OLAP 다차원 모델을 이용한 소프트웨어 사업대가기준의 개선

박혜자* · 황인수** · 권기태***

Improvement of Software Cost Estimation Guideline Using OLAP Multidimensional Model

Hyeja Park* · Insoo Hwang** · Ki-Tae Kwon***

■ Abstract ■

This paper presents the ways that can improve the Software Cost Estimation Guidelines in order to replace those that are expected to be abolished at February, 2012, and solve the problems that are being occurred in the current Software Cost Estimation Guidelines. By using multidimensional modeling of OLAP(On-Line Analytical Processing), this paper does three dimensional modeling that considers the product/service view, process view and skill view. Also, it presents the identification method of cost estimation data through the view of each dimension. Furthermore, it defines the software cost estimation process and adapts them into the bottom up estimation and the top down estimation. Finally, it proposes the access of cost estimation data by the multidimensional analysis of OLAP.

Keyword : Software cost estimation, Software cost estimation process, Software Cost Estimation Guideline, OLAP

1. 서 론

1970년대 이후 소프트웨어 위기에 대한 인식과 소프트웨어 개발의 자동화 기술의 발달, 소프트웨어 개발 표준화 준수 및 품질확보 노력 등으로 인해 소프트웨어 관련 공학기술과 관리기술 등이 급속하게 발전하였다[11].

이에 따라 정부에서는 소프트웨어의 개발과 유통을 촉진함으로써 소프트웨어 산업의 발전을 이루고, 향후 수출전략산업으로의 육성을 목적으로 1987년에 소프트웨어 개발촉진법을 공포하였으며, 이듬해 1988년 소프트웨어 개발촉진법 시행령을 공포하여 처음으로 소프트웨어 사업대가 개발비 산정기준을 마련하는 토대를 만들었다.

이후 매년 소프트웨어 사업대가의 지속적인 개선연구를 통하여 보완해 왔지만, 근본적으로 소프트웨어 산업의 기술발전의 속도에 따라가지 못하고 있고, 약 20여년 전에 만들어진 소프트웨어 사업대가기준의 고정된 틀 안에서 다양한 소프트웨어 비용산정 방법을 수용하지 못하고 있는 실정이다. 이로 인하여 소프트웨어 산업 환경변화와 글로벌 경쟁시대에 적응할 수 있는 자율경쟁에 기반을 둔 선진화된 소프트웨어 사업대가 산정기준의 도입 및 활용에 대한 요구가 증가하게 되었다. 또한 소프트웨어 사업대가기준의 근본 문제점을 해소하기 위하여 지식경제부와 정보통신산업진흥원에서는 2012년 2월에 현재의 소프트웨어 사업대가 기준을 폐지한다고 고시하였다[1, 4].

따라서 본 논문에서는 폐지 예정인 현행의 기준을 대체하기 위해 OLAP(On-Line Analytical Processing) 다차원 모델에 기반을 둔 소프트웨어 사업대가기준 개선 방안을 제시하여 소프트웨어 비용산정의 정확성과 일관성을 높이하고자 한다.

본 논문은 다음과 같이 구성된다. 제 2장은 소프트웨어 사업대가기준의 현황과 문제점에 대해 알아본다. 제 3장에서는 소프트웨어 사업대가기준 개선 방안을 살펴보고, 제 4장에서 제안된 개선 방안의 적용 방법을 설명한다. 마지막으로 제 5장을 통해 본 논문의 결론 및 향후 연구 과제를 제시한다.

2. 소프트웨어 사업대가기준의 현황 및 문제점

현행 소프트웨어 사업대가기준은 <표 1>과 같이 정보전략계획수립비, 소프트웨어 개발비, 데이터베이스 구축비, 시스템 운용환경 구축비, 소프트웨어유지보수비, 소프트웨어 운영비의 7개 모듈로 구성되어 있다[8].

<표 1> 소프트웨어 사업유형별 사업대가기준 적용 모듈

예산 구분	사업유형	사업수행요소	적용 기준
사업 수행 예산	건설팅	ISP(정보화계획수립)	정보화 계획 수립비
		BPR/IT 진단/평가/자문	기준 없음
	어플리케이션 개발	신규/변경/삭제 개발	소프트웨어 개발비
		패키지(COTS 활용)도입	기준 없음
		패키지 커스터마이징	소프트웨어 개발비
		재개발	소프트웨어 재개발비
	데이터 변환	변환 소프트웨어개발	소프트웨어 개발비
		데이터 정제/대사 등	기준 없음
		데이터베이스 구축	데이터베이스 구축비
	시스템	운용환경 설계/구축	시스템운용환경 구축비
	서비스	시스템 설치	기준 없음
		사용자/운영자 교육	
		시범운영/운영지원	
사업 관리 예산	유지보수	추가/변경/삭제 등	소프트웨어 유지보수비
	운영	어플리케이션/HW/NW	소프트웨어 운영비
	사업진행 관리	PMO 운영, 품질관리 등	기준 없음
	사업수행 감리	개발/운영	
		상주/비상주 등	

각 모듈의 문제점은 다음과 같다. 정보전략계획 수립비는 컨설턴트의 능력에 따른 수준별 시장 단

가를 고려하지 않고 정보전략계획 수립비의 단가가 확일적으로 책정되기 때문에 고급 컨설턴트의 활용이 필요한 경우 제약을 받을 수 있다. 소프트웨어 개발비는 기능점수당 단가에 관한 공감대가 형성되지 못해 측정한 기능점수 기반 개발비를 신뢰할 수 없으며, 소프트웨어 재개발비는 재개발 규모의 판단은 재사용 소프트웨어의 규모에 의존적이기 때문에 재사용 소프트웨어의 규모를 제시하지 못하는 경우에 적용이 불가능 하다. 소프트웨어 유지보수비는 유지보수가 개발에 종속된 사업이 아니라 오히려 개발 사업이 유지보수를 고려하여 수행되고 있는 실정에서 이를 반영하지 못하고 있다. 또한, 소프트웨어 운영비는 객관적인 산정 근거를 제공하지 않고 투입인력의 수와 기간에 의해 소프트웨어 운영비를 산정하도록 하므로 모호하며, 데이터베이스 구축비는 공공정보의 구축이 필요했던 시절에 만들어진 단순노동에 대한 개념으로 현재에는 활용되고 있지 않는 문제가 있다. 마지막으로 시스템 운용환경 구축비는 정보시스템 환경을 구축하는데 소요되는 공사비용을 산정하기 위한 것으로 소프트웨어와는 직접적으로 관련이

없고, [7]에 이미 포함되어 있다[4].

현행 소프트웨어 사업대가기준은 위와 같은 각 모듈 상의 문제점을 포함하고 있는 것뿐만 아니라 기본적으로 동일한 성격의 소프트웨어를 대상으로 전혀 상이한 접근 방법을 취하는 근본적인 문제점이 있다. 즉, 단일 비용체계를 기반으로 소프트웨어사업의 다양성을 합리적으로 수용하지 못하고 있다. 따라서 이러한 문제점의 보완과 소프트웨어 사업 전 분야에 대한 생명주기 모델인 ISO/IEC 12207:2008을 바탕으로 표준화된 대가산정 프레임워크와 소프트웨어 사업대가기준의 개선 방안이 필요하다.

3. 사업대가기준 개선안의 유도

현행 소프트웨어 사업대가기준은 같은 대상인 소프트웨어에 대해 사업유형별로 서로 다른 사업대가기준을 적용하게 되어 있어 일관성이 없다는 지적을 받고 있다. 이를 보완하기 위해 본 연구에서는 OLAP 다차원 모델을 기반으로 비용산정을 수행하는 새로운 개념의 사업대가기준 개선안을 제안한다.

〈표 2〉 주요 이해관계자들의 요구사항 및 원칙

주요 이해관계자	요구사항	원칙
지식경제부	선진화 모델을 지향하되 우리의 현실을 반영한 대가기준	개방성
	사용이 간단하면서도 다양한 사업에 적용이 가능해야함	단순성
	시장과 환경의 변화에 유연하게 적용할 수 있어야 함	유연성
행정안전부	사업의 특성을 반영한 적정 사업대가 산정이 가능해야 함	적정성
	사업의 유형 및 속성에 따른 다양한 사업에 적용할 수 있어야 함	다양성
발주자 협의회	사업의 유형 및 속성에 따른 다양한 대가 산정이 가능해야 함	다양성
	사용이 간단하며 다양한 사업에 적용할 수 있어야 함	단순성
대기업	사업의 유형 및 속성에 따른 다양한 대가 산정이 가능해야함	다양성
	사업의 특성을 반영한 적정 사업대가 산정이 가능해야 함	적정성
	요구사항 변경을 감안할 수 있는 유연한 대가기준이라야 함	유연성
중소기업	중소기업이 하청할 경우도 대가기준을 적용할 수 있어야 함	적정성
	부분 참여의 경우도 적정 대가 산정이 가능해야 함	모듈화
기획재정부	모든 유형의 소프트웨어 사업 예산산정 내역을 제공할 수 있어야 함	합리성
	수명주기비용이 감안된 적정 예산을 산정할 수 있어야 함	총소유비용
감사원	국가정보화 예산낭비를 근본적으로 방지할 수 있어야 함	총소유비용
	소프트웨어 산업의 육성과 발전에 기여할 수 있는 대가기준이라야 함	시장성

3.1 사업대가기준 선진화 방안 도출 전략

소프트웨어 사업대가기준의 사용자는 매우 많고 다양하기 때문에 이 모든 사용자의 요구사항을 조사할 수는 없어, 주요 이해관계자들을 중심으로 조사한 요구사항과 선진화를 위한 소프트웨어 사업대가기준이 지켜야할 원칙을 정리하면 <표 2>와 같다[4].

위 <표 2>의 원칙을 정리하면 소프트웨어 산업대가기준의 선진화 방안에서 지켜야할 원칙은 적정성, 단순성, 합리성, 유연성, 총소유비용의 5가지이다.

3.1.1 OLAP 다차원 모델링

소프트웨어 비용산정을 위해 수집된 데이터는 일관성 있게 통합, 유지되어야 하기 때문에 데이터웨어하우스로의 구축을 시도해 볼 수 있는데, 다차원 데이터 모델을 이용하면 OLAP을 통한 다차원 데이터 분석이 가능해 진다. OLAP은 의사결정과 기업정보시스템 등에 필요한 정보를 이용, 저장, 조작하는데 사용하는 다차원 도구로써 대용량 데이터베이스나 데이터웨어하우스에 정보요구자가 직접 접근하여 다차원 질의를 통해 대화식으로 정보를 탐색해나가는 과정이다[5]. OLAP의 다차원 데이터 분석을 위한 다차원 모델링은 큐브(cube)를 이용하는데 다차원 질의는 사용자가 큐브의 어떤 부분을 볼 것인지를 정의하는 것으로 큐브를 자신이 원하는 형태로 절단하여 살펴볼 수 있다. 이 다차원 데이터 분석의 개념을 접목하여 소프트웨어 비용산정에 필요한 데이터를 3차원으로 모델링한다. 즉, 비용은 대상을 선정하고, 그 대상에 대한 작업을 식별하고, 각 작업을 수행할 작업자를 결정하여 산정하는 것이다. 따라서 비용산정의 큐브를 [그림 1]과 같이 제품/서비스 관점, 공정 관점, 스킬 관점의 3차원 뷰로 모델링하고 가장 작은 단위인 워크패키지로 절단하여 비용산정을 위한 요소를 식별한다.

제품/서비스 뷰는 사업의 대상인 소프트웨어와

[그림 1] 사업대가기준 개선안 프레임워크 개념도

관련 서비스로, 고객이 대가 지급이 가능한 가시적인 성과물을 기준으로 구분한다. 공정 뷰는 [그림 2]와 같으며, 이는 ISO 12207의 소프트웨어/시스템 수명주기모델을 근간으로 만들어진 “공공부문 소프트웨어사업 발주·관리 표준 프로세스 프레임워크[2, 9]”로 이것을 이용하는 것이 가장 보편적이고 국제표준에 부합할 수 있다. 스킬 뷰는 한국소프트웨어진흥원이 발표한 “소프트웨어 직무능력표준[3]”의 <표 3>을 따르는 것이 현실적이며 표준적인 대안이 될 수 있다.

[그림 2] 공공부문 소프트웨어 사업 발주·관리 표준 프로세스 프레임워크

3차원 뷰의 사업대가기준 개선안 프레임워크를 이용하여 산정단위인 워크패키지를 식별하는데 이는 단위 제품/서비스를 식별하고, 여기에 해당되는 공정을 식별하며, 각 공정을 수행할 직무를 식별하면 [그림 3]과 같은 코드체계를 가진 WBS(Work Breakdown Structure)집합이 된다.

〈표 3〉 소프트웨어직무수행능력표준-통합분류(21종)

번호	직무	직종	분야
01	네트워크 소프트웨어 엔지니어링	소프트웨어 구현	IT서비스, 패키지 소프트웨어
02	데이터베이스 소프트웨어 엔지니어링	소프트웨어 구현	IT서비스, 패키지 소프트웨어
03	미들웨어 소프트웨어 엔지니어링	소프트웨어 구현	임베디드 소프트웨어
04	보안 소프트웨어 엔지니어링	소프트웨어 구현	IT서비스, 패키지 소프트웨어
05	시스템 소프트웨어 엔지니어링	소프트웨어 구현	IT서비스, 패키지 소프트웨어, 임베디드 소프트웨어
06	응용 소프트웨어 엔지니어링	소프트웨어 구현	IT서비스, 패키지 소프트웨어, 임베디드 소프트웨어
07	소프트웨어 테스트	소프트웨어 품질	IT서비스, 패키지 소프트웨어, 임베디드 소프트웨어
08	감리	소프트웨어 품질	IT서비스
09	품질보증	소프트웨어 품질	IT서비스, 패키지 소프트웨어, 임베디드 소프트웨어
10	기술교육	기술 서비스	IT서비스, 패키지 소프트웨어, 임베디드 소프트웨어
11	기술지원	기술 서비스	IT서비스, 패키지 소프트웨어, 임베디드 소프트웨어
12	제품기획	기획	패키지 소프트웨어, 임베디드 소프트웨어
13	시스템운영/유지보수	시스템 관리	IT서비스
14	소프트웨어 아키텍처	아키텍처	패키지 소프트웨어, 임베디드 소프트웨어
15	비즈니스 아키텍처	아키텍처	IT서비스
16	정보기술 아키텍처	아키텍처	IT서비스
17	기술영업	영업 및 마케팅	IT서비스, 패키지 소프트웨어, 임베디드 소프트웨어
18	마케팅	영업 및 마케팅	IT서비스, 패키지 소프트웨어, 임베디드 소프트웨어
19	IT컨설팅	컨설팅	IT서비스
20	비즈니스 컨설팅	컨설팅	IT서비스
21	프로젝트 관리	프로젝트 관리	IT서비스, 패키지 소프트웨어, 임베디드 소프트웨어

[그림 3] 산정단위의 식별체계

3.1.2 소프트웨어 비용산정 프로세스

소프트웨어 비용산정 시에는 산정근거가 필요하며, 산정근거는 타당한 절차에 따라야 한다. 따라서 소프트웨어 비용산정 프로세스를 정의하고, 정의된 프로세스에 따라 비용을 산정하게 된다.

사업대가기준 개선안에서는 산정 프로세스를 [그림 4]와 같이 제시한다. 이는 범위 및 요구사항 정

의 후, 하향식 산정방식과 상향식 산정방식으로 산정한 공수를 전문가 판단에 의해 조정하고, 산정한 비용을 분석하여, 시뮬레이션과 피드백을 반복하여 소프트웨어 비용을 산정하는 프로세스이다.

[그림 4] 사업대가기준 개선안의 산정 프로세스

공수조정 단계에서는 [그림 5]와 같이 전문가 판

단에 의해 하향식과 상향식 방식으로 산정된 공수의 차이를 필요시 반복 산정하여 조정한다.

[그림 5] 공수조정 단계

비용산정 단계에서는 [그림 6]과 같이 인건비와 경비 및 재료비를 합산하는 것으로, 인건비는 조정된 공수를 바탕으로 인건비 단가를 곱하여 구하고, 재료비는 식별된 하드웨어, 네트워크, 전산설비 등을 종류와 규격과 수량 및 가격을 파악하여 구하며, 경비는 사업활동에 필요한 각종 경비항목을 식별하여 구한 후 간접비 배부율을 반영하여 간접원가를 계산한다.

[그림 6] 비용산정 단계

비용분석과 시뮬레이션 단계에서는 [그림 7]과 같

이 비용분석을 하고 필요시 시뮬레이션을 통한 계속적인 피드백을 실시한다.

[그림 7] 비용분석 및 시뮬레이션 단계

3.1.3 OLAP 다차원 분석

OLAP 다차원 모델은 사용자 의사결정을 효과적으로 지원하기위해 양질의 데이터를 최종사용자가 쉽게 접근할 수 있도록 제공한다[6]. 사용자는 다차원 질의를 통해 주요 비즈니스 항목들을 다양한 각도에서 자유롭게 분석할 수 있다. 즉, 다차원 모델링의 3차원 뷰에 슬라이싱과 다이싱, 드릴다운과 드릴업 등의 기법을 이용하여 용이하게 접근할 수 있다.

3.1.3.1 슬라이싱과 다이싱

다차원 질의의 기본은 사용자가 다차원 모델 큐브의 어떤 부분을 볼 것인지 정의하는 것이다. 다차원 질의는 [그림 8]과 같이 마치 사용자가 큐브의 일부분을 자신의 원하는 형태로 슬라이싱(slicing)과 다이싱(dicing)으로 절단하여 살펴볼 수 있다. [그림 1]의 제품 뷰, 스킬 뷰, 공정 뷰의 3차원으로 구성된 큐브를 공정 뷰 축을 기준으로 슬라이싱과 다이싱 하었다고 하자. 큐브를 구성하는 차원 중 제품/서비스 뷰 차원은 보고서의 행을, 스킬 뷰 차원은 보고서의 열을 이루고 있으며 공정 뷰 차원은 보고서의 데이터를 결정하고 있다. 이처럼 다차원 데이터를 2차원 보고서로 표현하기

위해서는 열, 행, 페이지의 3가지 차원개념이 필요하다.

[그림 8] 슬라이싱과 다이싱

3.1.3.2 드릴다운과 드릴업

드릴다운(drill-down)은 요약된 형태의 데이터 수준에서 보다 구체적인 내용의 상세 데이터로 단계적으로 접근하는 분석기법이다[13]. 예로 E-ERP(Extended-Enterprise Resource Planning) 시스템의 영업관리 공수 관련 정보를 분석하다가 보다 상세한 고객관리 공수로 드릴다운하여 분석할 수 있다.

드릴업(drill-up)은 드릴다운과 반대의 방향으로 사용자가 정보를 분석하는 것을 말한다. 일반적으로 드릴다운과 드릴업의 경로는 모델의 계층구조를 따른다. 또한 사용자는 모델이 가진 다양한 속성에 의해서도 드릴다운이나 드릴업을 할 수 있다. 예를 들어, ‘테스팅’이라는 속성을 가질 때, 전체 공정의 공수 데이터를 분석하다가 각 테스트별 공수 데이터로 드릴다운 할 수 있다.

3.2 소프트웨어 개발 사업대가기준

3.2.1 다차원 데이터 모델

현행 소프트웨어 사업대가기준의 문제점은 요구사항을 명확히 할 수 없음에도 할 수 있다는 전제하에 요구사항 기능량을 산정하고, 품질수준에 대한 고려가 제한적이고 프로젝트 제약 요인이 고려되지 않으며, FP(Function Point) 단가 조사에 대한 신뢰가 부족하고, 사업자의 생산성 향상이 반영되지 못한다는 점, 그리고 예산의 수립 및 집행에 유연성이 부족하고 계약 및 변경의 경직성 때문에 대가결정이 어렵다는 점 등이다. 따라서 개선안에서는 요구사항 규모와 요구품질 수준 및 프로젝트 제약 등이 대가산정의 핵심요소인 개발사업은 요구사항 규모에 개발생산성이 반영된 규모당 단가를 곱하고, 여기에 품질수준과 납기 및 요구변경 등의 위험수준을 곱하여 구하는 것을 제안한다.

소프트웨어 개발 사업 관련 공정은 [그림 9]의 발주자 고유의 발주 프로세스, 수주자 고유의 공급 프로세스, 수주자 주도의 개발 프로세스와 지원 수명주기 프로세스, 양자 공통의 관리 프로세스를 포함한다.

[그림 9] 소프트웨어 개발사업 관련 공정

[그림 10]과 같이 소프트웨어 개발 사업비는 OLAP 다차원 분석이 가능하도록 3차원으로 모델링되는데, 발주 프로세스의 발주자 발주비용, 공급 프로세스의 사업자 공급비용, 개발 프로세스의 사업자 개발비용, 관리 프로세스의 발주자 관리비용 및 사업자 관리비용, 지원 프로세스의 발주자 지원비용 및 사업자 지원비용 등으로 구성되며 [그림 11]과 같은 식별도구를 사용할 수 있다.

[그림 10] 소프트웨어 개발사업 3차원 모델

3.2.2 소프트웨어 비용산정 절차

소프트웨어 비용산정 절차는 [그림 4]의 산정 프로세스에 따라 진행되며, OLAP 다차원 모델을 활용하면 다음과 같은 형식의 템플리트를 사용할 수 있다.

하향식 견적을 위해 산정절차([그림 4])의 '범위/요구사항 정의'는 [그림 12]의 ①과 같이 기능을 세부기능의 수준까지 상세하게 분할하고, '규모산정'은 [그림 12]의 ②-1과 같이 기능점수분석의 기능유형으로 식별하여 각 유형별 수량을 파악한다. '하향식 공수산정'은 [그림 13]과 같이 기준생산성에 생산성 보정계수를 보정한 적용생산성을 구한 후, [그림 14]와 같이 규모를 적용생산성으로 나누어 구한 공수를 적용 방법론의 공정별 비중에 공정단위 공수를 산정한다.

①	Req-id	시스템명	기능명	세부 기능명	기능규모점(FSM) 유형					
					정보명	입력	수정	삭제	출력	조회
②-1	RS0101	영입관리	고객관리	고객등록	고객기본정보	1	1	1	1	2
	RS0102			고객변경	고객추가정보	1	1	1	1	2
	RS0103			고객유지	고객관리정보	1	1	1	1	2
	RS0201	판매관리								
	RS0202									
	RS0203									
	RS0301	실적관리								
	RS0302									
	RS0303									
	RP0101	생산관리	제품관리							
합계						7	9	8	6	7

[그림 12] 비용산정 템플리트-1

상향식 견적을 위해서는 산정절차의 '범위/요구사항 정의'는 [그림 15]의 ①과 같이 기능을 최대한 상세하게 분할하고, 'WBS 도출'은 [그림 15]의

WBS id	제품/서비스 구분	발주자 활동	수주자 활동	발주자, 수주자 공동활동										SW직무수행능력									
		5.1 발주 프로세스	5.2 공급 프로세스	7.1 관리 프로세스	5.3 개발 프로세스	6.1 문서화 프로세스	6.2 관리 프로세스	6.3 품질보증 프로세스	6.4 확인 프로세스	6.5 환경관리 프로세스	6.6 합동운영 프로세스	6.7 감리 프로세스	6.8 문제해결 프로세스	SW구현	SW품질	기술서비스	시스템관리	영입및마케팅	컨설팅	프로젝트관리			
1	서비스																						
2	서비스																						
3	서비스																						
4	서비스																						
5	서비스																						

[그림 11] 소프트웨어 개발비 항목 식별 도구

2. 기준 생산성

사업부	업종	FP/MM	비고
금융	금융 공통	23.0	

3. 생산성 보정계수

생산성 영향 요소	능력 인자별 세부 항목	평가	Score	가중치	DI	WDI
사람	PM - 관리능력(리더십, 의사소통, 문제해결)		1	30%	0.3	0.09
	PM - 유사 프로젝트 관리 경험		1	10%	0.1	0.03
	개발자 - 유사 시스템 개발 경험		3	20%	0.6	0.18
	개발자 - 개발 플랫폼 경험		3	20%	0.6	0.18
	지원/관리 - 유사 프로젝트 개발/품질관리 경험		3	20%	0.6	0.18
	소계		11	1	2.2	0.66
기술	개발 도구/개발 언어의 성능		3	50%	1.5	0.23
	개발 플랫폼/Framework 적합성, 성능		3	50%	1.5	0.23
	소계		6	1	3.0	0.45
프로젝트/프로세스	사용자 요구의 명확도 및 안정성		3	50%	1.5	0.23
	개발방법론의 효과성 및 경험		3	20%	0.6	0.09
	프로젝트 환경의 안정성(개발 장소, 언어, 문화, 제도 등)		3	30%	0.9	0.14
	소계		9	1	3.0	0.45
프로젝트	규모(FP)		3	35%	1.1	0.42
	업무난이도(알고리즘)		3	35%	1.1	0.42
	품질보증/테스트 요구 수준		3	20%	0.6	0.24
	적용사이트 수		3	10%	0.3	0.12
	소계		12	1	3.0	1.20
TDI				2.76		
생산성 보정 계수				#NAME?		

[그림 13] 비용산정 템플리트-2

단계	액티비티	기준 비율	수행 여부	적용 비율	조정 비율	소요MM
분석	현황평가	2%	Y	2%	2%	0.6
분석	요구사항정의	4%	Y	4%	4%	1.2
분석	신논리모델구축	3%	Y	3%	3%	0.9
분석	기술구조정의	2%	Y	2%	2%	0.7
분석	프로토타이핑	4%	Y	4%	4%	1.1
설계	기본설계	3%	Y	3%	3%	0.9
설계	본안설계	3%	Y	3%	3%	0.9
설계	상세설계	4%	Y	4%	4%	1.2
설계	클라이언트운영설계	3%	Y	3%	3%	0.9
설계	컨버전설계	6%	Y	6%	6%	1.8
설계	테스트설계	6%	Y	6%	6%	1.8
개발	코딩 및 단위테스트 실시	31%	Y	31%	31%	9.2
개발	테스트실시	14%	Y	14%	14%	4.0
구현	구현계획수립	2%	Y	2%	2%	0.6
구현	컨버전 및 릴리즈	7%	Y	7%	7%	1.9
구현	시스템사용교육	2%	Y	2%	2%	0.6
구현	시범운영	4%	Y	4%	4%	1.2
					0%	0.0
					0%	0.0
					0%	0.0
		100.0%		100.0%	100.0%	29.5

[그림 14] 비용산정 템플리트-3

①		②-2			③-2					
WB S-id	제품/서비스명		수행 공정명		직무별 광수(Man/Day)					
	PJT 명	업무명	프로세스명	활동명	산출물명	SW 테스트	응용SW 명지니 어링	PM/QA O/CM	발주 관리	계
E-ERP 정보 전략 계획 프로젝트	인사관리	재무관리	5.1 발주	제안요청서 준비	-제안요청서 -계약조건, 계약서(조건) -유급지정량 발행(단)				20	20
			5.2 공급	제안서준비	제안서				15	15
			5.3 개발 프로세스	소프트웨어 요구사항분석	소프트웨어요구사항 기술서(단) 소프트웨어 요구사항 검토결과	15	5	5	5	25
			5.3 개발 프로세스	소프트웨어 자작시험	시험결과보고서(소프트웨어 자작시험) 소프트웨어 산출물검토결과	15	5		5	25
	재무관리	7.1 관리 프로세스	관리 실행 및 통제	관리실행기록 관리통제 결과 관리계획서(변경)			100		100	
		6.1 문서화 프로세스	문서화요구 개발	문서화 표준			10		10	
		6.3 품질보증 프로세스	품질보증수행	품질보증 결과(산출물)			35		35	
		합계					15	20	150	45

[그림 15] 비용산정 템플리트-4

②-2와 같이 ①에서 식별된 각 업무 기능별로 해당되는 개발사업 관련 프로세스의 활동을 식별한

다음, ③-2와 같이 각 활동별로 해당되는 직무(스킬)를 모두 찾아 각 직무별로 소요될 것으로 판단되는 공수를 기재한다.

현행 소프트웨어 사업대가기준은 동일 대상물을 상대로 상이한 접근방법을 취하고 있지만, 본 논문에서 제안하는 개선안은 이와는 달리 기획, 개발, 운영 및 유지보수 사업 모두 OLAP 다차원 모델을 기반으로 동일한 접근방법을 가진다.

4. 사업대가기준 개선안의 적용방법

소프트웨어 사업대가 산정기준 개선안을 실제로 적용할 경우, 사용자가 해당 기법에 사용할 충분한 정보를 획득할 수 있으면 상세법을, 보통의 정보를 획득할 수 있으면 보통법을, 기초적인 정보만 획득할 수 있으면 간편법을 사용하는 등 사용자의 정보획득 능력에 따른 3가지 유형이 있다. 본 논문에서는 지면 관계상 상세법만 제시한다.

4.1 하향식 상세법

소프트웨어 사업대가의 하향식 산정모델은 생산

성과 보정계수에 의한 방식을 선정했기 때문에 개발사업의 규모(DFP), 생산성계수(A), 규모보정계수(B), 공수조정 보정계수(EAF)로 구성된 다음과 같은 산정모델을 적용한다.

$$\text{공수} = A \times (DFP)^B \times (EAF)$$

구분	DFP	설계 변경율	코딩 변경율	통합시험 요구율	재사용 SW 평가노력	재사용 SW 년이도	재사용 SW 친숙도
신규개발							
완전재사용		0%	0%	%	0~8%		
수정재사용		%	%	%	0~8%	0~50%	0~1

(소프트웨어 개발규모 = 신규개발 규모 + 완전재사용 규모 + 수정재사용 규모) 보관지, 비관지, 최빈지
 소프트웨어 개발규모 = (낙관지 + 4 × 최빈지) + 비관지 / 6

[그림 16] 하향식 개발비 산정 중 규모산정

[그림 16]과 같이 소프트웨어 개발규모(DFP)는 신규개발, 완전재사용, 수정재사용으로 구분하여 입력하되, 신규개발은 DFP만 입력하면 되고, 완전재사용은 재사용 DFP, 통합시험 요구율, 재사용 소프트웨어 평가노력을 입력하고, 설계변경율과 코딩변경율은 0%로 고정하며, 수정재사용은 재사

용 DFP, 설계변경율, 코딩변경율, 통합시험 요구율, 재사용 소프트웨어 평가노력, 재사용 소프트웨어 이해도, 재사용 소프트웨어 친숙도를 입력한다. 위 3가지를 낙관치와 비관치 및 최빈치로 구분하여 입력하면, 개발 대상 소프트웨어 규모는 [그림 16]에 제시된 식에 의해 계산된다.

규모보정계수	매우 낮음	낮음	보통	높음	매우 높음	극히 높음
1. 친숙성(PREC)	전혀 모름 (0.20)	대체로 모름 (0.46)	잘 모름 (0.72)	대체로 친숙함 (2.48)	상당히 친숙함 (1.24)	완전히 친숙함 (0.00)
2. 유연성(FLEX)	엄격함 (0.07)	중중풀어줌 (0.45)	약간풀어줌 (0.84)	대체로 부합함 (2.03)	상당히 부합함 (1.01)	완전히 부합함 (0.00)
3. 완성성(RESL)	적음(<20%) (0.07)	약간(<40%) (0.55)	중중(<60%) (1.34)	대체로(<75%) (2.83)	상당히(<90%) (1.41)	완전히(>100%) (0.00)
4. 응집력(TEAM)	매우 어려움 (0.40)	대체로 어려움 (0.30)	기본 수준 (0.20)	대체로 합리적 (1.10)	상당함 합업 (1.10)	완전함 합업 (0.00)
5. 성숙도(PMAT)	레벨 1 하위 (7.00)	레벨 1 상위 (6.20)	레벨 2 (4.68)	레벨 3 (3.12)	레벨 4 (1.56)	레벨 5 (0.00)

$B = 0.91 + 0.01 \times \sum SF \dots 0 \leq B \leq 1.2262$

[그림 17] 규모 보정계수 판단

규모보정계수(B)는 [그림 17]과 같이 COCOMO II의 보정계수[10, 12]를 이용하여 각 수준을 평가 하되, 발주자가 결정할 수 있는 유연성, 완성성, 응집력에 대해서는 요구사항을 기준으로 결정하고,

규모보정계수	매우 낮음	낮음	보통	높음	매우 높음	극히 높음
1. 신뢰성(RELY)	약간 불만 (0.82)	간단히 회복(0.92)	다소의 영향 (1.00)	상당한 손실 (1.10)	연명치해 무리 (1.26)	N/A
2. DB규모(DATA)	N/A	D/P < 10 (0.90)	D/P < 100 (1.00)	D/P < 1000 (1.14)	D/P > 1000 (1.28)	N/A
3. 복잡도(CPLX)	(0.73)	(0.87)	(1.00)	(1.17)	(1.34)	(1.74)
4. 재사용(RUSE)	N/A	전혀 고려없음 (0.95)	프로젝트 차원(1.00)	프로그램 차원 (1.07)	제품라인 차원 (1.15)	다중 제품라인 (1.24)
5. 문서화(DOCU)	많은 수명주기 미고려(0.81)	약간의 수명주기 미고려(0.91)	적정수명주기 고려 (1.00)	여분의 수명주기 고려(1.11)	초과수명주기 고려 (1.24)	N/A
6. 시간제약(TIME)	N/A	N/A	~50% (1.00)	~70% (1.11)	~85% (1.22)	~95% (1.63)
7. 공간제약(STOR)	N/A	N/A	~50% (1.00)	~70% (1.05)	~85% (1.17)	~95% (1.46)
8. 기계제약(PVOL)	N/A	큰 변화가 매 12개월마다 있고, 작은 변화가 매 1개월마다 있는 경우 (0.87)	큰 변화가 매 6개월마다 있고, 작은 변화가 매 2주마다 있는 경우 (1.00)	큰 변화가 매 2개월마다 있고, 작은 변화가 매 1주마다 있는 경우 (1.15)	큰 변화가 매 2주마다 있고, 작은 변화가 매 2일마다 있는 경우 (1.30)	N/A
9. 분석능력(ACAP)	15% (1.42)	35% (1.39)	55% (1.00)	75% (0.85)	90% (0.71)	N/A
10. 프로그램능력(PCAP)	15% (1.34)	35% (1.15)	55% (1.00)	75% (0.88)	90% (0.76)	N/A
11. 응용경험(AEXP)	2개월 (1.22)	6개월 (1.10)	1년 (1.00)	3년 (0.88)	6년 (0.81)	N/A
12. 플랫폼경험(PEXP)	2개월 (1.19)	6개월 (1.09)	1년 (1.00)	3년 (0.91)	6년 (0.85)	N/A
13. 도구경험(ITE)	2개월 (1.20)	6개월 (1.09)	1년 (1.00)	3년 (0.91)	6년 (0.84)	N/A
14. 개발자안전(PCON)	48%/년 (1.29)	28%/년 (1.32)	12%/년 (1.00)	6%/년 (0.90)	3%/년 (0.81)	N/A
15. 도구사용(TOOL)	편접, 코드, 디버그 (1.17)	CASE(단순, Frontend, Backend), 작은 통합 (1.09)	기본 수명주기 도구, 보통의 통합 (1.00)	강력하고 성숙한 수명주기 도구, 보통의 통합(0.90)	강력하고 성숙한 순환의 수명주기 도구, 프로세스와 통합, 방법론, 재사용 (0.78)	N/A
16. 다중사이트(SITE)	헤어사이드건 및 대의 전화와 우편으로 의사소통 (1.22)	다중 도시와 다중 기업간 개인전화와 팩스료 의사소통 (1.09)	다중 도시 또는 다중 기업간 업무지역에 열로 의사소통 (1.00)	같은 도시 또는 사내지역에서 광대역 통신비로 의사소통(0.96)	같은 빌딩이나 복합 건물내에서 광대역 통신비와 회사으로 의사소통 (0.86)	완전히 통합된 통신망에서 멀티미디어로 자유자재의 의사소통 (0.80)
17. 밀접함(SCED)	정상연령의 75% 수준 (1.43)	정상연령의 85% 수준 (1.34)	정상연령과 같은 수준 (1.00)	정상연령의 130% 수준 (1.00)	정상연령의 160% 수준 (1.00)	N/A

EAF = 1. 신뢰성(RELY) × 2. DB규모(DATA) × 3. 복잡도(CPLX) × 4. 재사용(RUSE) × 5. 문서화(DOCU) × 6. 시간제약(TIME) × 7. 공간제약(STOR) × 8. 기계제약(PVOL) × 9. 분석능력(ACAP) × 10. 프로그램능력(PCAP) × 11. 응용경험(AEXP) × 12. 플랫폼경험(PEXP) × 13. 도구경험(ITE) × 14. 개발자안전(PCON) × 15. 도구사용(TOOL) × 16. 다중사이트(SITE) × 17. 밀접함(SCED)

[그림 18] 공수조정 보정계수 판단

사업자가 결정해야 하는 친숙성과 성숙도는 보통으로 하는 것이 바람직하나, 발주자가 사업자 선정의 요건으로 친숙성과 성숙도를 결정할 수도 있다. 규모 보정계수가 1보다 커지면 공수가 급격하게 상승하고, 1보다 작으면 완만하게 상승한다.

공수조정 보정계수(EAF)는 [그림 18]과 같이 COCOMO II의 17가지 요인[10, 12]을 각각 평가한 후 각 요인의 평가치를 모두 곱하여 구한다.

[그림 19]는 개발비 산정 하향식 상세법의 직무별 공수할당의 사례로, 하향식 상세법으로 구한 총 공수를 발주자, 사업자, 공통으로 구분하여 할당한 것을 각 공정별로 나눈 후에 다시 이것을 직무별로 나누어 해당 작업의 해당 직무 담당자에게 할당한다. 예를 들어, 개발프로세스의 소프트웨어 상세설계 활동의 소프트웨어 구성요소 상세설계 작업에 대한 소프트웨어 품질 담당자의 공수는 총 공수 중 사업자에게 할당된 2,000 공수에 대해 공정 그룹 내 공수 배부율과 직급 그룹 내 공수 배부율이 고려되어 $8(= 2,000 \times 0.2 \times 0.2 \times 0.5 \times 0.2)$ 공수가 할당이 된다.

4.2 상향식 상세법

기획, 개발, 운영 및 유지보수 사업의 3가지 사업 유형의 상향식 산정기준은 같은 프레임워크를 사용하기 때문에 다음 [그림 20]과 같이 적용공정을

구분해 볼 수 있다.

발주자 공정은 사업자 없이 발주자 단독으로 수행하는 공정이거나 사업자가 참여는 해도 발주자 단독으로 수행되어야 할 공정을 의미하고, 사업자 공정은 발주자와 무관하게 사업자 단독으로 수행하는 공정이거나 발주자가 참여는 해도 사업자 단독으로 수행되어야 할 공정을 의미한다. 또한 공통 공정은 발주자와 사업자가 각자의 조직운영의 필요에 따라 또는 각자의 독립적인 역할에 따라 수행되어야 할 공정을 의미한다.

상향식 산정방법은 WBS를 기반으로 산정하기 때문에 WBS만 도출이 되면 3가지 사업 유형의 기획, 개발, 운영 및 유지보수 사업대가 산정방법은 모두 동일하게 적용된다. 3가지 사업 유형 모두 다음의 산정방법을 따른다. 제품/서비스 구분은 작업 명의 식별에 필요한 수준까지, 공정구분은 작업 단위까지, 직무구분은 세부 직무까지 구분하여 Man Hour 단위의 공수를 기입한다. 공수는 발주자, 사업자, 공통 수행공정 구분을 기준으로 발주자용과 수주자용으로 구분하여 기입하고, 공수의 판단은 최하위 워크 패키지 단위로 식별된 제품/서비스의 규모와 복잡도, 작업의 산출물의 종류와 내용, 직무 수행자의 역량 등을 고려하여 워크 패키지 담당자, 검토자, 관리자의 경험 또는 직관에 의한 판단으로 결정한다.

	제품/서비스 구분					공정 구분			직무 구분									
	시스템명	서비스명	대기명	소기명	단위명	프로세스명	활동명	작업명	산출물명	SW구현	SW품질	기술서비스	기획	시스템관리	아키텍트	영업및마케팅	컨설팅	프로젝트관리
1000	E-ERP	영업관리	고객관리	고객정보관리	고객정보등록	5.1 발주 프로세스	10%	공급자 관리	30%	공급자 외의 협력	20%	회의록	0	0	0	0	0	100
2000	E-ERP	영업관리	고객관리	고객정보관리	고객정보등록	5.3 개발 프로세스	20%	소프트웨어상세설계	20%	SW구성요소상세설계	50%	SW설계기술서(안)	60	20	0	0	20	0
1000	E-ERP	영업관리	고객관리	고객정보관리	고객정보등록	7.1 관리 프로세스	20%	관리실행통제	40%	관리실행	50%	측정지	0	40	0	0	0	60
													16					24

공정 그룹내 공수 배부율

직무 그룹내 공수 배부율

1. 발주프로세스의 공급자와의 협력 작업에 대한 프로젝트 관리자의 공수 = $1,000 \times 0.1 \times 0.3 \times 0.2 \times 1 = 6$

2. 개발프로세스의 SW구성요소상세설계 작업에 대한 SW구현 담당자의 공수 = $2,000 \times 0.2 \times 0.2 \times 0.5 \times 0.6 = 24$

3. 관리프로세스의 관리실행 작업에 대한 SW품질 담당자의 공수 = $1,000 \times 0.2 \times 0.4 \times 0.5 \times 0.4 = 16$

[그림 19] 직무별 공수할당 예

[그림 20] 발주자, 사업자, 공통 공정 구분

제품/서비스 구분				공정 구분				직무 구분			
시스템명	서비스명	대기명	소기명	프로세스명	활동명	작업명	산출물명	SW구분	SW구분	기술구분	시스템구분
E-ERP	영업관리	고객관리	고객정보관리	51 발주 프로세스	공급자 관리	공급자 관리	공급자 관리	30	4	4	4
E-ERP	영업관리	고객관리	고객정보관리	53 개발 프로세스	소프트웨어 개발	소프트웨어 개발	소프트웨어 개발	30	4	4	4
E-ERP	영업관리	고객관리	고객정보관리	71 관리 프로세스	관리실 운영	관리실 운영	관리실 운영	30	4	4	4

[그림 21] 개발사업 대가산정기준 개선안 상세법

위의 [그림 21]은 개발비 산정방안의 상향식 상세법의 예로 E-ERP 시스템 구축 중 영업관리 서비스 시스템 구축을 사례로 들어 [그림 4]의 산정 프로세스에 따라 범위/요구사항을 정의하고, WBS도출 후 공수산정을 한 결과이다.

5. 결 론

2012년 2월 폐지 예정인 소프트웨어 사업대가기준은 국내 소프트웨어 산업의 건전한 발전과 민간자율에 의한 합리적인 대가산정이 가능하도록 개선될 필요성이 있다. 이에 본 논문에서는 현행 소프트웨어 사업대가기준의 개선 방안을 제시하였다.

기획, 개발, 운영 및 유지보수와 같이 사업 유형이 다르더라도 소프트웨어라는 대상물은 동일한

성격을 가지므로 일관된 대가산정 프레임워크를 공유할 수 있도록 OLAP의 다차원 모델을 이용하여 제품/서비스 뷰, 사업공정 뷰, 스킬 뷰의 3차원 모델을 통한 식별 방법을 제안하였다.

또한 소프트웨어 비용산정 시에는 객관적이고 일관성 있는 산정근거가 필요하며, 산정근거는 타당한 절차에 따라야 하기 때문에 소프트웨어 비용산정 프로세스를 정의하였고, 정의된 프로세스에 따른 소프트웨어 대가산정 방안은 하향식 비용산정 모형과 상향식 비용산정 모형을 모두 적용하였으며, OLAP 다차원 모델을 대상으로 하는 다차원 분석을 통해 사용자가 필요로 하는 정보 요구사항을 만족할 수 있으며 다량으로 발생하는 소프트웨어 공학 데이터를 효과적으로 분석할 수 있음을 제시하였다.

본 논문을 통해 국내 소프트웨어 산업의 제값주기 환경의 토대가 마련되고, 소프트웨어 산업의 경쟁력이 제고되는 기회가 되기를 기대한다.

향후 연구에서는 본 논문에서 제안한 소프트웨어 사업대가기준 개선안과 OLAP 도구를 결합하여 다차원 데이터 분석이 가능한 자동화된 비용산정 모델을 제시하고자 한다.

참 고 문 헌

- [1] 김우제, 권문주, “소프트웨어 사업대가기준 중 장기 발전 방향에 관한 연구”, 『한국전자거래학회지』, 제15권, 제1호(2010), pp.139-155.
- [2] 이상은, “공공부문 SW 사업 발주·관리 표준 프로세스 지침 적용방안”, 『전자정부포커스』, 제104권(2006), pp.106-113.
- [3] 박인섭, “SW 직무능력표준을 활용한 수요지향형 대학 SW 교육개선연구”, 『정보통신산업진흥원』, 2009.
- [4] 황인수 외, “SW 사업 대가기준 대체 및 선진화 방안 연구 보고”, 『정보통신산업진흥원』, 2010.
- [5] 이영재, 『지능의사결정지원시스템』, 생능출판사, 2009.
- [6] 조재희 외, 『OLAP 테크놀로지』, 시그마컨설팅그룹, 2006.
- [7] 지식경제부, “엔지니어링 사업대가의 기준”, 지식경제부공고 제2008-109호, 2008.
- [8] 지식경제부, “소프트웨어 사업대가의 기준”, 지식경제부고시 제2010-52호, 2010.
- [9] 지식경제부, “소프트웨어 사업 관리감독에 관한 일반기준”, 지식경제부고시 제2010-55호, 2010.
- [10] Boehm, B. et al., *Software Cost Estimation with COCOMO II*, Prentice-Hall, 2000.
- [11] Sommerville, *Software Engineering 8th Ed*, Addison-Wesley, 2007.
- [12] Boehm, B. et al., “Cost models for future software life cycle processes : COCOMO 2.0”, *Annals of Software Engineering*, Vol.1, No.1 (1995), pp.57-94.
- [13] Kuznetsov, S. D. and A. Yu, Kudryavtsev, “A Mathematicla Model of the OLAP Cubes”, *Programming and Computer Software*, Vol. 35, No.5(2009), pp.257-265.

◆ 저 자 소 개 ◆

박 혜 자 (hye7911@gwnu.ac.kr)

강릉대학교 컴퓨터공학과에서 공학사, 교육학석사를 취득하였고, 동 대학교 박사과정을 수료하였다. 주요 관심분야는 소프트웨어공학, 소프트웨어 비용산정, AHP 응용, 데이터웨어하우스 등이다.

황 인 수 (insoohwg@naver.com)

현재 고려사이버대학교 컴퓨터정보통신학과에서 겸임교수 및 한국정보기술원가표준원에서 부원장으로 재직 중이며, 서울대학교에서 학사, 고려대학교에서 경영정보학 석사를 취득하였고, 삼성SDS(주)에서 Master Technical Expert를 역임하였다. “A Case Study on Process Productivity Model for Software Projects in Korea”, “정보시스템감리보고서 품질개선방안에 관한 연구”, “FP 측정사례집” 등의 다수의 논문, 연구보고서, 저서를 국내외에서 발표하였다. 주요 관심분야는 소프트웨어 비용산정, 소프트웨어 감리, 프로젝트 관리 등이다.

권 기 태 (ktkwon@gwnu.ac.kr)

현재 강릉대학교 컴퓨터공학과 정교수로 재직 중이며, 서울대학교 계산통계학과에서 학사, 석사, 박사를 취득하였다. University of Southern California에서 Barry W. Boehm의 지도로 Post-Doc. 과정을 이수하였다. “기능점수와 소프트웨어 측정”, “소프트웨어공학”, “프로그래밍 언어론” 등 총 12권의 서적을 출간하였고, 국내외에 총 72편의 논문을 발표하였다. 주요 관심분야는 소프트웨어 비용산정, 소프트웨어 신뢰도, 소프트웨어 아키텍처, 데이터 마이닝, AHP 응용 등이다.