

<http://dx.doi.org/10.7236/JIWIT.2012.12.6.155>

JIWIT 2012-6-20

## 실현 가능한 소프트웨어 개발기간 추정

### An Estimating for Practical Software Development Duration

이상운\*, 최명복\*\*

Sang-Un Lee, Myeong-Bok Choi

**요 약** 개발인력을 투입하는 양에 상관없이 소프트웨어를 개발하는데 소요되는 일정 단축에는 한계가 있다. 그러나 획득되는 실제 개발된 소프트웨어 데이터들을 분석하여 보면 불가능한 단기간의 개발기간 또는 비현실적으로 장기간 개발기간 데이터들까지 포함하고 있다. 기존에 제안된 개발기간 추정 모델들은 이러한 제약사항을 고려하지 않고 실제 개발된 소프트웨어 모두를 대상으로 개발일정을 유도하고 있다. 본 논문은 개발 최소기간과 최적 개발기간(정상적인 개발기간)에 대한 기준을 제시한다. 제시된 기준은 실제 수행된 프로젝트를 대상으로 5개의 기준 중에서 가장 성능이 좋은 개발기간 추정 모델을 얻은 기준을 판단하였다. 또한, 이에 기반하여 개발기간을 추정하는 모델을 제시한다.

**Abstract** The compression of software development schedule has a limit without regarding to a development manpower. However, the actually developed softwares includes the impossible short time or unreal long time date. Previously suggested models are driving the development schedule from all of the actually developed software without considering these constraints. This paper recommends a standard of minimum and the most suitable development time period. This paper has chosen a criterion that shows the best development schedule among 5 criteria based on the actually performed projects. Moreover, this paper suggests a model that estimates the development schedule based on the best criterion.

**Key Words** : Impossible Region, Practical Region, Optimal Region, Impractical Region, Cost-Schedule Trade-off

## 1. 서 론

소프트웨어 개발의 성공 여부는 고객이 요구하는 품질 수준(Quality)을 만족하는 제품을 주어진 일정(On Schedule or Duration)과 예산범위 내(On Cost or Budget)에서 납품할 수 있느냐로 결정된다<sup>[1]</sup>. 대부분의 회사들은 주어진 일정에 맞추어 프로젝트를 성공적으로 종료시킬 수 있는지를 알지 못한다. 더욱 중요한 사실은 계약된 일정이 타당하지조차도 알지 못한다는 것이다<sup>[2]</sup>. 따라서 소프트웨어 개발에 소요되는 노력(Effort,  $E$ )과

기간(Duration,  $D$ )은 소프트웨어의 규모(Size)로부터 유도된다. 소프트웨어 규모와 노력의 양이 추정되었을 경우, 어느 정도의 개발기간이 소요될 것인가 또는 개발인원을 추가로 투입하면 얼마나 기간을 단축시킬 수 있는가는 프로젝트 계획을 작성하는 관리자에게는 필수적으로 요구되는 정보이다. 소프트웨어 개발인원(또는 비용)을 증가시키면 개발기간이 단축되는 관계를 갖고 있다. 그러나 개발기간을 어느 한계 이하로는 단축시킬 수는 없다<sup>[3-8]</sup>. 기존의 개발기간( $D$ ) 추정 모델은 개발노력

\*정회원, 강릉원주대학교, 멀티미디어공학과

\*\*중신회원, 강릉원주대학교 멀티미디어공학과

접수일자 : 2012년 8월 9일, 수정완료 : 2012년 10월 28일

게재확정일자 : 2012년 12월 14일

Received: 9 August 2012 / Revised: 28 October 2012 /

Accepted: 14 September 2012

\*\*Corresponding Author: cmb5859@gmail.com

Dept. of Multimedia Engineering, Gangnung-Wonju National University Wonju Campus, Korea

( $E$ )에 기반을 두고 추정되는  $D = a \times E^b$  형태를 취하고 있다<sup>[4]</sup>. 여기서  $a$ 와  $b$ 는 상수이다. 그러나 이 형태의 모델을 제시하는데 적용된 데이터들은 현실적으로 개발이 불가능한 단 기간 내에 개발된 데이터 또는 비현실적으로 장기간에 걸쳐 개발된 프로젝트들도 포함시켜 모델을 제시하고 있다. 이로 인해 주어진 모델을 적용하여 실제 프로젝트 개발기간 추정시 현실적으로 적용이 불가능한 문제점을 갖고 있다.

본 논문에서는 이러한 문제점을 해결하기 위해 먼저 소프트웨어 개발기간에 대해 개발이 불가능한 영역, 현실적 가능영역과 비현실적 영역에 대한 설정 기준에 대한 관련 연구를 검토하여 적합한 기준을 제시한다. 이어서 이 기준에 따라 현실적 영역과 최적 영역에 해당되는 데이터에 기반하여 개발기간을 추정하는 모델을 제시한다.

2장에서는 개발기간 추정과 관련된 연구와 문제점을, 3장에서는 실제 개발 가능기간에 기반한 개발기간 추정 모델을 제시하고, 4장에서는 제안된 모델의 적합성을 평가하여 본다.

## II. 개발기간 추정 관련연구와 문제점

### 1. 개발노력과 개발기간 추정

소프트웨어 비용 추정과정으로 그림 1과 같이 요구사항과 비용인자로부터 개발 노력, 기간과 인력 프로파일을 구할 수 있다<sup>[9,10]</sup>. 특정 규모의 소프트웨어를 개발하기 위한 노력과 개발기간에는 다양한 속성들이 영향을 미친다. 이들 속성은 COCOMO II<sup>[11]</sup>의 20개 비용인자(Cost-Driver)와 ISBSG Benchmark Release 6<sup>[12]</sup>의 17개 속성에서 근거를 찾을 수 있다. 다양한 속성들 중 가장 중요한 속성으로 규모가 있다. 대표적인 소프트웨어 규모 척도로는 LOC(Lines Of Code), FP(Function Point), FFP(Full Function Point), UCP(Use Case Point)가 활용되고 있다.

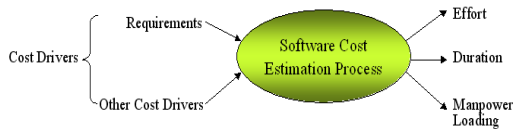


그림 1. 소프트웨어 비용 추정 과정  
Fig 1. S/W Cost Estimation Process

규모, 노력과 개발기간 간에는 그림 2의 관계가 있다.

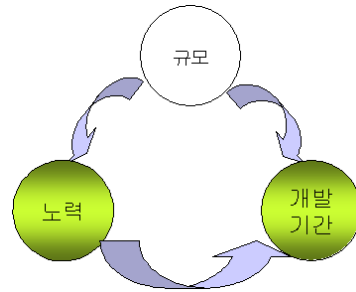


그림 2. 규모, 노력과 개발기간 관계  
Fig 2. Relation between Size, Effort, and Development Duration

소프트웨어 규모로부터 노력과 기간이 추정될 수 있고, 노력으로부터 기간이 추정될 수 있다<sup>[13,14]</sup>. 이에 기반하여 대표적인 개발 노력 추정 모델은 규모에 기반하여 식 (1)의 형태를 보이고 있다.

$$E = a \times Size^b \text{ 또는 } E = a + b \times Size^c \quad (1)$$

여기서  $E$ 는 개발노력(Person-months),  $a, b, c$ 는 경험적으로 유도된 상수이며  $Size$ 는 주요 비용인자로 KLOC (Thousand of Lines Of Code), FP(, FFP 또는 UCP이다. 또한, 대표적인 개발비용 추정 모델은 노력과 규모에 기반하여 식 (2)의 형태를 보이고 있다<sup>[3]</sup>.

$$D = a \times E^b \text{ 또는 } D = a \times Size^b \quad (2)$$

개발노력에 기반한 개발기간 추정 모델을 고찰해 보자. LOC 기반의 개발노력과 기간 추정에 관한 대표적인 연구는 표 1에 제시되어 있다.

표 1. 대표적인 LOC 기반 개발노력과 기간 추정 모델  
Table 1. Development Effort and Duration Estimation Model based on the Representative LOC

모델	개발노력 추정 모델	개발기간 추정 모델
COCOMO 모델	$E = 3.6KLOC^{0.91}$	$D = 2.5E^{0.32}$
Walston Felix 모델	$E = 5.2KLOC^{0.91}$	$D = 4.1E^{0.36}$
Boyston 모델	$E = 5.78 + 3.11KLOC^{1.0}$	$D = 2.15E^{0.33}$

개발노력에 기반한 개발기간 추정 모델 관련연구는 Jones<sup>[4]</sup>, Putnam과 Mayers<sup>[7]</sup>과 Oligney<sup>[13,14]</sup>가 있다. 식 (2)에서  $a$ 의 값은  $2.5 \leq a \leq 5.0$ 까지 다양한 값이 제시되었다. 맥승  $b$ 의 값에 대해 Jones<sup>[4]</sup>는 6,753개 프로젝트 분석 결과  $0.36 \leq b \leq 0.43$ 을 얻었다. 이에 비해, Putnam과 Mayers<sup>[7]</sup>는  $0.25 \leq b \leq 0.38$ 를 제안하고 있다. Oligney<sup>[13,14]</sup>는 ISBSG Benchmark Release 4<sup>[15]</sup>의 312개 프로젝트를 대상으로 분석한 결과 표 2의 모델을 얻었다.

표 2. ISBSG Release 4 기준 개발기간 추정 모델  
Table 2. ISBSG Release 4 Standard Development Duration Estimation Model

Platform	모델	결정계수	참고문헌
Main Frame (MF) 208개 프로젝트	$D = 0.458E^{0.366}$	0.522	Oligney et al. [13]
Mid-Range (MR) 65개 프로젝트	$D = 0.548E^{0.360}$	0.434	
Personnel Computer (PC) 39개 프로젝트	$D = 1.936E^{0.201}$	0.140	
MF+MR 273개 프로젝트	$D = 0.461E^{0.369}$	0.510	Oligney et al. [14]

표에서 얻은  $0.20 \leq b \leq 0.37$ 는 Putnam과 Mayers<sup>[7]</sup>의 결과와 유사한 반면 Jones<sup>[4]</sup>의 결과보다는 매우 낮은 값이다. 표 2에서 제시한 모델의 성능은 결정계수 (Coefficient of determination,  $R^2$ )로 나타내고 있다. 종속변수의 값 ( $D$ )은 독립변수 ( $E$ )에 의해 결정되는 부분과 미지의 오차의 합으로 나타나며, 회귀직선에 의해 설명되는 변동 비율을 결정계수라 한다. 결정계수의 값이 클수록 쓸모 있는 회귀직선이 되지만 좋은 모델로 선정하기 위한 기준은 설정되어 있지 않다. Oligney et al.<sup>[13,14]</sup>가 제안한 모델들은 실측 데이터의 약 50% 이하를 설명하고 있어 좋은 모델로 보기는 어렵다.

ISBSG Benchmark Release 6<sup>[12]</sup>의  $E$ 와  $D$ 가 있는 데이터 637건을 대상으로 분석하여 보면 그림 3의 모델을 얻을 수 있다.

모델의 정확도를 평가하기 위해 Briand et al.<sup>[16]</sup>이 적용한 MMRE(Mean Magnitude of Relative Error)와 Conte<sup>[17]</sup>와 Fenton<sup>[18]</sup>의 Pred(0.25)를 적용하였다.

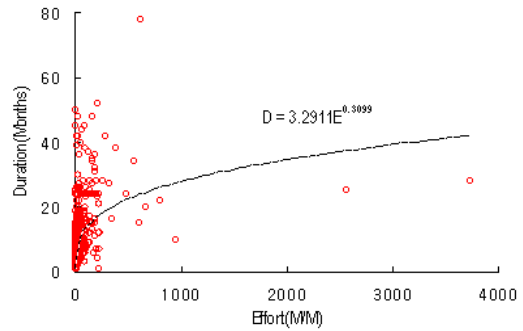


그림 3. ISBSG Release 6 기준 개발기간 추정 모델  
Fig 3. ISBSG Release 6 Standard Development Duration Estimation Model

MMRE가 작은 값이면 평균적으로 좋은 모델임을 알 수 있다. Pred(0.25)는 예측된 값의 MRE가 실측값의  $\pm 25\%$  범위에 있는 개수를 의미한다. Conte et al.<sup>[17]</sup>와 Fenton<sup>[18]</sup>은  $MMRE \leq 0.25$  (25% 이하)이면 개발비용을 예측하는 모델로 채택 가능하며, Pred(0.25)가 75% 이상일 때 좋은 예측모델로 될 수 있음을 제안하였다. 이 모델의 결정계수는 0.2669, MMRE는 56.13, Pred(0.25)는 0.3485이다. 또한  $E$ 와  $D$  간에 상관계수가 0.2809로 상관관계가 매우 미약하고 실제 데이터의  $\pm 25\%$  범위에 속하는 비율이 34.85%로 매우 낮다. 따라서 개발비용을 추정하는 모델로서는 부적절함을 알 수 있다.

지금까지의 연구 결과 개발 플랫폼별로 구분된 경우 또는 다양한 개발 조건을 고려하지 않고 수집된 전체 데이터를 대상으로 개발기간을 추정할 수 있는 적절한 모델이 없는 실정이다. 이러한 문제점을 해결할 수 있는 방법은 개발기간에 영향을 미치는 다양한 속성들을 파악하여 주성분 분석을 통해 영향을 가장 크게 미치는 몇 개의 속성들을 대상으로 다변량 분석으로 모델을 제시하는 방법이 있다. 그러나 이러한 기법들을 적용하더라도 본래 획득된 데이터가 동일한 규모라도 개발 인력 또는 개발 기간에 큰 편차를 갖고 있어 적용에 어려움을 겪고 있다. 이와 같은 결과를 나타내는 주요한 문제점으로는 벤치마킹 데이터에 더 이상 단축할 수 없는 불가능 개발기간 영역과 개발기간을 연장할 수 없는 비현실적 개발기간에 해당되는 프로젝트들이 존재함에도 불구하고 이들 데이터를 포함시켜 적합한 모델을 찾고자 했기 때문이라고 판단된다.

## 2. 개발 일정 단축 한계점

모든 프로젝트는 그림 4와 같이 비용, 기간과 품질로 관리되고 있으며 이들을 프로젝트 3중 제약사항 또는 Iron Triangle이라 부른다. 따라서, 소프트웨어 개발의 성공 여부는 일정, 노력과 품질로 결정된다<sup>[1]</sup>.

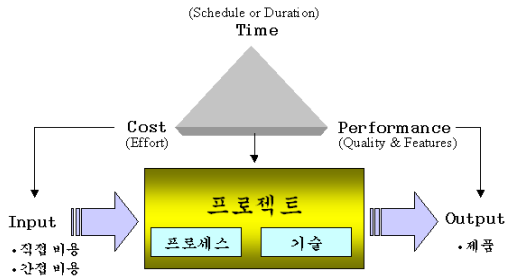
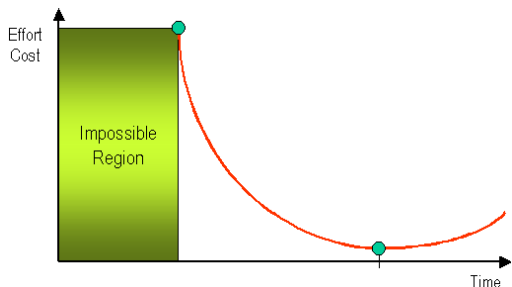


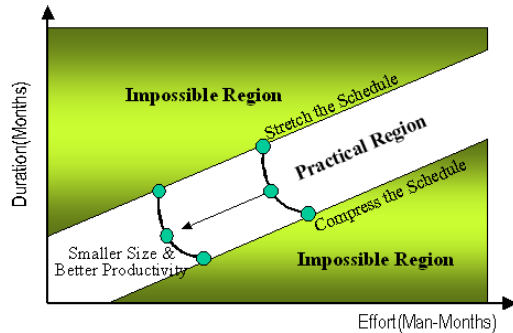
그림 4. 프로젝트의 3중 제약사항  
Fig 4. The Triple-Limitation of Project

개발 조직은 항상 보다 저렴하게(Cheaper), 보다 빠르게(Faster), 보다 좋은 품질(Better)의 시스템을 획득하려는 고객의 압력을 받고 있다. 즉, 프로젝트의 3중 제약사항 모두를 만족시켜야만 한다. 그러나 고객은 비용, 시간과 성능간의 대안분석(Trade-off)을 인식하는 것이 중요하다. 프로젝트를 착수하기 전에 고객의 옵션을 기술하는 다음의 격언을 고려할 필요가 있다. “빠르게, 저렴하게, 좋은 제품을 얻을 수 있다. 이 중에 2가지만 선택하라”.

전형적인 하나의 프로젝트를 대상으로 살펴볼 때 그림 4에서 Performance = Cost(or Effort) x Schedule(or Duration)으로 표현된다. 여기서 Performance를 고정시키면 E와 D는 그림 5의 관계를 갖고 있다. (a)는 독립변수인 개발기간에 따른 종속변수인 개발노력의 관계인데 반해, (b)는 독립변수인 개발노력에 따른 종속변수인 개발기간의 관계로 상호 반대의 결과를 나타낸다.



(a) Silfri, Jones, Carbno, Boehm과 Putnam과 Mayers 제안 형태



(b) Hallowell 제안 형태

그림 5. 전형적인 프로젝트의 노력과 개발기간 관계  
Fig 5. Relation between Effort and Development Duration of Typical Project

Putnam과 Mayers<sup>[7]</sup>는 25년간 수백 개의 개발된 시스템을 분석한 결과 특정한 최소 개발 기간(Minimum Development Duration) 이하에서는 시스템을 완료한 경험이 없다는 결론을 얻었다. 다시 수천 개의 프로젝트들을 조사한 결과에서도 이 격언이 유효함을 증명하였다. 즉, 소프트웨어 개발분야에서는 개발인력을 무한히 추가 하더라도 어느 시점 이하에서는 시스템을 성공적으로 완료할 수 없는 최소한의 개발기간이 존재하며, 이를 불가능 영역(Impossible Region)이라 부른다<sup>[3-7]</sup>. 프로젝트의 규모, 응용형태의 어려움, 개발조직의 생산성 등의 요인으로 인해 프로젝트들마다 많은 차이점을 갖고 있다. 이들 3개 요인이 조합함으로써 불가능영역이 달라질 수 있다. 프로젝트를 성공적으로 완료하기 위해서는 불가능영역을 회피할 수 있어야만 한다. 따라서 불가능영역이 어디에 존재하는지를 아는 것이 중요하다. 즉, 미리 정해진 불가능영역을 인지하는 것은 의사결정자로서 큰 위안이 될 수 있으며, 입찰을 심사숙고할 때 이 개념을 알지 못하는 입찰자들을 앞지를 수 있는 지식을 보유하게 된다.

우리는 최소한의 개발기간에서 개발이 진행되는데 만족하지 않을 것이다. 개발기간을 연장시킴으로서 노력과 결점을 줄일 수 있다. 따라서 개발노력과 개발기간 모두를 고려하여 적절한(최적의) 값을 설정하는 것이 중요하며 이를 비용-기간 대안분석이라 한다.

다음과 같이 시점을 정의하면 개발기간과 개발비용 관계는 그림 6과 같이 4개의 영역으로 분류될 수 있다. 그림 6에서  $T_c$ 는 최소 개발완료 시점(더 이상 단축할 수 없는 개발기간),  $T_d$ 는 최적의 개발완료 시점(개발노력과 개발기간 고려),  $T_o$ 는 최적의 개발완료 시점(최소

의 개발비용 소요시점,  $T_s$ 는 최대 개발완료시점(개발 기간을 더 이상 연장시킬 수 없는 1명이 개발하는 경우의 개발기간), 그리고  $T_a$ 는 실제 개발 완료가 요구되는 시점 ( $T_d \leq T_a \leq T_o$ )을 나타낸다.

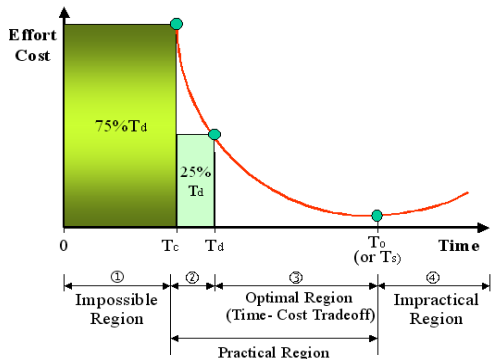


그림 6. 개발기간 영역 분류  
Fig 6. Development Duration Region Classification

정상적인(Nominal) 소프트웨어의 개발 일정 ( $T_d$ )을 최대한으로 단축시킬 수 있는 시점 ( $T_c$ )에 대해 대부분의 연구자들은 약 75%에 한계가 있다고 동의하고 있다. 즉,  $T_c = 75\% T_d$ 가 된다.  $T_c$  시점에서는  $T_d$  시점에서 소요되는 노력의 43%가 추가로 소요된다<sup>[19]</sup>. Putnam과 Mayers<sup>[7]</sup>는 일반적으로  $T_d$ 는  $T_c$ 의 약 130% 근방에 존재한다고 제시하였다.  $T_c = 75\% T_d$ 로부터  $T_d \approx 130\% T_c$ 가 됨을 증명할 수 있다.  $T_d$  시점 이후에는 노력은 보다 감소하고 결점은 보다 작아진다. 이 개념을 프로젝트 계획단계에서 적용하기 위해서는  $T_d$ 의 값을 알아야만 한다. 이에 대해 다양한 연구 결과가 있다. 실제 프로젝트들을 대상으로 한 경험적 결과로 얻은 각 시점 ( $T_c, T_d, T_o$ ) 연구 결과는 표 3에 제시하였다.

Sifri<sup>[3]</sup>는  $T_c = 75\% T_d$ 로,  $T_o$ 는 경험적으로 볼 때  $2T_d$  ( $T_d$  시점 비용의 50% 감소 시점)로 정의하였다. Jones<sup>[4]</sup>와 Marin<sup>[21-23]</sup>은  $T_d$ 만을, Boehm<sup>[6]</sup>은  $T_c$ 만을 제시하였다. Carbno<sup>[5]</sup>는  $T_s$ 를 1명이 소프트웨어를 개발하였을 경우의 개발기간(Limit of Stretch)으로 정의하고,  $T_c$ 를 제안하고 있다.

표 3.  $T_c, T_d, T_o$ 의 경험적 정의

Table 3. Experiential Definition of  $T_c, T_d, T_o$

참고문헌	$T_c$	$T_d$	$T_o$ or $T_s$	분석 대상
Sifri <sup>[3]</sup>	$25\% T_d$	-	$2T_d$	750개 프로젝트
Jones <sup>[4]</sup>	-	$2.5E^{0.33}$	-	-
Boehm <sup>[6]</sup>	$2.15 \times \sqrt[3]{E}$ $= 2.15 \times E^{0.33}$	-	-	99%의 프로젝트가 만족
Carbno <sup>[5]</sup> Ward <sup>[20]</sup>	$\sqrt{E}$	-	1명 개발기간	Rules of Thumb
Putnam <sup>[7]</sup> Mayers <sup>[7]</sup>	$\sqrt{E}$	-	$130\% T_c$	4000개 프로젝트
Marin <sup>[21-23]</sup>	-	$3.0E^{0.33}$	-	Rules of Thumb

Putnam과 Mayers<sup>[7]</sup>도 Carbno<sup>[5]</sup>와 동일한  $T_c$  값을 제시하고 있다. Sifri<sup>[3]</sup>는  $T_d$ 를 정의하지 않아  $T_c$ 와  $T_o$  값을 알 수 없다. 또한 Jones<sup>[4]</sup>의  $T_c$ 와  $T_o$  값, Boehm<sup>[6]</sup>의  $T_d$ 와  $T_o$  값과 Carbno<sup>[5]</sup>와 Putnam과 Mayers<sup>[7]</sup>의  $T_d$  값을 알 수 없다. 따라서 이들 연구 결과로부터  $T_c, T_d, T_o$ 의 명확한 값을 유도할 수 없다. 이와 같이 다양하게 정의된  $T_c, T_d, T_o$  값들과 이들 값 모두를 제시한 연구 결과 부재로 인해 실제 적용시 어느 기준을 적용할 것인가에 혼란을 초래하고 있다.

### III. 실현 가능 기간에 기반한 개발기간 추정 모델

본 장에서는 기존의 개발기간을 추정하는 모델들이 갖고 있는 문제점들을 고려하여 개발 기간을 더 이상 단축시킬 수 없는 불가능 영역과 관련된 연구를 수행한다. 이에 기반하여, 실제로 개발 일정을 더 이상 단축시킬 수 없는 영역과 더 이상 연장시킬 수 없는 영역에 속한 실측 개발 프로젝트를 제외시키고 실제로 수행 가능한 개발기간이 소요된 프로젝트를 대상으로 개발노력에 따른 개발기간을 추정하는 모델을 제시한다.

#### 1. 개발 불가능과 최적 개발 시점 결정

정상적인 개발일정 ( $T_d$ )의 값은 얼마인가? 이 정보

를 알아야만 일반적으로 75%  $T_d$ 인  $T_c$  값과 더불어  $T_o$  값도 결정할 수 있다. 의견이 분분한  $T_c$ ,  $T_d$ 와  $T_o$  시점의 문제점을 해결하기 위해 표 3의 결과를 조합해 표 4와 같이  $T_c$ ,  $T_d$ 와  $T_o$  값을 정의하였다.

표 4.  $T_c$ ,  $T_d$ ,  $T_o$  값 유도

Table 4. Value Derivation of  $T_c$ ,  $T_d$ ,  $T_o$

선정 기준	$T_c$	$T_d$	$T_o$	관련 기준
기준 1	$2.15E^{0.33}$	$T_c$ 로부터 유도 ( $T_c=75\%T_d$ )	$2T_d$	Boehm <sup>[6]</sup> 과 Sifri <sup>[3]</sup>
기준 2	$T_d$ 로부터 유도 ( $T_c=75\%T_d$ )	$2.5E^{0.33}$	$2T_d$	Jones <sup>[4]</sup> 와 Sifri <sup>[3]</sup>
기준 3	$T_d$ 로부터 유도 ( $T_c=75\%T_d$ )	$3.0E^{0.33}$	$2T_d$	Marin <sup>[21]</sup> 과 Sifri <sup>[3]</sup>
기준 4	$\sqrt{E} (=E^{0.50})$	$T_c$ 로부터 유도 ( $T_c=75\%T_d$ )	1명 개발 기간	Carbno <sup>[5]</sup> 와 Sifri <sup>[3]</sup>
기준 5	$\sqrt{E}$	$T_c$ 로부터 유도 ( $T_c=75\%T_d$ )	$2T_d$	Carbno <sup>[5]</sup> 와 Sifri <sup>[3]</sup>
기준 6	$\sqrt{E}$	$T_c$ 로부터 유도 ( $T_c=75\%T_d$ )	130% $T_c$	Putnam <sup>[7]</sup> 과 Mayers <sup>[7]</sup> 와 Sifri <sup>[3]</sup>

ISBSG Benchmark Release 6<sup>[12]</sup>의 개발기간과 개발노력 값이 모두 기록된 637개 데이터를 기준으로 적용한 결과 기준 6의 경우,  $T_d > T_s$ 로 현실성이 없어 적용 대상에서 제외한다. 각각의 기준에 대해 유도된  $T_c$ ,  $T_d$ 와  $T_o$  값은 표 5에 제시하였다.

표 5. 기준별  $T_c$ ,  $T_d$ ,  $T_o$  값 설정

Table 5. Value Settlement on Standard of  $T_c$ ,  $T_d$ ,  $T_o$

선정기준	$T_c$	$T_d$	$T_o$
기준 1	$2.15E^{0.33}$	$2.865E^{0.33}$	$5.730E^{0.33}$
기준 2	$1.875E^{0.33}$	$2.5E^{0.33}$	$5.0E^{0.33}$
기준 3	$2.25E^{0.33}$	$3.0E^{0.33}$	$6.0E^{0.33}$
기준 4	$E^{0.50}$	$1.332E^{0.50}$	1명 개발기간
기준 5	$E^{0.50}$	$1.332E^{0.50}$	$2.664E^{0.50}$

## 2. 실현 가능 기간에 기반한 개발 기간 추정 모델

ISBSG Benchmark Release 6<sup>[12]</sup>의 637개 프로젝트를

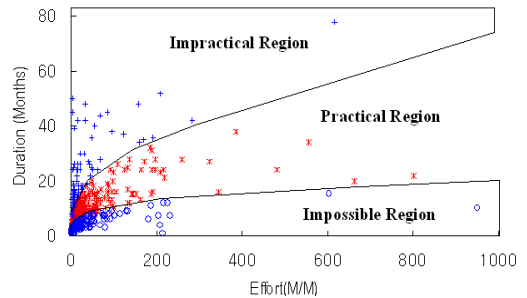
대상으로 표 5의 선정 기준별로 데이터를 분석한 결과는 표 6에 제시하였다. 표 6의 데이터에서 실제영역과 최적영역에 해당되는 프로젝트의 이력 데이터들을 대상으로 가장 적합한 회귀분석 모델인  $D = a \times E^b$  을 제시한다.

표 6. 실험에 적용된 데이터

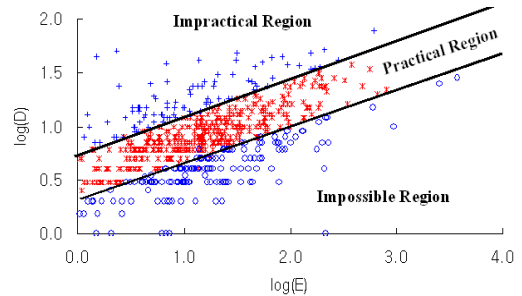
Table 6. Experiment Data

선정 기준	$0 < T_a < T_c$ (불가능 영역)	$T_c \leq T_a < T_o$ (실제 영역)	$T_d \leq T_a \leq T_o$ (최적 영역)	$T_o < T_a$ (비현실 영역)
기준1	171개	367개	242개	99개
기준2	121개	387개	285개	128개
기준3	184개	367개	238개	86개
기준4	89개	352개	276개	196개
기준5	89개	351개	275개	197개

기준 1에 대해 불가능 영역, 실제 영역과 비현실 영역의 데이터를 분류하여 표기하면 그림 7과 같다. 여기서 (a)는 실제의 개발노력(월 인원)과 개발기간(월)의 관계를 표기한 것이며, (b)는 개발노력과 개발기간에 로그를 취한 값이다.



(a) 실측치 관계



(b) 로그 관계

그림 7. 실제 수행 프로젝트의 개발노력과 개발기간 관계  
Fig 7. Relation between Effort and Development Duration of Practical Performance Project

이 그림에서 알 수 있듯이 그림 5의 Hollowell<sup>[8]</sup>이 제안한 개발노력과 개발기간 관계는 오류가 있으며,  $\log(E)$ 와  $\log(D)$ 과의 관계가 실제적으로 Hollowell<sup>[8]</sup>이 제안한 형태와 동일하며, 불가능영역은 다시 위 부분은 비현실적 영역으로, 아래 부분은 불가능 영역으로 분류됨을 알 수 있다.

기준 1에 의해 실제영역에서 개발된 367개 프로젝트의 개발기간에 따른 개발노력의 관계는 그림 8에 제시되어 있다. 프로젝트의 개발기간이 증가할수록 개발노력도 비선형적으로 증가하는 형태를 취한다. 개발노력에 따른 개발기간을 추정하는 모델을 제시하기 위해서는 이 관계의 축을 상호 교환시켜야 한다. 또한, 비선형 관계를 갖는 데이터의 회귀분석을 수행하기 위해서는 선형관계로 변환시키기 위해 변수변환을 수행해야 한다.

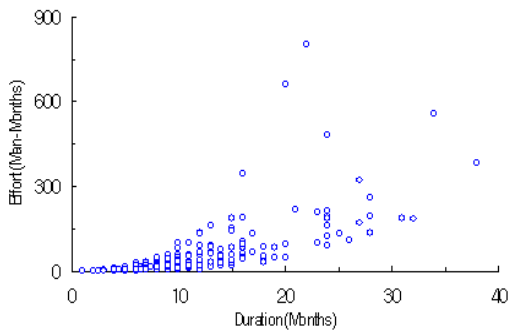


그림 8. 실제영역 수행 프로젝트의 개발기간과 개발노력 관계  
Fig 8. Relation between Development Duration and Effort of Practical Region Performance Project

$D = a \times E^b$  모델을 얻기 위해,  $D' = \log D$ ,  $E' = \log E$ ,  $a' = \log a$  라 하면,  $D' = a' + bE'$ 로 선형방정식이 되며, 이 식으로부터 얻은 모수를 이용하여  $D = 10^{a'} \cdot E^b$ 로 개발기간을 추정하는 모델을 얻을 수 있다. 표 6의 실제영역과 최적 영역에 해당되는 데이터들에 대해 통계 패키지를 이용해 회귀분석을 수행하였다. 예로 기준 1의 데이터에 대해 개발노력과 개발기간에 로그를 취한 그래프에 선형 회귀분석을 수행한 결과 그림 9의  $\log(D_o)$ 와  $\log(D_b)$ 를 얻었다.

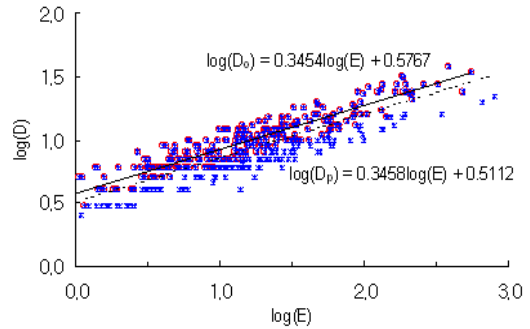


그림 9. 변수 변환을 통한 선형 회귀분석  
Fig 9. Linear Regression Analysis through Variable Change

이 수식들로부터  $D_o = 3.7731 \cdot E^{0.3454}$ 과  $D_b = 3.2449 \cdot E^{0.3458}$ 를 얻을 수 있다. 그래프에서 \*는 실제영역에 해당되는 데이터를, o는 최적영역에 해당되는 실측 데이터를 의미한다. 이 추정 모델의 값과 실측값을 그림 10에 제시하였다. 또한  $D_b$ 는 실제영역 실측 데이터에 대한 개발기간 추정 모델을 의미한다.

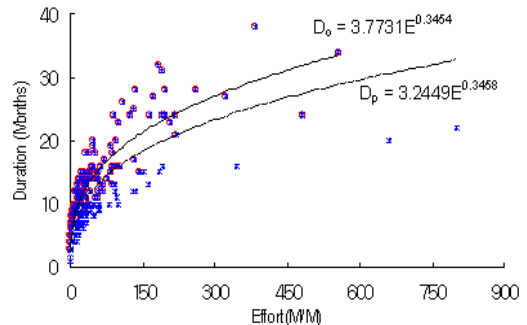


그림 10. 기준 1 적용 개발기간 추정 모델  
Fig 10. Development Duration Estimation Model with Standard 1 Application

#### IV. 모델 평가

5개 기준에 따라 실제영역과 최적영역에서 개발된 프로젝트들에 대한 개발기간 추정모델과 성능(결정계수, 표준오차와 MMRE)은 표 7에 제시되어 있다.

표 7. 개발기간 추정 모델

Table 7. Development Duration Estimation Model

실제영역	모델	모델 성능		
		결정계수	MMRE	Pred(0.25)
기준 1	$D_p = 3.2449E^{0.3458}$	0.7300	22.72	0.6022
기준 2	$D_p = 2.9561E^{0.3429}$	0.7275	23.58	0.5788
기준 3	$D_p = 3.4290E^{0.3388}$	0.7205	25.13	0.5640
기준 4	$D_p = 1.6920E^{0.5285}$	0.6193	29.47	0.4972
기준 5	$D_p = 1.8998E^{0.4644}$	0.8163	22.91	0.4217

최적영역	모델	모델 성능		
		결정계수	MMRE	Pred(0.25)
기준 1	$D_o = 3.7731E^{0.3454}$	0.8544	15.73	0.9132
기준 2	$D_o = 3.2634E^{0.3520}$	0.8491	17.11	0.8967
기준 3	$D_o = 4.0183E^{0.3381}$	0.7197	15.99	0.7479
기준 4	$D_o = 1.8859E^{0.5333}$	0.6904	23.85	0.5580
기준 5	$D_o = 2.0009E^{0.4832}$	0.8988	16.83	0.7855

모델 분석 결과, Sifri<sup>[3]</sup>와 Boehm<sup>[6]</sup>이 제시한 기준을 통합시킨 기준 1을 적용하여 개발기간을 추정하는 모델이 가장 적합한 것임을 알 수 있다. 본 모델은 Oligny<sup>[13,14]</sup>와 ISBSG Benchmark Release 6<sup>[12]</sup>의 전체 데이터를 기준으로 유도한 개발기간 추정 모델보다 성능이 크게 향상되었다. 기준 1의 실제 기간 내의 데이터를 적용한 모델은 Conte et al.<sup>[17]</sup>이 제안한  $MMRE \leq 0.25$  기준을 만족하며, 최적 기간 내의 데이터를 적용한 모델은  $MMRE \leq 0.25$ 와  $Pred(0.25) \geq 75\%$ 의 기준 모두를 만족한다. 따라서,  $2.15E^{0.33} \leq T_a \leq 5.73E^{0.33}$  범위 내에서 개발된 프로젝트는 현실성이 있으므로 이들 개발기간 이내가 되도록 개발노력을 설정하는 것이 필요하다.

## V. 결론 및 향후 연구과제

소프트웨어를 개발시 일반적으로 고객은 보다 빠르게, 보다 저렴하게 개발하도록 압력을 가하고 있다. 그러나 실제적으로 소프트웨어의 개발기간을 더 이상 단축시킬 수 없는 개발 불가능 영역이 존재한다. 이 개념을 알고 있어야만 소프트웨어의 성공적인 개발이 가능하며, 입찰이나 계약시 보다 타당한 정보를 가질 수 있다.

기준에 제안된 개발기간 추정 모델들은 이러한 정보를 고려하지 않고 현실적으로 타당하지 않은 단기간 또

는 장기간의 개발기간 데이터들에 적합되어 현실적으로 적용시 괴리를 발생시키고 있다.

본 논문은 소프트웨어의 개발기간을 개발 가능영역, 최저 개발영역과 비현실적인 개발기간 영역에 대한 기준을 제시하였다. 기존 연구된 다양한 기준을 조합하여 5가지 가능한 기준을 유도하였다. 실제 수행된 프로젝트들을 대상으로 각각의 기준에 따라 실제 가능영역과 최적의 개발기간 영역의 데이터들을 대상으로 적합한 모델을 유도하였다. 모델 유도 결과  $T_d = 2.865E^{0.33}$ ,  $T_c = 2.15E^{0.33}$ 와  $T_o = 2T_d$  시점을 적용하였을 경우 가장 적합한 개발기간 추정 모델을 얻었다.

본 제안된 모델은 개발노력에 따른 개발기간 추정 모델만을 제시하였으며, 소프트웨어 규모에 따라 개발기간이 어떠한 경향을 보이는지에 대한 기준은 제시하지 않았다. 따라서 추후 이 분야에 대한 연구를 수행할 예정이다.

## 참고 문헌

- [1] A. J. Shenher, "Improving PM: Linking Success Criteria to Project Type," Project Management Institute, Creating Canadian Advantage through Project Management Symposium, Calgary, 1996.
- [2] M. Mah, "Controlling Software Development," SPIN Newsletter, 1997.
- [3] G. Sifri, "Accurate Estimates Critical for Software Development Projects," ESI International, Inc., 2001
- [4] D. R. Jones, "Project Scheduling," Augsburg College, 1999.
- [5] C. Carbone, "Optimal Resource Allocation for Projects," Project Management Journal, 1999.
- [6] B. W. Boehm, "Software Engineering Economics," Prentice Hall, 1981.
- [7] L. H. Putnam and W. Myers, "Familiar Metric Management - Time-to-Market, [http://www.qsm.com/fimm\\_08.pdf](http://www.qsm.com/fimm_08.pdf)
- [8] D. L. Hollowell, "Software Project Management Meets Six Sigma," <http://software.isixsigma.com/library/content/c030813a.asp>
- [9] M. Islam, "SEG 3100: Software Development for



- Large Scale Systems," [http://www.csi.uottawa.ca/~misbch/SEG\\_3100\\_Lech2ProjectEstimation.ppt](http://www.csi.uottawa.ca/~misbch/SEG_3100_Lech2ProjectEstimation.ppt), 2004.
- [10] M. R. Vidger, and A. W. Kark, "Software Cost Estimation and Control," National Research Council Canada, 1994.
- [11] B. Boehm, B. Clark, E. Horowitz, R. Modachy, R. Shelby, and C. Westland, "The COCOMO 2.0 Software Cost Estimation Model," USC Center for Software Engineering, 1995.
- [12] ISBSG, "Worldwide Software Development - The Benchmark Release 6," Victoria, Australia International Software Benchmarking Standards Group, 2000.
- [13] S. Oligny, P. Bourque, A. Abran, and B. Fournier, "Exploring the Relation Between Effort and Duration in Software Engineering Projects," World Computer Congress, Beijing, China, 2000.
- [14] S. Oligny, P. Bourque, A. Abran, and B. Fournier, "Developing Project Duration Models in Software Engineering," Draft Version - To be Submitted to the Journal of Systems and Software, <http://www.lrgl.uqam.ca/publications/pdf/528.pdf>
- [15] ISBSG, "Worldwide Software Development - The Benchmark Release 4," Victoria, Australia International Software Benchmarking Standards Group, 1997.
- [16] L. C. Briand, K. E. Elmam, D. Surmann, I. Wiczork, and K. D. Maxwell, "An Assessment and Comparison of Common Software Cost Estimation Modeling Techniques," International Software Engineering Research Network, Technical Report, ISERN-98-27, 1998.
- [17] S. Conte, H. E. Dunsmore and V. Y. Shen, "Software Engineering Metrics and Models," Benjamin/Cummings., 1986.
- [18] N. E. Fenton and S. L. Pfleeger, "Software Metrics: A Rigorous and Practical Approach," International Thomson Computer Press, 1997.
- [19] C. Abts, A. W. Brown, S. Chulani, B. K. Clark, E. Horowitz, R. Madachy, D. Reifer, and B. Steece, "Software Cost Estimation with COCOMO II," Prentice-Hall, 2000.
- [20] J. A. Ward, "Productivity Through Project Management: Controlling the Project Variables," Information Management, 1994.
- [21] Marin Consultancy, "Estimation, Marin Solutions Technical Paper, 2001.
- [22] J. B. Park, H. S. Yang, "Quality Evaluation Method of Open Source Software," Journal of the Korea Academia-Industrial, cooperation Society, Vol. 13, No. 5 pp. 2353-2359, 2012.
- [23] D. S. Kim, H. C. Kim, "The Study of Software Reliability Model from the Perspective of Learning Effects for Burr Distribution," Journal of the Korea Academia-Industrial Cooperation Society, Vol. 12, No. 10 pp. 4543-4549, 2011.

## 저자 소개

### 이 상 윤(정회원)



- 1995년~1997년 : 경상대학교 컴퓨터 과학과 (석사)
- 1998년~2001년 : 경상대학교 컴퓨터 과학과 (박사)
- 2007년~현재 : 강릉원주대학교 과학 기술대학 멀티미디어공학과 부교수

<주관심분야 : 소프트웨어 공학, 그래프 알고리즘 등>

### 최 명 복(중신회원)



- 1994년 : 아주대학교 컴퓨터공학과 (석사)
- 2001년 : 아주대학교 컴퓨터공학과 (박사)
- 1997~현재 : 강릉원주대학교 멀티미디어공학과 교수
- 2004년~현재 : 한국인터넷방송통신

학회 이사

<주관심분야 : 소프트웨어 공학, 알고리즘 등>