

# 사례연구를 통한 오픈소스 소프트웨어 비용 산정방법에 대한 연구

신성진\*, 이후재\*\*, 류성열\*\*  
 \*숭실대학교 컴퓨터학과  
 e-mail : {rwim, hooklee, syrheiw}@ssu.ac.kr

## A study on Cost Estimation of Open Source Software through a Case Study

Sung-Jin Shin\*, Hoo-Jae Lee\*\*, Sung-Yul Rhew\*\*\*  
 \*Dept. of Computing, Soong-Sil University

### 요 약

최근 기업들은 소프트웨어의 품질, 개발 속도, 비용 등의 문제에 대한 해결을 위해서 오픈소스 소프트웨어를 활용한 개발 방법을 사용하고 있다. 이에 따라 OSS 를 비롯한 오픈소스에 대한 여러 분석이 있었지만, OSS 를 활용할 때 이에 따른 비용이 줄어들 것이라라는 예상은 많았지만, 그것을 실제로 증명할 만한 사례나 연구는 현재까지 많이 존재하지 않았다. 본 연구에서는 OSS 를 활용하기 위한 비용요소 및 범위를 제시하고 실제 사례를 통해서 각 비용을 구해서 이에 대한 적절성 및 분석을 수행하고 향후 연구를 제시한다.

### 1. 서론

최근 기업들은 소프트웨어의 품질, 개발 속도, 비용 등의 문제에 대한 해결을 위해서 오픈소스 소프트웨어(Open Source Software: OSS)를 활용한 개발 방법을 사용하고 있다[8]. 이에 따라 OSS 를 비롯한 오픈소스에 대한 여러 분석이 있었지만, OSS 를 활용할 때 이에 따른 비용이 줄어들 것이라라는 예상은 많았지만, 그것을 실제로 증명할 만한 사례나 연구는 현재까지 많이 존재하지 않았다.

따라서 본 연구에서는 관련 비용 연구를 바탕으로 선정할 OSS 의 범위를 정한 다음 각 범위의 OSS 비용 산정방법을 제시하고, 하나의 사례를 통해서 산정 방법에 대한 유효성 검토 및 향후 개선사항을 제시한다.

### 2. 관련연구

#### 2.1. Dan Woods 의 오픈소스 비용

Dan Woods(2005)는 “Open Source for the Enterprise”를 통해서 오픈소스를 활용하였을 때의 비용 및 세부 비용 요소를 다음 <표 1>과 같이 제시하였다[1]. <표 1>을 통해서 오픈소스를 활용하였을 때 각 비용을 알 수 있었지만, 위의 요소를 가지고 실제 비용을 어떻게 구해야 되는가에 대한 설명이 없어서 오픈소스의 분류 별 비용의 종류를 파악하는 것으로만 참고가 가능하다. 따라서, 본 연구에서는 위의 요소를 토대로 실제 비용을 어떻게 구할 수 있는지를 분석하고 하나의 사례를 들어서 수행하도록 한다.

<표 1> 오픈소스 활용 분류 별 세부 비용

비용 요소	세부 비용
선정비용	검색 및 테스트 환경 구축, 선정기준 연구
설치/설정비용	실제 프로그램 설치 및 설정, 개발자 교육
통합/맞춤비용	소스코드 분석을 통한 수정 및 통합, 테스트
운영/지원비용	H/W, S/W 및 네트워크 구축, 보안 및 백업지원

#### 2.2 공개소프트웨어의 비용구조에 관한 연구

한국소프트웨어진흥원(2003)에서는 공개소프트웨어의 비용구조에 관한 연구를 수행하였다. 이 연구에서는 공개소프트웨어 TCO 의 구성에 대해서 80.6%가 도입비용 및 관리비용으로 나누고, 관리비용은 기술 지원, 하드웨어 유지보수, 업그레이드, 추가 라이선스 비용으로 나누는 것이 타당하다 하였다[6].

이 연구에서의 TCO 조사 분석 결과에서 OSS 의 비용절감 측면을 앞의 비용 구조로 나누었을 때 다음 <표 2>와 같이 평가하였다[6]. 이러한 비용절감 측면을 토대로 하였을 때 나올 수 있는 결론은 초기 비용 및 관리 비용의 전반적인 요소가 다른 소프트웨어보다도 OSS 가 적게 든다는 것을 알 수 있다.

<표 2> OSS 의 비용절감 요소

비용 요소	세부 비용
도입비용	공개소프트웨어 자체는 무료이지만, 이에 대한 지원사항을 얼마의 금액을 지불함
업그레이드 비용	장기적 업그레이드 비용이 적게 든다. 다운로드만 받거나 적은 비용의 재배포 가능 공개소프트웨어 공급자 변경이 자유로움
라이선스비용	라이선스 관리나 소송의 위험이 없음

하지만 이 연구에서 위 사항을 토대로 한 실제 사

례는 제시되었지만, 구체적인 비용 산정 방법은 제시되지 않았다. 제시된 사례가 각각 환경 및 비용 산정 기준도 다 다르기 때문에 사례를 통한 비용 산정 방법을 쉽게 정할 수 없었다.

따라서 본 연구에서는 앞에서 언급된 비용구조를 가지고 이를 간단한 OSS 활용 사례에서도 그대로 적용이 될 수 있는 것인지를 알아보고, 또한 앞의 비용 구조가 일치하는가를 분석한다.

### 2.3 OSS 의 범위 선정을 위한 연구

OSS 비용 연구 수행을 위해서는 OSS 의 범위가 어디에 속하는가를 알아보고, 이에 대한 분석이 필요하다. 정낙수(2005)는 대부분의 기업에서의 OSS 도입 범위를 운영체제, 사무자동화, 유틸리티 및 어플리케이션(이하 어플리케이션)으로 나누고, 이들의 적용 범위에 따라 OSS 의 적용 범위가 결정된다[7]. 또한 OSS 적용 범위의 확대를 위해서는 Windows 와 Linux 를 동시 지원하는 어플리케이션 단계의 소프트웨어를 적용하여 OSS 에 대한 인식을 개선한 후, 도입 규모도 점점 넓혀 나가야 한다고 하였다[7]. 이를 토대로 OSS 의 비용을 구하기 위해서는 먼저 규모가 작은 어플리케이션에 대한 연구가 우선되어야 한다.

OSS 어플리케이션을 구하기 위한 방법으로는 특정 업체와의 계약 또는 오픈소스 커뮤니티에서 도입과 같은 두 가지 방법이 있다. 현재 Sourceforge 와 같은 오픈소스 커뮤니티는 점점 성장하고 있고[3], 이에 따른 OSS 의 적용을 위해서 T. R. Madanmohan(2004)은 기존의 소프트웨어에 뒤떨어지지 않고도 올바른 지원 체계가 이루어지는 OSS 를 선정해야 한다고 하였다[2]. 이 기준에 맞춰서 OSS 어플리케이션을 도입하면 기존 소프트웨어와 큰 차이가 없으므로 오픈소스 커뮤니티를 통한 도입방법도 점점 늘어나고 있다. 또한, 최근의 소프트웨어 개발이 컴포넌트 중심으로 되어있다는 것을 감안하여[9], 컴포넌트 기반의 소프트웨어를 사용하는 것이 비용 산정에도 더욱 적절하다.

따라서 위의 내용을 종합하여 본 연구에서 수행할 OSS 의 범위를 다음 <표 3>과 같이 정할 수 있다.

<표 3> OSS 활용을 위한 사례 선정

구분	세부 비용
적용 범위	어플리케이션 Windows 와 Linux 를 동시에 지원하는 소프트웨어
도입/적용방식	오픈소스 커뮤니티를 통한 도입
도입선정기준	지원 및 관리가 올바르게 이루어진 소프트웨어
소프트웨어형태	컴포넌트 기반 소프트웨어

### 3. OSS 비용 요소 분석

대부분의 OSS 는 라이선스 비용 및 제품 비용이 들지 않는다 하였지만, 교육이나 수정/맞춤, 업그레이드 등에 따른 비용 요소가 실제 상용 소프트웨어보다 더 나올 수도 있다는 것을 제시하였다[1, 2, 6, 7]. 이때 앞의 연구를 통해서 공통적으로 도출할 수 있는 OSS 의 비용 요소들 중 본 연구에서 사용하게 될 세부 비용 요소를 본 연구에서는 다음과 같이 제시한다.

- OSS 의 이해를 위한 교육 비용
- OSS 의 수정 및 맞춤에 필요한 비용
- OSS 활용을 위한 유지보수 비용

#### 3.1 교육 비용

OSS 는 일반적인 소프트웨어와는 사용법 자체도 다르고 OSS 별로 일정한 사용에 대한 기준이 없기 때문에, 이에 대한 교육을 별도로 필요로 한다. 오픈소스 커뮤니티를 통하여 OSS 를 도입할 경우에는 개발자가 직접 OSS 를 획득한 다음, 관련 기술문서의 이해 및 직접 설치 설정을 통해서 습득한다. 본 사례에서는 단일 사용자가 OSS 를 활용한 경우이므로 앞에 언급된 노력 외에 다른 것은 필요로 하지 않는다.

$$\text{교육비용} = \text{개발자의 습득 시간대 비용} + \text{다른 개발자에게 알려주는 교육 시간대 비용}$$

#### 3.2 수정 및 맞춤 비용

OSS 는 소스 코드를 볼 수 있는 특징이 있기 때문에 이를 원하는 시스템에 맞게 수정하는 작업을 필요로 한다. 그렇다면 OSS 의 수정 비용을 구하기 전에 수정에 필요한 요소가 무엇이 있는지를 살펴본다.

- 수정에 필요한 시간 및 노력
- 수정하려고 하는 OSS 의 규모
- 수정 이후의 테스트 시간

여기서 수정에 필요한 시간 및 노력은 개발자들이 투자한 시간을 중심으로 한다. 그리고 수정하려고 하는 OSS 의 규모를 예측하기 위해서는 소스코드 분석을 통해서 규모를 판단하게 된다. 따라서 소스코드 분석을 위해서 클래스를 규모 단위로 하여 구한다[4]. 테스트 시간의 경우는 하나의 기능 단위에 따른 테스트를 수행하므로, 컴포넌트를 단위로 하여 테스트를 하는 것이 적절하다. 따라서 본 연구에서는 컴포넌트 기반수정 및 맞춤 비용을 다음과 같이 제시한다.

$$\text{수정/맞춤비용} = \text{수정노력} * \text{클래스 개수} + \text{테스트노력} * \text{컴포넌트 개수}$$

#### 3.3 유지보수 비용

OSS 의 유지보수 비용은 앞의 2.2 절의 연구의 관리비용이라 할 수 있다. 관리비용은 기술지원, 하드웨어 유지보수, 업그레이드, 추가 라이선스 비용으로 나눈다 하였는데[6], 그렇다면 오픈소스 커뮤니티를 통한 어플리케이션 OSS 도입 시 각 사항이 어떤 식으로 적용되는지를 다음 <표 4>를 통해서 알아본다.

<표 4> 오픈소스 커뮤니티를 통한 어플리케이션 OSS 활용 시의 관리비용

구분	내용
기술지원	벤더가 없으므로 별도로 존재하지 않음
추가 라이선스	벤더가 없으므로 별도로 존재하지 않음
H/W 유지보수	환경구축이 되어있으므로 별도로 존재하지 않음
업그레이드	버전 변경 시 적용 노력 필요

오픈소스 커뮤니티를 통하여 OSS 를 도입할 때에는 OSS 에 대한 벤더가 따로 존재하지 않기 때문에 별도의 기술지원 및 추가 라이선스는 없다. 또한 도입하는 OSS 가 어플리케이션인 것을 감안하면, 도입 전에 모든 환경은 구축된 상태이고 이에 알맞은 OSS 를 도입하게 되므로 역시 별도의 비용이 없다.

업그레이드 비용의 경우는, 버전 변경 시에 따른 관리가 필요하기 때문에 변경된 소프트웨어에 대해서 이에 맞는 수정/맞춤 작업을 해야 한다. 업그레이드 역시 수정/맞춤의 일부분을 다시 수행하는 절차이므로, 비용을 구하기 위해서는 업그레이드에 필요한 수정 규모를 업그레이드 기간에 따라 미리 정하고 기간에 맞게 파악한 후 이에 따른 수정 노력을 구한다. 본 연구에서는 이에 알맞은 방법으로 단순이동평균법 [5]을 이용하여 비용을 산정하는데, 이 방법은 이전 시기 별로 존재하는 규모를 가지고 단위 기간 내의 규모에 따른 평균치를 구하고 이에 따른 추세선을 구해서 전체의 필요한 노력을 산정한다.

$$\text{업그레이드 비용} = \text{단위이동평균법을 이용한 기간 내 클래스 규모} * \text{클래스 하나의 평균 수정시간}$$

오픈소스 소프트웨어 자체에도 유지보수가 필요하고 이에 대한 관리가 지속적으로 이루어져야 하지만, 오픈소스 커뮤니티 상의 어플리케이션 단계에서는 업그레이드 비용 외에 필요한 비용은 특별히 존재하지 않은 것으로 나타났다. 따라서 본 연구에서의 유지보수 비용은 업그레이드 측면의 비용부분을 고려한다.

#### 4. 사례 분석

OSS 에 대한 비용을 구하기 위해서 앞의 <표 1>의 기준을 가지고 개인일정관리 기능을 하는 어플리케이션인 BORG Calendar 를 선정하여 이에 대해서 비용을 어떻게 구하는 지를 살펴보았다. 해당 OSS 가 위의 기준에 올바른 지를 확인하기 위해서 다음 <표 5>와 같이 나타내었다.

<표 5> 실제 연구에 사례 분석에 사용할 OSS

구분	내용
OSS 명칭	BORG Calendar (개인일정관리)
OSS License	GPL License
OSS 획득방법	Sourceforge 를 통해서 획득
OSS 사용기간	1년 주기로 결정
OSS 형태	컴포넌트 기반 소프트웨어
지원 운영체제	Windows, Linux, Mac OS, OS Independent
프로젝트 활동	172,626 개 프로젝트 중 1,809 위 5년 활동기간, 업데이트 35 회, 활동력 99.42% 버그 148 건 중 141 건 처리(95.3%)
OSS 사용자 능력	중급 프로그래머(Intermediate Programmer) 급여: 시간당 10,000 원 계산

본 연구에 사용할 OSS 의 활용 형태 및 운영체제, 형태는 앞에서 제시한 기준에 만족하였고, 선정 기준에 있어서도 실제 Sourceforge 에 나와있는 같은 종류의 OSS 보다 더욱 만족할 만한 기준을 가지고 있었다. 이의 활용을 위한 환경은 다음 <표 6>과 같다.

<표 6> BORG Calendar 환경

구분	내용
운영체제	Windows XP
Java	J2SDK1.4.2_16
Eclipse	Eclipse 3.0.2
MySQL	MySQL 5.0.51a

#### 4.1 교육 비용

BORG Calendar[10]를 도입하였을 때의 교육 비용은 단일 사용자의 경우 프로그램을 실제로 다운로드하여 설치 및 설정을 하는 시간과 관련 문서 및 도움말을 통한 이해 시간에 대한 비용이다. 다음 <표 7>은 실제 설치/설정 및 교육에 필요한 시간을 나타낸다.

<표 7> BORG Calendar 설치/설정에 필요한 시간

구분	내용	비용
다운로드/설치 시간	30 분	5,000 원
설정 및 문서 이해 시간	3 시간	30,000 원
교육 비용	-	35,000 원

실제 사례에서는 단독으로 연구를 수행하였기 때문에 교육 시간은 오래 걸리지 않았다. 하지만 설정 및 문서 이해시간은 어플리케이션에 따라 달라질 수 있다. 이는 BORG Calendar 가 다양한 환경의 OS 를 지원한다고 해도 이에 따른 데이터베이스 설정이 필요하였고, 또한 Java 개발 환경에서 필요로 하는 별도의 API 및 연동을 필요로 하였기 때문에, 더욱 복잡한 환경에서 OSS 를 활용할 때 이에 따른 설정 및 문서 이해 시간은 많이 필요하다.

또한, 단일 사용자가 아닌 기업 내에서 사용할 경우에는 많은 사용자가 이용해야 하므로, 이에 따른 교육에 대한 노력도 감안을 해서 비용을 산정해야 한다.

#### 4.2 수정 및 맞춤 비용

수정 및 맞춤 비용은 소스코드 분석 및 수정 시간과 실제 테스트 시간으로 나눌 수 있고, 또한 프로그램의 규모도 알아야 한다. BORG Calendar 역시 컴포넌트로 구성된 소프트웨어이므로, 단위를 클래스와 컴포넌트로 하여 개수를 구한다. 분석 결과, 클래스는 모두 167 개가 있고, 컴포넌트는 20 개가 있는데, 하나의 단위에 대한 소스코드에 대한 분석/수정 시간은 평균 10 분이 소모되었고, 테스트 시간은 2 시간으로 하였다. 이를 토대로 한 비용은 다음 <표 8>과 같다.

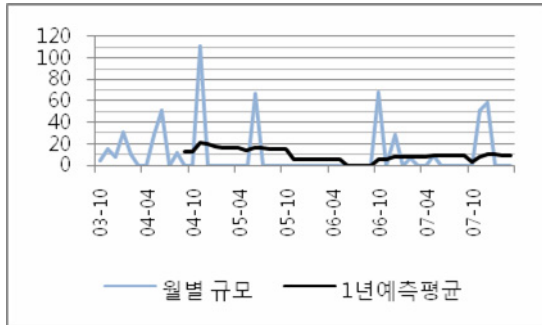
<표 8> OSS 설치/설정에 필요한 시간

구분	내용	비용
분석/수정시간	167 * 10 분 = 27.83 시간	278,300 원
테스트 시간	20 * 2 시간 = 40 시간	400,000 원
수정/맞춤비용	-	678,300 원

#### 4.3 유지보수 비용

유지보수 비용은 앞에 언급한 바와 같이 업그레이드 비용을 중심으로 하고, 이에 따른 업그레이드 노력을 구한다. BORG Calendar 는 2003 년 10 월부터 지금까지 버전 변경이 35 차례 이루어졌고, 버전 변경 주기가 불규칙하기 때문에 버전 변경 시 필요한 클레

스 수정 규모를 월 단위로 측정하였다. 그 다음으로 는 OSS 사용기간을 1 년 단위로 한다고 하였으므로, 변경 규모에 따른 추세를 1 년 평균으로 하여 구하였다. 다음 <그림 1>은 버전 변경에 대해서 OSS 업그레이드를 수행할 때의 규모를 나타낸 것이다.



<그림 1> 업그레이드에 따른 클래스 변경 규모

<그림 1>에서는 월별 규모 및 1 년 예측 평균 추세를 나타냈는데, 향후 1 년 동안의 OSS 업그레이드 규모를 알기 위해서는 추세선의 전체적인 평균을 측정하여야 한다. 개발 초기 단계에는 버전 변경에 따른 규모 변동이 커서 추세선 역시 불규칙하였지만, 2007 년 이후에는 추세선이 거의 규칙적으로 나타나기 때문에 추세선 전체의 평균값인 10.64 를 가지고 1 년 동안의 클래스 변경 규모를 산정한다. 이 때의 변경 규모는 약 128 로 나타났으며, 이를 이용했을 때의 업그레이드 비용은 다음 <표 9>와 같이 나타낸다.

<표 9> 업그레이드 비용

구분	내용	비용
업그레이드	128 * 10 분 = 21.33 시간	213,300 원
유지보수 비용	-	213,300 원

4.4 컴포넌트의 총 비용

다음 <표 10>은 앞의 절차에 따른 비용의 합을 나타낸 것이다.

<표 10> 실제 사용된 OSS 의 총 비용

절차	BORG Calendar
교육 비용	35,000 원
수정 및 맞춤 비용	678,300 원
유지보수 비용	213,300 원
<b>총 비용</b>	<b>926,600 원</b>

이 결과를 통해서 알 수 있는 사실은 실제 사례 연구를 단독으로 수행했기 때문에 이에 따른 교육 비용이 많이 들지는 않지만, 이를 수정하는 데에 따른 노력은 많이 필요하다는 사실을 알 수 있었다.

실제로 큰 규모의 어플리케이션을 도입하거나 특정 업체와의 계약을 통해서 이루어진다면 이에 따른 개발자 수도 점차 늘어나고 소프트웨어의 크기도 커지기 때문에 비용도 점점 증가할 것이다. 따라서 어플리케이션 소프트웨어의 경우는 이와 비슷한 다른 상용 소프트웨어를 구매해서 사용하는 것보다 초반에는 비용이 더 들지 몰라도, 장기적인 면으로 봤을 때에

는 OSS 를 도입하여 활용하는 것이 비용 절감을 가질 것이라 기대한다.

5. 결론

본 연구에서는 OSS 의 비용을 구하기 위해서 먼저 필요한 요소를 파악하고, 이에 따른 범위를 정하였다. OSS 의 활용 형태가 어플리케이션으로 한정하여, 이를 또한 오픈소스 커뮤니티 상에서의 도입으로 한정하여 비용을 구했는데, 이 때에 따른 비용 요소를 기본적인 노력에 따라서 구하였다.

본 연구에 제시된 비용 요소를 세 가지를 제시하였지만, 그것이 모든 비용 요소라 할 수는 없다. 따라서, OSS 의 비용을 산정할 때 OSS 에 대한 범위를 먼저 정한 후, 범위에 따른 비용 지표에 대한 연구가 우선적으로 이루어져야 할 것이다. OSS 의 형태 및 범위에 따른 비용 지표에 대한 연구가 이루어지게 되었을 때 OSS 의 비용 연구에 대한 체계가 확립된다면, OSS 의 전반적인 비용을 구해서 이의 올바른 활용도 도모할 수 있을 것이다.

참고문헌

- [1] Dan Woods, Gautam Guliani, Open Source for the Enterprise, O'Reilly, 2005. 7.
- [2] T. R. Madanmohan, Rahul De', "Open Source Reuse in Commercial Firms", IEEE. Volume 21, 2004, pp. 62-69.
- [3] J. Lemer, J Tirole, "The Economics of Technology sharing Open Source and Beyond", NBER Working Paper, No. 10956, 2004.
- [4] G. Antonioli, R. Fiutem, C. Lokan, "Object-Oriented Function Points: An Empirical Validation", Empirical Software Engineering, 8, 225-254, 2003.
- [5] Ya-lun Chou, Statistical Analysis, Holt International, 1975, ISBN 0030894220, section 17.9.
- [6] 한국소프트웨어진흥원, "공개소프트웨어 비용구조에 관한 연구", 2003.
- [7] 정낙수, 황선태, "소프트웨어 비용 절감을 위한 공개 소프트웨어 의 적용절차", 한국컴퓨터종합학술대회 논문집 Vol. 32, No. 1, 2005, pp. 451-453
- [8] 김종배, 김두연, 류성열, "오픈소스 컴포넌트 활용에 있어서의 소프트웨어 개발방법론 적용에 관한 탐색적 연구: 마르미-III 방법론을 이용한 웹포털 개발 사례를 중심으로", 韓國 SI 學會誌, 第 5 卷 第 1 號, 2006. 5, pp. 149-164.
- [9] 서정석, 김승렬, "컴포넌트 개발 틀을 사용한 소프트웨어 개발 노력도에 관한 연구", 한국산업정보학회논문지 제 5 권 제 3 호, 2000. 9.
- [10] BORG Calendar, <http://www.mbcsoft.com>