

# 기능점수(Function Point)산정 및 활용 방안

발주지원센터 | 송기호(khsong@software.or.kr)

국내 정보통신 분야가 급속하게 발전함에 따라 다양한 형태의 소프트웨어사업들이 국가 정보화사업을 주축으로 용역발주 되고 있다. 또한, 최근 수년 동안 정보시스템 구축과 운영 사업의 증가로 국가예산에서 정보화예산이 차지하는 비중이 커짐에 따라 정보화 예산 관리의 과학화와 효율화를 위해 사업의 소요 비용을 초기에 보다 정확하고, 정밀하게 예측하는 것이 중요한 문제로 대두되고 있다. 일반적으로 소프트웨어 프로젝트들은 환경, 기술력, 자본, 개발규모 등 다양한 환경적인 요소들이 동반되기 때문에 소프트웨어 개발 비용을 예측하는 것은 어려운 문제일 수밖에 없고, 중요하다고 할 수 있다.

이러한 문제점을 해결하기 위하여 정보통신부와 한국소프트웨어진흥원에서는 “소프트웨어사업대가기준”을 제정·고시하여왔다. 과거 소프트웨어 사업대가기준의 소프트웨어 개발비용의 경우 본수(本數) 방식이나 스텝(step)수 방식을 이용하여 개발규모를 산출하고 개발 규모에 따라 비용을 산정하는 방식을 따라왔다. 그러나 본수 및 스텝수 방식의 모호성과 부정확성을 보완하기 위하여 2000년경부터 개발자 관점이 아닌 사용자 관점에서 사용기술과는 무관하게 소프트웨어 개발규모를 측정할 수 있는 국제 표준 기법인 기능점수(Function Point)방식을 기본 산정 방식으로 하는 대가기준에 대한 연구를 수행하여 왔으며, 2004년 2월 소프트웨어규모산정 방식을 본수 방식에서 기능점수 방식 중심으로 전환하였다. 소프트웨어규모산정 방식이 기능점수방식으로 전환됨에 따라 합리적으로 규모를 수량화 할 수 있게 되었고, 이를 기반으로 사업초기 단계에서 개발비용을 예측할 수 있는 근거가 마련되었다. 이에 기획예산처에서는 좀더 근거 있고 합리적인 예산 반영을

위하여 소프트웨어개발비와 관련하여 예산 요구 자료를 기능점수방식으로 제출하도록 권고 하고 있다.

하지만, 2004년 소프트웨어개발비 산정방식을 본수에 기능점수 방식으로 전환 후 대다수의 기관에서 기능점수의 개념에 대한 이해 부족과 적용의 어려움으로 이를 잘 활용 하지 못하고 있는 실정이다. 따라서, 국내에서 기능점수 방식을 사용하게 된 근거인 소프트웨어 사업대가기준에 대하여 언급하고 주로 기능점수의 내용과 활용방안에 대하여 언급한다.

## 1. 소프트웨어사업대가기준 소개

정보통신부 고시로 규정하고 있는 소프트웨어사업 대가의 기준은 국가기관 등이 소프트웨어 개발, 데이터베이스 구축, 정보전략계획 수립 등의 정보화사업을 추진함에 있어 정보통신기술의 발전 및 사회적 여건변화에 유연하게 대처하고, 소프트웨어 산업과의 선순환적 구조를 가질 수 있도록 소프트웨어사업에 대한 예산수립이나 발주시 적정비용 등을 산정하기 위한 기준을 제공하는 것을 목적으로 하고 있으며 구성범위는 다음과 같고, 앞으로는 소프트웨어개발비 산정방식인 기능점수 방식을 중심으로 설명하려 한다.

[ 표 1. 소프트웨어사업대가기준 구성 ]

구 분	비용산정 방식	비용구성
소프트웨어개발비	기능점수	개발원가, 직접경비, 이윤
	코드라인수	
	투입인력과 기간	직접인건비, 제경비, 기술료, 직접경비
유지보수 및 재개발비	유지보수	유지보수 시점에서 소프트웨어개발비의 10~15%
	재개발	-
시스템운용환경 구축비	시스템운용환경설계비	공사비요율(기본설계비, 실시설계비)
	시스템운용환경 조성비	공사비
DB구축비	원시자료유형별 데이터량	인건비, 제경비, 이윤, 직접경비
ISP 수립비	컨설팅지수	공수×(컨설팅지수) <sup>0.95</sup> +10,000,000원

### 소프트웨어개발비

소프트웨어개발비는 국제 표준인 ISO 12207의 소프트웨어 개발 기본공정을 적용한 소프트웨어 개발의 단계별 공정<sup>1)</sup>을 수행하는데 필요한 개발원가와 직접경비 그리고 이윤의 합으로 구성되어 있다. 또한 소프트웨어 개발 원가의 산정방식은 크게 두 부류로 나눌 수 있으며, 부류에 따라 세 가지 방식으로 구분된다.

첫 번째, 소프트웨어개발규모에 의한 산정방식에 따라 소프트웨어개발비를 산정하는 방식이 있고, 두 번째는 Mam/Month방식으로 이미 우리에게 친숙한 투입인력과 기간에 의한 산정방식으로 나눌 수 있다. 또한 첫 번째 소프트웨어개발규모에 의한 산정방식은 기능점수 방식과 코드라인수(Line of Code) 방식으로 나누어진다.

소프트웨어사업대가기준에서 소프트웨어 개발규모에 의한 산정 방식인 기능점수 방식과 코드라인수 방식은 개발하려는 소프트웨어의 개발 크기(기능점수, 코드라인수)를 산정하고 여기에 단가(기능점수당 단가, 코드라인당 단가)와 보정요소<sup>2)</sup>를 곱하여 산정하는 방식이다.

[ 표 2. 소프트웨어개발원가 산정 방법 ]

소프트웨어개발원가 산정방법		산정 방법
개발규모에 의한 산정방법	기능점수	기능점수×기능점수당가×보정계수
	코드라인수	코드라인수×코드라인수당가×보정계수
투입인력과 기간에 의한 산정 방법		투입인력수×투입기간×기술자등급별단가

1) 소프트웨어사업대가기준 [별표 1] 소프트웨어 개발의 단계 및 공정

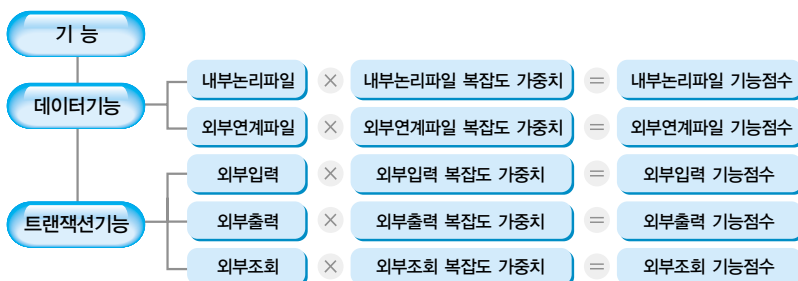
2) 소프트웨어사업대가기준 [별표 10~13]의 보정요소는 규모보정계수, 언어보정계수, 품질 및 특성 보정계수, 어플리케이션 보정계수로 나누어진다.

## 2. 기능점수(Function Point) 산정 방법

국내 소프트웨어사업대가기준은 Albrecht가 기능점수 모형을 제안한 이후, 이를 보완하고 조직에 맞게 수정한 여러 가지 변형된 기능점수 방식이 있지만, 그 중에서도 가장 널리 사용되고 있는 국제기능점수사용자그룹(IFPUG:International Function Point Users Group)에 제안한 기능점수 모형을 채택하고 있으며 이에 대한 기능점수 모형에 대하여 살펴본다.

기능점수 방식은 사용자 관점에서 소프트웨어 규모를 산정하는 방법으로, 주로 논리적 설계를 기초로 하여 소프트웨어가 사용자에게 제공하는 기능의 수를 수치로 정량화하고 소프트웨어의 규모를 산정하는 방식이다. 기능점수 방식은 구매하고자 하는 응용패키지의 규모 산정, 소프트웨어의 품질과 생산성 분석, 소프트웨어 개발과 유지보수를 위한 비용과 소요자원 산정 등에 사용하지만 현재 국내에서는 소프트웨어 신규개발사업에 한정하여 비용을 산정하는 용도로 사용하고 있다. 기능점수 방식은 먼저 개발하려는 소프트웨어의 범위 및 경계를 설정하고 데이터기능(Data Function)과 트랜잭션기능(Transaction Function)을 도출한 후 복잡도 가중치를 적용하여 기능점수를 산출하는 방식이다. 즉, 데이터기능점수(Data Function Point)와 트랜잭션기능점수(Transaction Function Point)의 합계를 [그림 1]에서와 같이 구한다음, 시스템의 특성에 따라 그 값을 보정함으로써 최종적인 소프트웨어의 기능점수를 계산하는 방식이다. 여기에서는 이를 “정규기능점수법”이라고 하겠다. 데이터기능점수와 트랜잭션기능점수의 합을 보정전 기능점수라고 하고, 보정전 기능점수에 보정요소를 곱한 것을 보정기능점수라고도 한다.

[ 그림 1. 데이터 기능과 트랜잭션 기능 및 기능점수 산정 ]



데이터기능점수는 내부 및 외부 자료 요구사항을 만족시키기 위해 사용자에게 제공되는 기능을 말한다. 데이터기능에는 내부논리파일(ILF: Internal Logical File)과 외부연계파일(EIF: External Interface File)이 있다. 내부논리파일은 응용시스템에서 유지(maintain)되고 사용자가 식별 가능한 논리적으로 연관된 자료 및 제어정보(control information)의 그룹을 말한다. 다시 말해서 내부논리파일의 기본적인 용도는 개발하려는 응용시스템의 내에서 단위프로세스(elementary process)에서 유지되는 데이터를 저장하는 것이다.

외부연계파일은 개발하려는 응용시스템에서 참조되지만 다른 응용시스템에서 유지되며 사용자가 식별 가능한 논리적으로 연관된 자료 및 제어정보의 그룹을 말한다. 즉, 외부연계파일의 용도는 개발하려는 해당 응용시스템내의 단위 프로세스에서 참조되는 데이터를 저장하는데 있다. 외부연계파일은 반드시 다른 응용시스템의 내부논리파일이 된다.

데이터기능의 기능점수는 국제기능점수사용자그룹에서 발간하고 있는 기능점수 측정 매뉴얼 (CPM: Counting Practice Manual)에서는 내부논리파일과 외부연계파일 모두 데이터요소유형(DET: Data Element Type)<sup>3)</sup>과 레코드요소유형(RET: Record Element Type)<sup>4)</sup>의 개수에 따라 [표 3]와 같이 복잡도를 결정하고, 그 복잡도에 따라 [표 4]와 같이 가중치를 정하여 각 기능별 기능수에 가중치를 곱하여 산정한다.

[ 표 3. ILF/EIF 복잡도 ]

레코드요소 유형(RET)의 개수	데이터요소유형(DET)의 개수		
	1~19	20~50	51이상
1	낮음	낮음	보통
2~5	낮음	보통	높음
6이상	보통	높음	높음

[ 표 4. ILF/EIF 가중치 ]

복잡도	가중치	
	ILF	EIF
낮음	7	5
보통	10	7
높음	15	10

3) 데이터요소유형(DET): 사용자가 식별가능하고 비 반복적인 유일한 필드

4) 레코드요소유형(RET): ILF나 EIF안에서 사용자가 식별 가능한 데이터요소의 서브그룹

트랜잭션기능은 데이터를 처리하기 위해 사용자에게 제공되는 기능을 말한다. 트랜잭션기능에는 외부입력(EI: External Input), 외부출력(EO: External Output), 외부조회(External inQuery)의 세 가지 기능이 있다.

외부입력은 응용시스템 외부에서 들어오는 데이터 및 제어정보를 처리하는 단위프로세스이다. 외부입력은 주로 내부논리파일을 유지하거나 변경하는데 사용한다. 외부출력은 응용시스템 밖으로 데이터나 제어정보를 보내는 단위프로세스이다. 외부출력은 주로 데이터 및 제어정보의 조회 외에 처리 로직을 통해서 사용자에게 정보를 제공하는데 사용된다. 처리 로직은 수학적 공식 및 계산식을 포함하거나 유도되는 데이터를 생성해야 한다. 이것은 외부조회와 구분되는 사항이기도 하다. 외부조회는 응용시스템 외부에 데이터나 제어정보를 보내는 단위프로세스이다. 외부조회는 주로 데이터나 제어정보 조회를 통해 사용자에게 정보를 제공하는데 있다.

트랜잭션기능의 기능점수는 외부입력, 외부출력, 외부조회 모두 참조파일유형(FTR: File Type Reference)과 데이터요소유형(DET: Data Element Type)의 개수에 따라 [표 5], [표 6]와 같이 복잡도를 결정하고, 그 복잡도에 따라 [표 7]와 같이 가중치를 정하여 각 기능별 기능수에 가중치를 곱하여 산정한다.

[ 표 5. EI의 복잡도 ]

참조파일 유형(FTR)의 개수	데이터요소유형(DET)의 개수		
	1~4	5~15	16이상
0~1	낮음	낮음	보통
2	낮음	보통	높음
3이상	보통	높음	높음

[ 표 6. EO/EQ의 복잡도 ]

레코드요소 유형(RET)의 개수	데이터요소유형(DET)의 개수		
	1~5	6~19	20이상
0~1	낮음	낮음	보통
2~3	낮음	보통	높음
4이상	보통	높음	높음

[ 표 7. EI/EO/EQ의 가중치 ]

복잡도	가중치	
	EI/EO	EQ
낮음	3	4
보통	4	5
높음	6	7

5) 참조파일유형(FTR) : 트랜잭션 기능에 의해 읽히거나 유지되는 내부논리파일 또는 트랜잭션기능에 의해 읽히는 외부연계파일

### 3. 소프트웨어 개발비 산정을 위한 간이기능점수 모형

예산수립시나 사업초기에 기능점수 방식으로 각 기능의 복잡도와 가중치를 도출해 내기는 어려운 경향이 있다. 예산수립 시나 사업제안단계에서는 기능 수준만 도출해 내고 분석/설계단계에서 비로소 기능의 속성까지 상세하게 도출되므로, 속성의 개수(DET, RET, FTR의 개수)를 고려하여 기능의 복잡도를 결정하는 것은 사업초기단계에 현실적으로 무리가 따를 수 있다.

따라서 소프트웨어사업대가기준에서는 사업초기 각 기능의 복잡도와 가중치를 결정하기 어려운 현실을 반영하여 [표 8]의 평균복잡도가중치를 제시 하여 보다 기능점수 방식을 사용하기 편리하게 하였다. 평균복잡도가중치는 기능이 도출되고 복잡도를 결정하는 단계를 생략하고, 제시된 평균복잡도 가중치를 바로 각 기능의 수에 곱하게 된다. 평균복잡도가중치란 과거 수행된 소프트웨어의 기능점수 산정결과를 통계분석하여 내부논리파일, 외부연계파일, 외부입력, 외부출력, 외부조회에 적용된 복잡도에 대해 계산한 가중치들의 평균값을 말한다. 평균복잡도가중치의 적용은 다음과 같은 경우에 적절하게 사용한다. 예산수립단계 및 사업제안단계, 사업초기에 기능점수 산정에 필요한 자료가 충분하지 않은 경우 또는 정상적인 기능점수 산정 결과에 대한 검증이 필요한 경우이다. 여기서는 평균복잡도가중치를 사용한 기능점수 방식을 간이기능점수 방법이라 하겠다.

[ 표 8. 2007년도 평균복잡도가중치 ]

유형	내부논리파일	외부연계파일	외부입력	외부출력	외부조회
가중치	7.4	5.5	3.9	5.1	3.8

따라서, 소프트웨어사업대가기준에서 제시하고 있는 기능점수 방식은 복잡도 적용방법에 따라 크게 국제기능점수사용자그룹에서 정의하는 정규기능점수법과 소프트웨어사업대가기준에서 제시하는 간이기능점수법 두 가지로 구분 할 수 있으며 [그림 2]와 같다.

[ 그림 2. 정규기능점수법과 간이기능점수법 ]



#### 4. 기능점수 활용 방안

예산수립 단계나 사업의 제안단계에서 발주자의 요구사항이 명확히 정의 된다는 것은 사업계약 이후 개발단계에서 기능점수 방식에 의한 규모산정 및 공수 예측에 따라 사업관리 및 개발공정에 있어 보다 체계적인 접근이 가능하며, 발주자 측면에서 개발과정에 적극적인 참여가 필요하여 프로젝트 목표 일정과 범위에 대한 실현가능성을 높여 준다는 것을 의미한다.

기능점수 방식의 기본 정신은 사업초기에 개발하려는 소프트웨어의 범위와 경계를 식별하고 사용자관점에서 기능들을 도출하여 요구사항을 명확히 하고 이를 수치로서 정량화하는 것이다. 구현하려는 소프트웨어 기능들의 목록과 이에 점수를 부여하여 소프트웨어의 개발규모를 정량화 한다는 것은 사업관리 측면에서는 범위관리, 일정관리, 형상관리에서 기능점수 목록을 활용하여 프로젝트 편성 및 변경통제를 강화할 수 있으며 개발공정에서는 기존 산출물의 추가 보안을 통하여 기능점수를 상세 모형으로 유지관리 할 수 있다는 의미와 같다. 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 요구사항관리(Requirement Management)가 크



게 향상될 수 있을 것이다.

위에서와 같이 기능점수는 예산수립이나 사업의 제안 단계에서 요구사항을 정리 할 수 있으며, 이를 정량화 할 수 있기 때문에, 근거 있는 예산 및 사업비를 산정에 도움이 된다. 또한 잘 만들어진 기능점수 목록은 소프트웨어개발사업의 관리에 중요한 수단으로 활용 될 수 있다.

