성도 감정 가이드라인은 완성도 감정업무의 지침서로 활용할 수 있으며, 컴퓨터 프로그램 완성도 및 개발비 감정을 위한 절차와 방법을 제공하는 것이다. 소프트웨어 개발비 산정의 세부 내용은 “SW사업 대가산정 가이드”를 참조하도록 되어 있다. 다수의 개발비 산정 감정에서 성공적으로 감정이 이루어져 왔으나, 최근의 몇몇 개발비 감정에서 감정 결과에 대한 논의를 요하는 사례가 발견된 바, 본 논문은 개발비 감정 시 “SW사업 대가산정 가이드”를 적용할 경우에 발생하는 문제점을 분석하고자 한다.

### Abstract

The development cost appraisal of the computer program has been assumed that the appraiser to determine such as development costs on the process from the software engineering aspects. “Completeness appraisal guidelines of a computer program” can be used as a guideline for complete appraisal business, is to provide the procedures and methods for the development costs and the degree of completion of computer programs. For more information about the software development costs calculation, we can refer to the “SW cost estimation guide”. Although successful appraisal of a number of development costs based on the guide has been processed, but a number of cases requiring discussion of appraisal results have been discovered. In this paper, we try to analyze the problems that occur when you apply the “SW cost estimation guide”.

**한글키워드** : 개발비 감정, 기능 점수, 완성도 감정

**keywords** : development cost, cost evaluation, function score, completeness evaluation

### I. 서 론

컴퓨터 프로그램 저작물 감정 유형에 따른 분류에 의하면 소프트웨어 감정은 유사(동일도) 감정, 완성(하자)도 감정, 개발비용 산정 감정으로 구분된다. 이 중에서 개발비용 산정 감정은 소프트웨어 개발에 있어서 그 공정상의 개발비용 등

\* 강릉원주대학교 컴퓨터공학과 교수  
(email: ktkwon@gwnu.ac.kr)

접수일자: 2016.5.13. 심사완료: 2016.5.30.

게재확정: 2016.6.19.

을 소프트웨어 공학적인 측면에서 판단하여 산정하는 것으로 되어 있다[1]. 이러한 개발비 산정 감정은 분쟁 대상의 프로그램을 개발하는데 따르는 소요비용이 어느 정도인지 판단하는 것이 목적이다[2].

한국저작권위원회에서는 컴퓨터 프로그램 완성도 감정업무의 지침서로 활용할 수 있도록 “프로그램 완성도 감정 가이드라인”을 발표한바 있다[3]. 완성도 감정 가이드라인에서는 프로그램 완성도 감정 추진 단계별로 수행내용을 제시하여 컴퓨터 프로그램 완성도 감정인의 업무에 적용하고, 컴퓨터 프로그램 완성도 감정 시 필요한 각종 산출물 및 완성도 감정 사례들을 제공하여 담당자들이 참고할 수 있도록 하며, 특히 감정인 애로사항을 최대한 검토 및 반영하여 해결 방안을 모색하는데 주안점을 둔다. 최종적으로 완성도 감정 가이드라인은 컴퓨터 프로그램 완성도(하자) 및 개발비 감정을 위한 절차와 방법을 제공하는 것이다[4].

완성도 감정 가이드라인에서는 소프트웨어 개발비 산정을 크게 기능점수 방식과 투입공수에 의한 방식으로 구분하고 있으며, 세부 내용은 한국소프트웨어산업협회의 “SW사업 대가산정 가이드[5]”를 참조하도록 되어 있다. 기능점수 방식은 소프트웨어 개발 규모를 기능점수로 산정한 후, 전체 기능점수에 기능점수당 단가와 보정계수를 곱하는 방식이다. 구체적인 기능점수 계산 방식은 한국소프트웨어산업협회의 “SW사업 대가산정 가이드” 또는 한국소프트웨어산업협회의 “국제표준기반 기능점수 산정 안내서[6]”를 참조하도록 되어 있다. 투입공수에 의한 방식은 과거의 유사 소프트웨어 개발 사업의 투입인력 정도를 기초로 한 경험적 판단에 의해 사업대가를 산정하는 방식으로, 기능점수 방식의 적용이 곤란한 소프트웨어 유형에 한하여 제한적으로 적용하도록 되어 있다.

다수의 개발비 산정 감정에서 한국소프트웨어 산업협회의 “SW사업 대가산정 가이드”에 의해 성공적으로 감정이 이루어져 왔으나, 최근의 몇몇 개발비 감정에서 감정 결과에 대한 논의를 요하는 사례가 발견된 바, 본 논문은 개발비 감정 시 “SW사업 대가산정 가이드”를 적용할 경우에 발생하는 문제점을 분석하고자 한다. 본 논문은 장. 서론에 이어서 2장에서 SW사업 대가산정 가이드의 주요 내용을 정리한 후에, 3장의 국제표준 기능점수 산정과정과 비교한 사례별 문제점을 4장에서 정리한다. 5장에서 결론 및 추후 연구과제를 기술한다.

## 2. SW사업 대가산정 가이드

### 2.1 대가산정 가이드의 목적 및 배경

SW사업 대가산정 가이드는 국가·지방자치단체·국가 또는 지방자치단체가 투자하거나 출연한 법인 또는 기타 공공단체 등(이하 “국가기관 등”이라 한다)에서 소프트웨어의 기획, 구현, 운영 등 수명주기 전체 단계에 대한 사업을 추진함에 있어 이에 대한 예산수립, 사업발주, 계약 시 적정대가를 산정하기 위한 기준을 제공하는 것을 목적으로 하고 있다[5].

대가산정 활동은 SW사업 전체 수명주기 동안 반복적으로 수행되는 활동으로 발주자나 수주자를 비롯한 다양한 이해관계자들에게 큰 영향을 미치는 중요한 활동이다.

가이드를 통하여, 국가 정보화사업의 합리적이고 객관적인 대가산정을 유도하여 국내 SW사업의 품질을 향상시키고 제값주기 환경을 지속 정착시켜 SW산업의 경쟁력을 제고하는 효과를 거둘 수 있을 것으로 기대하고 있다[5].

기존의 공공부문 SW사업 대가산정은 「소프트

트웨어사업 대가의 기준」과 「엔지니어링사업대가의 기준」을 활용하여 왔다. 그러나 2010년 2월 26일 고시된 「소프트웨어사업 대가의 기준」(지식경제부 고시 제2010-52호) 부칙 제4조(소프트웨어사업 대가의 기준 재검토)에 의거하여 “정부는 소프트웨어사업에 적용되는 사업대가가 민간 자율로 결정되도록 유도하기 위하여 동 기준을 시행일로부터 2년이 되는 시점에 폐지한다.”라고 고시됨에 따라 SW사업대가의 기준은 2012년 2월 26일 이후 더 이상 적용될 수 없게 되었다. 이에, 한국소프트웨어산업협회는 소프트웨어산업진흥법 26조에 의거하여, 국가기관 등에서 SW사업 대가산정 시 준용할 수 있도록 「SW사업 대가산정 가이드」를 대체방안으로 마련하였다[5].

가이드는 정적인 고시체제에서는 담지 못한 SW산업의 동적인 상황과 글로벌 표준에 입각한 ISO12207 기반의 소프트웨어의 수명주기(기획, 구현, 유지관리·운영) 전반에 걸쳐 대가산정 방법을 알기 쉽게 설명하고, 보다 편리하게 수행할 수 있는 도구 제공과 사용자의 편의와 이해 제고를 목적으로 하여 개발하였다[5].

또한, 공공부문 SW사업의 대가산정 기준으로 활용하기 위한 근거를 「행정기관 및 공공기관 정보시스템 구축·운영 지침」(안전행정부)과 국가의 「예산 편성·집행 지침」(기획재정부)에 명시하였다[5].

## 2.2 소프트웨어 개발비 산정 절차

소프트웨어 개발비는 소프트웨어 개발의 단계별 공정을 수행하는데 필요한 개발원가와 직접경비 그리고 이윤의 합으로 구성토록 되어 있다.

소프트웨어 개발비는 그림 1과 같이 다음 절차에 따라 산정한다.

1) 개발할 소프트웨어의 규모를 기능점수(Function Point, FP)로 측정한다. 기능점수는 국

제표준화기구(ISO/IEC 20926)에서 정의한 소프트웨어 규모의 측정 단위로 그 값은 수치값으로 표현된다.

2) 위에서 산정된 기능점수에 각각 기능점수당 단가를 곱하여 보정전 개발원가를 구한다.

3) 위에서 도출된 보정전 개발원가는 개발하고자 하는 소프트웨어의 규모, 어플리케이션의 복잡성 등과 같은 각 사업별로 발생할 수 있는 특성적 요인들을 고려하지 않은 개발비용이다. 본 기준에서는 이러한 요인들을 크게 소프트웨어 개발 규모, 어플리케이션 유형, 개발언어, 품질 및 특성으로 구분하였으며 이에 대한 보정계수를 적용하여 개발원가를 구하도록 하고 있다.

4) 위에서 구한 개발원가에는 소프트웨어 개발 사업에서 일반적으로 소요되는 직접경비 항목은 포함되지 않는다. 따라서 이러한 직접경비 항목을 별도로 고려할 수 있도록 하고 있다.

5) 이윤은 개발원가의 0~25% 범위 내에서 반영할 수 있도록 하고 있다.

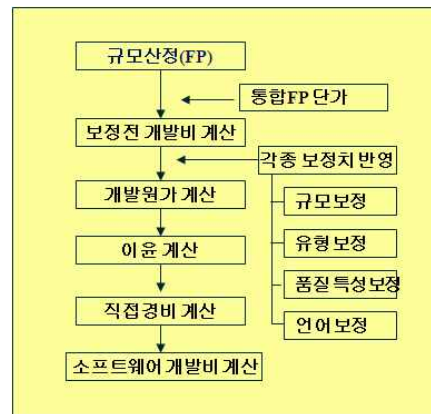


그림 1. 개발비 산정 과정  
Fig. 1. Development cost estimate process

소프트웨어 규모산정을 위한 가장 보편적인 척도인 기능점수(FP)는 대상 소프트웨어가 명확하게 정의되고, 외부 시스템과의 인터페이스가

분명한 상태에서 그림 2와 같이 구성된다.



그림 2 기능점수 구성 요소

Fig. 2. Configuration elements of function score

기능점수는 그림 2와 같은 5가지 유형의 기능을 추출하여 각각의 복잡도를 측정한다. 각 기능의 복잡도는 단순, 보통, 복잡의 3 단계로 구분하여 평가한다. 복잡도에 주어지는 가중치는 기능 유형별로 상이하다.

### 3. 국제 표준 기능점수와의 비교

#### 3.1 국제표준 기능점수 산정 프로세스

국제표준화기구(ISO/IEC 20926) 및 기능점수를 관리하는 국제기구인 IFPUG에서 정의하는 기능점수의 산정 절차는 그림 3과 같다.

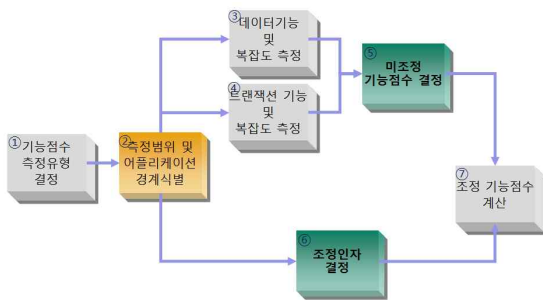


그림 3. 국제 표준 기능점수 산정 프로세스  
Fig. 3. Function cost estimate by global index

기능점수 산정 프로세스의 첫 번째 단계는 기능점수 측정 유형을 결정하는 것이다. 측정 유형은 개발 기능점수, 개선 기능점수, 어플리케이션 기능점수와 같이 3가지로 구분할 수 있다[7]. 개발 프로젝트 기능점수값은 데이터 변환에 필요한 기능은 물론, 최초의 어플리케이션 기능점수 계산 과정에서 계산될 기능들을 포함한다. 만약 인사관리 어플리케이션을 새로 개발하는 어플리케이션으로 교체하려면 새로운 어플리케이션에 의해 제공되는 기능을 계산해야 한다. 또한 기존 데이터 파일에 있는 데이터들을 새로운 데이터 파일로 변환하기 위해 사용자가 요구하는 변환 기능도 계산해야 한다. 개발 프로세스가 진행됨에 따라 개발 프로젝트 기능점수값은 자주 갱신되어야 한다. 왜냐하면 이 값은 원점에서 시작하여 계산한 것이 아니므로 이미 식별된 기능들을 재검토하고, 일명 “범위 변경(scope creep)”에 따라 추가된 기능들을 포착하려는 것이다. 기능점수 계산은 그림 4와 같이 여러 단계에서 수행할 수 있다.

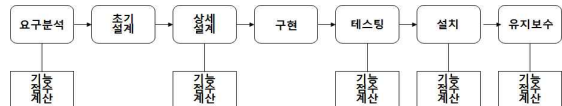


그림 4. 소프트웨어 생명주기와 기능점수  
Fig. 4. Function score and life period

개선 기능점수는 기존 어플리케이션에 대한 변경을 측정하는 것으로 새로운 기능의 추가, 기존 기능의 삭제, 기존 기능의 변경에 의해 사용자에게 제공되는 통합된 기능을 포함한다. 개선 프로젝트에도 변환 기능이 존재할 수 있다. 개선 작업이 끝나면 어플리케이션의 기능 변경을 적절히 반영하기 위해 어플리케이션 기능점수값이 변경되어야 한다.

어플리케이션 기능점수는 개발 혹은 개선 후에 설치된 어플리케이션이 규모를 측정하는 것이

다. 이것은 기준선 기능점수 또는 설치된 기능점수라고도 불리며 어플리케이션이 현 시점에서 최종 사용자에게 제공하는 기능을 나타낸다. 각 활동별 어플리케이션 기능점수값을 합하면 설치되어 현재 이용되거나 개선된 모든 어플리케이션의 기능점수값이 계산된다.

### 3.2 SW 대가산정 가이드의 산정 프로세스

한국소프트웨어산업협회의 “SW사업 대가산정 가이드”의 기능점수 산정 절차를 그림 3과 비교하면 그림 5와 같이 나타낼 수 있다.

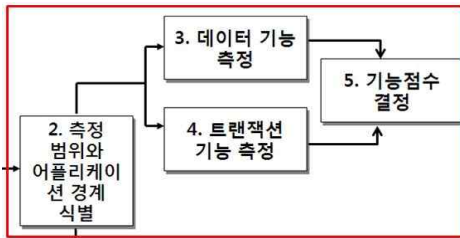


그림 5. SW 대가산정 가이드와의 비교  
Fig. 5. SW development cost guide

그림 3의 국제표준 기능점수 산정 프로세스와 비교하면 한국소프트웨어산업협회의 “SW사업 대가산정 가이드”의 산정 절차는 그림 5와 같이 표준 프로세스의 일부로 구성된다. 즉, “SW사업 대가산정 가이드”에서는 개선 기능점수와 어플리케이션 기능점수는 적용되지 않으며 오직 개발 프로젝트의 기능점수만 정의된다.

표 1. 간이법 기능점수 산정 양식  
Table 1. Simple function score estimate

기능유형	가중치		합계
	평균복잡도		
내부논리파일	( )	× 7.5	
외부연계파일	( )	× 6.4	
외부입력	( )	× 4.0	
외부출력	( )	× 5.2	
외부조회	( )	× 3.9	
총 기능점수			

### 3.3 간이법 적용을 위한 평균 복잡도

한국소프트웨어산업협회의 “SW사업 대가산정 가이드”에서는 각 기능요소별 복잡도를 표 1의 간이법 기능점수 산정 방식과 표 2의 정통법 기능점수 산정 방식을 제시하고 있다.

표 2. 정통법 기능점수 산정 양식  
Table 2. Function score estimate standard

기능유형	가중치			합계
	낮음	보통	높음	
내부논리파일	( ) × 7	( ) × 10	( ) × 15	
외부연계파일	( ) × 5	( ) × 7	( ) × 10	
외부입력	( ) × 3	( ) × 4	( ) × 6	
외부출력	( ) × 4	( ) × 5	( ) × 7	
외부조회	( ) × 3	( ) × 4	( ) × 6	
총 기능점수				

표 2의 정통법 기능점수 산정 방식은 국제표준의 기능점수 방식을 따르고 있다. 그러나 일반적으로 기능별 복잡도를 판별하기 어려운 경우(발주 시)에는 표 1의 평균복잡도를 적용한 간이법을 사용하도록 되어 있다. 평균복잡도는 국제표준의 기능점수 산정 방식에는 제공되지 않는 것이다. 이것은 국내 소프트웨어사업대가기준이 본수 방식과 스텝수 방식에서 객관적이고 검증 가능한 국제표준의 기능점수를 도입하려는 시점에서 국내 소프트웨어 사업의 특성 상 수발주자의 기능점수 관련 전문기술의 부족으로 인해 기능점수 도입의 저항이 크기 때문에 일시적으로 도입할 의도였으나, 한국소프트웨어산업협회의 “SW사업 대가산정 가이드”에도 계속 유지되고 있다.

### 3.4 보정계수와 조정인자(VAF)

“SW사업 대가산정 가이드”의 개발비 산정 구조는 다음과 같다.

$$\begin{aligned}
 & \text{개발비} = \text{개발원가} + \text{이윤} + \text{직접경비} \\
 & \text{개발원가} = \text{기능점수} \times \text{단가} \times \text{보정계수} \\
 & \text{이윤} = \text{개발원가} \times 25\% \text{이내} \\
 & \text{직접경비}
 \end{aligned}$$

보정된 개발원가의 보정에 이용되는 보정계수는 어플리케이션 유형 보정계수, 언어 보정계수, 규모 보정계수, 품질 및 특성 보정계수로 구성된다. 어플리케이션 유형 보정계수는 업무처리용, 과학기술용, 멀티미디어용, 지능정보용, 시스템용, 통신제어용, 공정제어용, 지휘통제용 등의 유형에 따라 1.0부터 2.2 사이의 적절한 보정계수를 선택한다.

언어 보정계수는 발주자가 특정 언어의 사용을 요구할 경우에만, 개발과 직접적으로 관련이 있는 구현 및 시험 단계에만 언어별로 0.6에서 1.9 사이의 적절한 보정계수를 적용한다.

다음은 규모 보정계수로, 산정된 기능점수를 아래의 식에 대입한다.

$$\text{규모 보정계수} = 0.108 \times \log_e(\text{기능점수}) + 0.2229$$

품질 및 특성 보정계수를 구하기 위해서는 먼저 영향도 판단 기준에 따른 각 항목별 영향도를 구한다. 영향도의 합(총 영향도)을 다음 식에 대입하여 품질 및 특성 보정계수를 구한다.

$$\text{품질 및 특성 보정계수} = 0.025 \times \text{총영향도} + 1.0$$

한국소프트웨어산업협회의 “SW사업 대가산정 가이드”의 보정계수는 상기 기술한 보정계수를 모두 곱해서 개발원가를 보정한다.

반면에 국제표준의 기능점수는 그림 3의 6단계에서 구한 조정 인자(Value Adjustment Factor: VAF)를 이용하여 그림 3의 최종 단계에서 기능점수 자체를 조정하게 되어 있다. 14개의 일반 시스템 특성(GSC)은 조정 인자(Value Adjustment Factor: VAF)로 요약된다. VAF는 최종 조정된 기능점수값을 산출하기 위해 미조정 기능점수값을 +/- 35% 보정한다[8].

대체적으로, 단순한 배치 어플리케이션은 총점수(총 영향도인 TDI)가 15 이하, 프론트-엔드

배치 어플리케이션은 15에서 30사이, 대화식 어플리케이션은 30에서 45사이, 실시간, 통신 또는 프로세스 제어 시스템은 30에서 60사이로 예상할 수 있다. 조정 인자(VAF)를 계산하는 절차는 다음과 같다.

1) 14개의 GSC를 각각의 지침에 따라 각 GSC의 영향도를 결정하기 위해 0에서 5까지의 점수로 평가한다.

2) 14개 GSC의 영향도를 모두 더해 총 영향도(TDI)를 계산한다.

3) 다음 공식에 TDI를 대입하여 조정 인자(VAF)를 계산한다.

$$\text{VAF} = (\text{TDI} \times 0.01) + 0.65$$

## 4. 개발비 감정 사례별 문제점

### 4.1 사례 1: 기능점수 측정유형의 문제점

3.1절에서 기술한 것과 같이 한국소프트웨어산업협회의 “SW사업 대가산정 가이드”의 기능점수 측정 유형은 개발 기능점수로 한정되어 있다. 그러나 다양한 감정 요청 사례들을 보면 순수한 개발보다는 개선 사업, 유지보수 사업, 재개발 사업, 기존 시스템의 커스터마이징 사업 등 광범위한 영역을 아우르고 있다.

기존 감정 사례에서 소프트웨어사업 대가산정 기준에 의한 감정이 원심에서는 인정되었으나, 대법원에서 인정받지 못한 바 있다. 대법원 판결에 의하면, “소프트웨어사업 대가산정 기준”에 비추어보면 기능점수 방식의 개발비 산정은 개발 대상 소프트웨어가 어느 정도 고유성과 독자성을 가지고 있는 경우에 적합한 방식이라는 것이다. 즉 국제표준 기능점수 산정 프로세스 첫 단계의 기능점수 측정 유형 중 개발 기능점수라는 의미

이다. 그러나 감정 대상 소프트웨어의 경우, 이미 별도의 계약으로 제3자에게 의뢰하여 개발한 소프트웨어를 포함하고 있으며, 개발자가 이러한 방식으로 기존에 개발한 프로그램을 기초로 일부 수정을 거쳐 개발한 사실이 객관적인 증거에 의해 명백한 반면에 개발비 감정은 감정 대상 프로그램 전체를 원점에서 새로 개발하는 것을 전제로 하는 “소프트웨어사업 대가산정 기준”을 그대로 적용하였기 때문에 적정 개발비를 산정하지 못했다는 것이다.

그 외에도 부수적으로 규모 보정계수의 오 적용 및 이윤의 포함 등을 원심 판결 파기의 이유로 들고 있으나, 본질적으로는 “소프트웨어사업 대가산정 기준”의 기능점수 측정 유형인 개발 기능점수가 감정 대상에는 해당하지 않는다는 의미이다.

감정 요청서에는 대법원 판결에서 명시한 바와 같이 감정 대상 프로그램이 별도의 계약으로 제3자에게 의뢰하여 개발한 소프트웨어를 포함하고 있으며, 개발자가 이러한 방식으로 기존에 개발한 프로그램을 기초로 일부 수정을 거쳐 개발한 사실에 관련한 자료가 제시되지 않아 원점에서 새로 개발한 프로그램으로 가정하고 감정이 진행된 것이다.

본 감정 사례와 같이 감정 대상 프로그램 전체가 원점에서 개발되는 것이 아닌 경우에 한국소프트웨어산업협회의 “SW사업 대가산정 가이드”를 그대로 적용하기에는 여러 측면에서 제한이 따른다. 다양한 유형의 개발비 감정 사례에 적용하기 위한 방안을 찾아야 할 것이다.

#### 4.2 사례 2: 기능점수 당 단가와 복잡도

“SW사업 대가산정 가이드”의 기능점수 당 단가는 개발 기능점수를 전제로 유도된 기능점수 당 단가이다. 또한 이 단가는 국내 주요 대규모

SI 업체들의 기 프로젝트 자료를 통해 유도된 대형 사업의 기능점수 당 평균 단가이며, 이 기능점수 당 단가에는 그림 6과 같이 다양한 요소가 포함되어 있다.

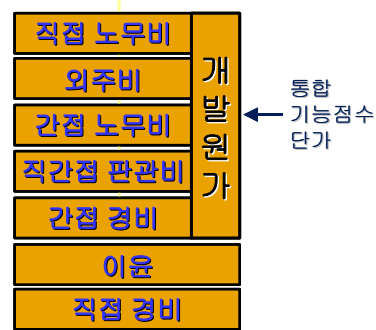


그림 6. 기능점수 당 단가 구조  
Fig. 6. Price structure by function cost

감정 대상 프로그램의 경우에는 대개 평균 단가가 유도되었던 대규모 SI 프로젝트와는 성격이 다른 경우가 일반적이다. 따라서 “SW사업 대가산정 가이드”의 기능점수 당 단가를 성격과 유형이 전혀 다른 결과는 사례에 적용할 경우, 개발비가 지나치게 과대 추정될 가능성이 존재한다.

개발비의 과대 추정을 막고, 감정 대상 프로그램에 적합한 개발비를 산정하기 위한 방안이 필요하다.

“SW사업 대가산정 가이드”에서는 기능점수 산정 시 적용되는 복잡도를 표 1과 표 2처럼 평균 복잡도와 국제표준 기능점수의 복잡도를 제시하고 있다. 실제 감정 사례에서는 이 복잡도 외에도 감정인 임의로 별도의 복잡도를 적용하려는 시도가 있을 수 있다. 물론 상기 기술한 바와 같이 기능점수 유형이 다른 경우에 “SW사업 대가산정 가이드”를 보완하여 적용하려는 시도에서 복잡도를 임의로 수정하여 적용할 수는 있으나 구체적인 근거를 제시하여야 할 것이다.



4.3 사례 3: 대가산정 가이드 자체의 오류

기능점수 산정을 위한 표준 문서 및 매뉴얼은 IFPUG에서 발간된 CPM[9]이며 국내 번역서[10]가 존재한다. “SW사업 대가산정 가이드”에서는 상기 서적의 내용대로 기능점수 측정 방법을 기술하고 있으나, 지면의 한계 및 전문지식의 부족으로 인해 모호하게 기술된 부분이 존재한다.

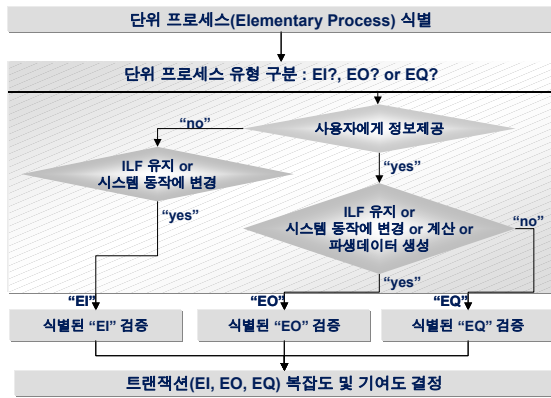


그림 7. 트랜잭션 기능점수 식별 과정  
Fig. 7. Transaction score verification process

예를 들어, 트랜잭션 기능점수를 정확하게 도출하기 위해서는 그림 7과 같은 과정을 거친다. 단위 프로세스는 트랜잭션 기능요소를 구분하는 기준이며, 각 유형은 단위 프로세스의 주요 의도(primary intent)에 의해 구분된다.

그러나 “SW사업 대가산정 가이드”에서는 단위 프로세스의 유형을 구분하는 결정적인 역할을 단위 프로세스의 주요 의도에 대한 기술이 전혀 언급되어있지 않아, 트랜잭션 기능 유형에 대한 오류가 발생할 가능성이 높다. 또한 “SW사업 대가산정 가이드”에서는 표 3과 같은 양식을 제공하고 있다. 표의 기술 의도는 단위 프로세스와 연관된 데이터 기능점수를 식별하는데 도움을 주기 위한 의도였을 것으로 짐작되나, 관련된 설명

없이 수차례 이와 같은 표를 제시하고 있어, 기능점수에 관한 전문 지식이 없는 경우 단위 프로세스 별로 데이터 기능점수를 산정해야 하는 것으로 오해할 우려가 있으며, 실제로 감정 과정에서 발생하는 심각한 문제점 중의 하나이다.

표 3. 데이터 기능점수 산정 예  
Table 3. Example of function score estimate

서브시스템	단위프로세스	내부논리 파일	외부연계 파일	구분
SNS 확산형 홈페이지	국민토론 정보	1		신규
	국민제안 정보	1		신규
	설문 정보	1		신규
	외부기관 콘텐츠 정보		15	신규
소 계		3	15	

5. 결론

프로그램 개발비 산정 감정은 소프트웨어 개발에 있어서 소프트웨어 개발 공정상의 비용을 소프트웨어 공학 측면에서 판단하여 산정하는 것이다. 이러한 개발비 산정 감정은 감정 대상의 프로그램을 개발하는데 따르는 비용이 어느 정도인지 판단하는 것이 기본적인 목적이다.

한국저작권위원회에서는 프로그램 완성도 감정업무의 지침서로 활용할 수 있도록 “프로그램 완성도 감정 가이드라인”을 발표하였다. 완성도 감정 가이드라인에서는 프로그램 완성도 감정 추진 단계별로 수행내용을 제시하여 컴퓨터 프로그램 완성도 감정인의 업무에 적용하고, 컴퓨터 프로그램 완성도 감정 시 필요한 각종 산출물 및 완성도 감정 사례들을 제공하여 담당자들이 참고할 수 있도록 한다. 궁극적으로 완성도 감정 가이드라인은 컴퓨터 프로그램 완성도(하자) 및 개발비 감정을 위한 절차와 방법을 제공하기 위한 것이다.

완성도 감정 가이드라인에서는 소프트웨어 개



발비 산정을 크게 기능점수 방식과 투입공수에 의한 방식으로 구분하고 있으며, 세부 내용은 한국소프트웨어산업협회의 “SW사업 대가산정 가이드”를 참조하도록 되어 있다. 기능점수 방식은 소프트웨어 개발 규모를 기능점수로 산정한 후, 전체 기능점수에 기능점수당 단가와 보정계수를 곱하는 방식이다. 구체적인 기능점수 계산 방식은 한국소프트웨어산업협회의 “SW사업 대가산정 가이드”를 참조하도록 되어 있다.

다수의 개발비 산정 감정에서 한국소프트웨어산업협회의 “SW사업 대가산정 가이드”를 적용하여 성공적으로 감정이 실시되었으나, “SW사업 대가산정 가이드”가 가지는 본질적인 문제점에서 기인하는 감정의 한계를 벗어나기 위해, 본 논문은 개발비 감정 시 “SW사업 대가산정 가이드”를 적용할 경우에 발생하는 문제점을 대표적인 사례별로 분석하였다. 추후 연구에서 본 논문에서 분석한 사례별 문제점의 대안을 제시하고자 한다.

### 참고 문헌

[1] 한국저작권위원회, “저작권 관련 감정사건 판례집(1)”, 2010.12.  
 [2] 한국저작권위원회, “저작권 관련 감정사건 판례집(2)”, 2013.07.  
 [3] 한국저작권위원회, “프로그램 완성도 감정 운영 가이드라인”, 2015.11.  
 [4] 권기태, “컴퓨터 프로그램 완성도 감정 개요”, 한국 소프트웨어 감정평가학회 논문지, 11권 2호, 2015.12.  
 [5] 한국소프트웨어산업협회, “SW사업대가산정 가이드(2015년 개정판)”, 2015.05.  
 [6] 한국소프트웨어산업협회, “국제표준 기능점수 산정 안내서”, 2014.11.  
 [7] 권기태, “기능점수 분석 방법론”, 동아인쇄출판, 2015.  
 [8] 권기태 외 공역, “기능점수와 소프트웨어 측정”, 도서출판 그린, 2004.

[9] IFPUG, “Counting Practice Manual”, Release 4.2.1 International Function Point Group, 2008.  
 [10] 권기태 외 공역, “기능점수 측정 실무 매뉴얼”, 한국정보화측정진흥원, 2010.

### 저자 소개



권기태(Ki-Tae Kwon)

1986년 서울대학교 계산통계학과 졸업  
 1988년 서울대학교 계산통계학과 석사 졸업  
 1993년 서울대학교 계산통계학과 박사 졸업  
 1996년 미국 Univ. of Southern California, 전산학과 Post-Doc.  
 현재 강릉원주대학교 컴퓨터공학과 교수

<주관심분야 : 소프트웨어공학, 데이터마이닝, 지능시스템>