

# 정보과학회 컴퓨팅의 실제 논문지

## KIISE Transactions on Computing Practices

VOLUME 23, NUMBER 9, SEPTEMBER 2017

---

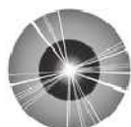
이기종 사물들의 효과적 동작을 위한 사물인터넷 미들웨어 .....	전수빈, 한영탁, 이충산, 서동만, 정인범	517
재사용 모듈 추출을 위한 오픈 소스 기반 소프트웨어 시스템 구축 .....	변은영, 박보경, 장우성, 김영철, 손현승	535
계층적 포인터 네트워크를 이용한 상호참조해결 .....	박천음, 이창기	542
IoT 태그 및 소셜 메시지 기반 사물 제어 시스템 .....	백승민, 진연주, 하권우, 한상욱, 정진우	550
안정적 유전자 특징 선택을 위한 유전자 발현량 데이터의 부트스트랩 기반 Lasso 회귀 분석 .....	조정희, 윤성로	557
캐시 이미지의 동적 관리 방법을 이용한 명령어 캐시 성능 개선 .....	서효중	564

---

### 단편 논문

가상화 환경에서 네트워크 I/O 성능 간섭 해결을 위한 .....	고현석, 이경운, 박현찬, 유 혁	572
피드백 제어 CPU 스케줄링 기법 분석		

---



## 한국정보과학회

KOREAN INSTITUTE OF INFORMATION SCIENTISTS AND ENGINEERS

# 재사용 모듈 추출을 위한 오픈 소스 기반 소프트웨어 시스템 구축 (Constructing an Open Source Based Software System for Reusable Module Extraction)

변은영<sup>†</sup> 박보경<sup>†</sup> 장우성<sup>†</sup> 김영철<sup>\*\*</sup> 손현승<sup>\*\*\*</sup>  
(Eun Young Byun) (Bokyung Park) (Woosung Jang) (R. Young Chul Kim) (Hyun Seung Son)

**요약** 소프트웨어 시장 규모가 확대되면서 다양한 요구사항을 만족시키는 대규모 소프트웨어가 개발되고 있다. 이로 인해 소프트웨어 복잡도가 증가하게 되고 품질 관리는 어려워졌다. 특히, 레거시 시스템의 개선 및 새로운 시스템 개발 환경에서 재사용은 중요하다. 이 논문에서는 품질을 인증 받은 모듈을 재사용하는 방법을 제안한다. 재사용 적용 레벨은 코드 영역(메소드, 클래스, 컴포넌트), 프로젝트 도메인, 비즈니스로 나누어진다. 이 논문에서는 소프트웨어 복잡성을 측정하는 결합도와 응집도 기반의 재사용 메트릭과 메소드와 클래스 레벨에 따라 “재사용에 적합한 모듈 덩어리”를 시각화하는 재사용 적합 모듈 추출 메카니즘을 제안한다. 레거시 프로젝트에 역공학 기법을 적용하여 어떤 모듈/객체/덩어리가 재사용할 수 있는지를 식별하고 확장 시스템을 개발하거나 유사한 새로운 시스템을 개발하기 위해 재사용한다면 소프트웨어의 신뢰성을 보장하고 소프트웨어 개발 단계에서 필요한 시간과 비용을 절감시킬 수 있다.

**키워드:** 소프트웨어 재사용, 재사용 메트릭, 소프트웨어 고품질, 가시화, 자동화 시스템

**Abstract** Today, the scale of the computer software market has increased, and massive sized software has been developed to satisfy diverse requirements. In this context, software complexity is increasing and the quality of software is becoming more difficult to manage. In particular, software reuse is important for the improvement of the environments of legacy systems and new system development. In this paper, we propose a method to reuse modules that are certified by quality. Reusable levels are divided into code area (method, class, and component), project domain, and business levels. Based on the coupling and cohesion of software complexity, we propose a reusable module extraction mechanism with reusability metrics, which constructs a visualization of the “reusable module’s chunk” based on the method and class levels. By applying reverse engineering to legacy projects, it is possible to identify reusable modules/objects/chunks. If these modules/objects/chunks are to be reused to develop an extension system or similar new system, we need to ensure software reliability in order to reduce the time and cost of software development.

**Keywords:** software reuse, reusability metrics, software high quality, visualization, automatical system

· 이 논문은 2015년 교육부와 한국연구재단의 지역혁신장인력양성사업의 지원을 받아 수행된 연구임 (NRF-2015H1C1A1035548)

· 이 논문은 제43회 동계학술발표회에서 '가치 있는 모듈 식별을 위한 오픈 소스 기반 소프트웨어 현대화 시스템 구축'의 제목으로 발표된 논문을 확장한 것이다

<sup>†</sup> 학생회원 : 홍익대학교 소프트웨어공학연구소  
eybyun@selab.hongik.ac.kr  
park@selab.hongik.ac.kr  
jang@selab.hongik.ac.kr

<sup>\*\*</sup> 정회원 : 홍익대학교 컴퓨터정보통신공학과 교수  
bob@selab.hongik.ac.kr

<sup>\*\*\*</sup> 정회원 : 홍익대학교 소프트웨어공학연구소  
(Hongik Univ.)  
son@selab.hongik.ac.kr  
(Corresponding author임)

논문접수 : 2017년 3월 8일  
(Received 8 March 2017)

논문수정 : 2017년 6월 29일  
(Revised 29 June 2017)

심사완료 : 2017년 7월 4일  
(Accepted 4 July 2017)

Copyright©2017 한국정보과학회 : 개인 목적이나 교육 목적인 경우, 이 저작물의 전체 또는 일부에 대한 복사본 혹은 디지털 사본의 제작을 허가합니다. 이 때, 사본은 상업적 수단으로 사용할 수 없으며 첫 페이지에 본 문구와 출처를 반드시 명시해야 합니다. 이 외의 목적으로 복제, 배포, 출판, 전송 등 모든 유형의 사용행위를 하는 경우에 대하여는 사전에 허가를 얻고 비용을 지불해야 합니다.  
정보과학회 컴퓨팅의 실제 논문지 제23권 제9호(2017. 9)

### 1. 서론

현대 소프트웨어는 IT를 포함한 사회·문화 전반의 다양한 분야에서 사용되고 있다. 사용 범위가 확장되면서 사용자의 요구사항 또한 다양화되었고 이를 만족하는 소프트웨어의 복잡도는 증가했다. 따라서 개발에 더 많은 시간과 비용이 필요할 뿐만 아니라 유지보수 시에도 어려움을 겪고 있다[1]. 이런 문제를 해결하기 위해서 SP인증, 소프트웨어 개발 방법론, 테스트 등의 많은 연구들이 진행되고 있다[2]. 그 중에서도 재사용은 레거시 시스템에서 품질을 인정받은 모듈을 재사용함으로써 품질을 향상시키고 소프트웨어의 신뢰성을 높일 수 있다. 뿐만 아니라 개발에 필요한 시간과 비용을 절감시킬 수 있는 중요한 방법이다.

그림 1은 재사용의 적용 레벨이다. 코드 개발(코드, 객체, 컴포넌트), 프로젝트 도메인(웹 서비스, 패턴, 프레임 워크), 비즈니스(SOA, 어플리케이션 디자인) 세 가지로 나누고 각 레벨의 하위 요소들이 존재한다. 상위 레벨로 올라갈수록 재사용의 효과가 크고 투자 수익도 증가하게 된다[3].

이 논문은 응집도와 결합도를 사용한 재사용 메트릭을 정의하고 이를 사용하여 재사용 모듈을 자동식별하고 가

시화하는 시스템을 구축한다. 이를 통해 대규모 소프트웨어의 복잡한 내부 구조의 이해를 돕고, 재사용 모듈/덩어리 식별을 통해 재사용성을 높여 품질을 개선한다. 다음 장은 관련 연구로 기존 연구에 대해 설명한다. 3장은 메소드와 클래스 레벨에서의 재사용을 설명한다. 4장은 재사용 모듈 식별 시스템의 구조와 가시화 결과에 대해 기술한다. 마지막으로 5장은 결론 및 향후 연구를 언급한다.

### 2. 관련 연구

재사용성을 측정하기 위한 많은 연구들을 진행되었다[4-6]. 가장 대표적인 재사용성 측정 방법은 CK(Chidamber & Kemerer) 메트릭이다. 이는 객체 지향 소프트웨어에서 복잡도, 재사용성, 모듈화, 캡슐화를 측정하는 여러 요소들로 구성된다. 구체적으로, WMC(Weighted Methods per Class)는 한 클래스의 메소드의 수, DIT(Depth of Inheritance Tree)는 상속 계층의 깊이, NOC(Number of Children)는 한 클래스를 상속받은 하위 클래스들의 수, CBO(Coupling Between Objects)는 하나의 클래스가 다른 클래스의 변수에 접근하거나 메소드를 호출하는 경우의 수, RFC(Response for Class)는 하나의 클래스에 있는 메소드의 수와 그 메소드들과 호출 관계에 있는 메소드들의 수를 더한 값, LCOM(Lack of Cohesion of Methods)는 한 클래스에서 메소드들이 속성을 공통으로 참조하는 경우의 수를 측정한 값이다[7-10].

표 1은 CK메트릭 가이드라인이다. 각 요소들과 복잡도, 재사용성, 모듈화, 캡슐화의 상관관계를 나타낸다. CK메트릭의 CBO와 LCOM은 재사용성과의 관계를 표기하지 않았지만 모듈화와 재사용성이 상관관계를 갖기 때문에 이들도 재사용성에 영향을 준다고 볼 수 있다.

CK메트릭을 활용한 기존 연구들은 다음과 같다. Zahara[7]는 재사용성을 측정하기 위해 결합도, 응집도, 상속을 고려한다. CK메트릭의 CBO, LCOM, DIT를 측정하고 4가지의 회귀 알고리즘인 다중 선형 회귀, 모델 트리 M5P, 표준 인스턴스 기반 학습 스키마 IBk, 메타 학습 스키마 Additive 회귀를 적용했다. Shri[8]는 객체 지향

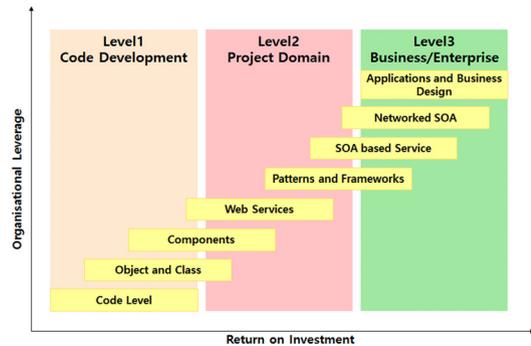


그림 1 재사용의 적용 레벨[3]  
Fig. 1 Level of Reuse

표 1 CK 메트릭 가이드라인[10]

Table 1 CK Metrics guidelines

Metric	Goal	Level	Complexity (To develop, to test and to maintain)	Reusability	Encapsulation, Modularity
WMC	Low	▼	▼	▲	
DIT	Trade-off	▼	▼	▼	
		▲	▲	▲	
NOC	Trade-off	▼	▼	▼	
		▲	▲	▲	
CBO	Low	▼	▼		▲
RFC	Low	▼	▼		
LCOM	Low	▼	▼		▲

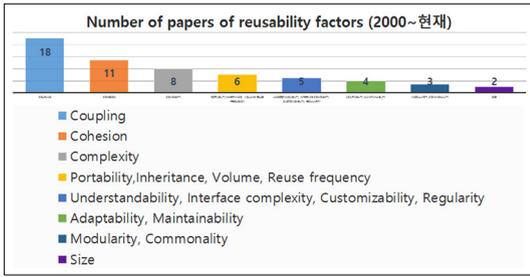


그림 2 재사용성 요소의 적용 논문 현황

Fig. 2 Applied paper Status of Reusability factor[11]

소프트웨어의 재사용성을 예측하기 위해 CK메트릭, K-means 알고리즘 및 의사 결정 접근법을 사용한다. Sandhu[9]는 McCabe의 Cyclomatic 복잡도 메트릭, 규칙성 메트릭, Halstead 소프트웨어 지표, 재사용 빈도 메트릭, CK 메트릭으로 재사용성 평가 시스템을 구축했다. 이와 같이 재사용성을 측정하기 위해서는 어떤 요소를 어떤 메트릭을 사용하여 측정할지가 중요하다.

그림 2는 CK 메트릭을 포함하여 재사용성 측정에 사용되는 요소들이 기존 논문들에서 적용된 현황이다. 가장 많이 사용된 요소는 결합도, 응집도이다. 하지만, CK 메트릭은 여러 종류로 나누어지고 종류별 세기도 다른 결합도와 응집도를 고려하지 않는다. 따라서 이 논문에서는 종류별 결합도와 응집도를 측정함으로써 재사용 메트릭을 개선하고자 한다.

### 3. 재사용 레벨

#### 3.1 재사용 메트릭

표 2는 이전 연구에서 응집도와 결합도의 종류별 수치를 기반으로 정의한 재사용화 성숙도 메트릭이다[12]. 응집도가 강할수록 재사용에 적합하기 때문에 강할수록 높은 점수를 갖고, 결합도는 약할수록 재사용에 적합하기 때문에 약할수록 높은 점수를 갖는다. 응집도와 결합도의 종류별 수치를  $ce$ ,  $co$ 로 각각 평균화하고, 재사용

표 2 재사용화 성숙도 메트릭[12]

Table 2 Reusability Metrics

$$\sum_{i=0}^n ce = ce_f + ce_s + ce_m + ce_p + ce_t + ce_l + ce_c \quad \dots\textcircled{1} \text{ Sum of Cohesion}$$

$$\sum_{j=0}^m co = co_d + co_s + co_c + co_e + co_m + co_n \quad \dots\textcircled{2} \text{ Sum of Coupling}$$

$$R = \frac{1}{2} \left( \frac{\sum_{i=0}^n ce_i}{n} + \frac{\sum_{j=0}^m co_j}{m} \right) \quad (R > 4) \quad \dots\textcircled{3} \text{ Reusability Metrics}$$

수치인  $R$ 을  $ce$ 와  $co$ 의 평균으로 정의한다.  $R$ 이 점수의 전체 범위의 중간 수치인 4보다 클 경우 재사용에 적합한 모듈로 가정한다.

#### 3.2 메소드-클래스 레벨

메소드 레벨에서의 재사용은 내부 응집도가 높고 다른 메소드와의 결합도가 낮을 경우 적합하다. 그림 3은 메소드 레벨에서의 응집도와 결합도이다. 메소드 안에 요소(A,B,C...)들은 명령어를 의미한다. method1, 4는 내부 요소들 간에 응집도와 서로 간의 결합도가 있다. 이 경우 method1과 method4를 복합 모듈로 재사용하는 것이 효율적이다. 클래스 레벨에서도 메소드와 마찬가지로 내부 응집도가 높고 다른 클래스와의 결합도가 낮을 경우 적합하다.

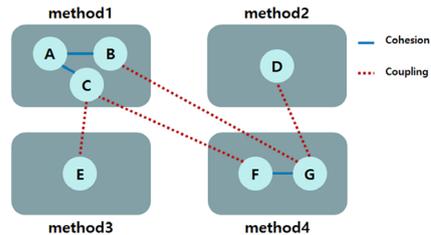


그림 3 메소드 레벨에서의 응집도와 결합도

Fig. 3 Cohesion and Coupling of Method Level

### 4. 재사용 모듈 식별 시스템 구축

#### 4.1 시스템 구성도

소스 코드에서 재사용에 적합한 모듈(클래스, 메소드)을 가시화하기 위한 시스템을 구축한다. 그림 4는 시스템의 전체적인 구조로 총 5단계로 구분할 수 있다. 재사용성 측정을 위해 응집도와 결합도를 적용하고 빈도수와 재사용 체인(복합 모듈)을 고려한다.

- Step1 (Code Input) : 코드 입력 단계는 가시화의 대상이 되는 소스 코드를 xCodeParser에 입력한다. 이 논문에서는 객체 지향 언어인 Java 기반의 코드를 입력한다.
- Step2 (Source Analysis) : 소스 분석 단계는 입력된 코드를 xCodeParser를 사용해 분석한다. 자바로 작성된 소스 코드인 \*.java파일의 구문 분석을 통해 ASTM을 생성한다. 각 java파일마다 ASTM이 추출되고 확장자는 \*.sastm이다.
- Step3 (DB save) : 데이터베이스 저장 단계는 Step2에서 분석된 정보인 ASTM 데이터를 구조화하여 데이터베이스에 저장한다. 변수와 메소드의 정보는 Component, 메소드 간의 호출 관계와 객체 생성 정보는 Link, 메소드의 응집도와 결합도는 Cohesion, Coupling으로 구조화하여 저장한다. 데이터베이스는 서버가 필요하지 않은 간단한 용도이기 때문에 가벼운 데이터베이스인 SQLite를 사용한다.

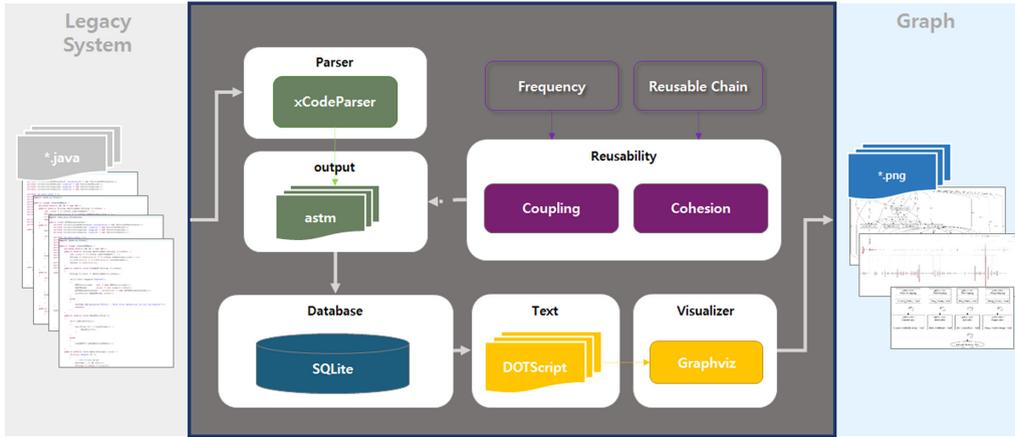


그림 4 시스템 구성도  
Fig. 4 System Structure

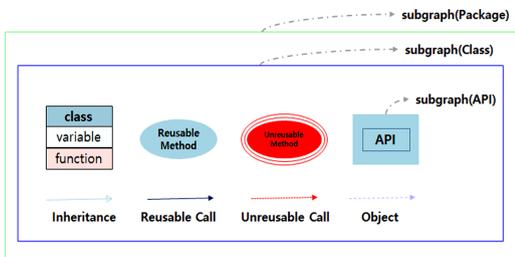


그림 5 가시화 노드 & 관계 종류  
Fig. 5 Type of Visualization Node and Relation

- Step4 (Analysis of Structure) : 구조 분석 단계는 데이터베이스에 저장된 데이터를 클래스와 메소드 레벨로 구분하여 DOT Script로 재해석한다.
- Step5 (Visualization) : 가시화 단계는 Step4에서 재해석한 DOT Script를 오픈 소스인 Graphviz를 이용해 가시화한다. 클래스와 메소드 레벨을 모두 가시화한 그림과 클래스 레벨과 메소드 레벨을 각각 가시화한 그림이 생성된다.

**4.2 재사용 모듈의 자동식별 가시화**

가시화 결과는 그림 파일(.png)로 추출된다. 모든 클래스와 메소드를 노드로 가시화한다. 클래스 노드에서 내부의 구조 정보(변수, 메소드)를 나타낸다. 컴포넌트(클래스, 메소드)간의 관계(상속, 호출, 객체 생성 관계)를 화살표로 가시화한다. 그림 5는 가시화 노드, 관계의 종류를 나타낸다. 패키지(Green)와 패키지 내부의 클래스(Blue)를 서브 그래프로 표현한다. 클래스를 노드로 표현할 때는 내부의 구조를 나타내는 테이블 형식을 사용하고 선언된 변수와 메소드의 정보를 갖는다. 메소드는 타원형 노드로 나타내는데 재사용이 적합한 메소드는 하늘색, 부적합한 메소드는 빨간색 삼중타원으로 표시된다.

모든 메소드 노드 내부에는 응집도 수치가 표기한다. 사용된 외부 API는 서브 그래프로 묶어서 표현한다. 컴포넌트(클래스, 메소드, API)간의 관계는 상속, 호출, 객체 생성 관계로 구분할 수 있다. 재사용에 적합한 호출 관계는 남색 실선으로 표시되고, 부적절한 호출은 빨간색 점선으로 표시된다. 모든 호출 관계 화살표에는 결합도 수치와 호출 빈도수가 표기된다.

타겟 코드는 자바로 작성된 M-PVMS(Metamodel based Photovoltaic Monitoring System)이다. 이는 클라이언트와 서버간의 데이터 통신을 수행하는 태양광 모니터링 시스템이다[13]. 가시화를 통해 복잡한 프로그램의 구조가 어떤 컴포넌트들로 구성되어 있고, 컴포넌트들이 어떤 관계로 이루어져 있는지 확인할 수 있다.

**4.2.1 메소드 레벨 가시화**

그림 6은 메소드 레벨에서 가시화한 그래프이다. 재사용에 부적합한 메소드와 호출 관계는 빨간색으로 표시하여 가시성을 높였다. 단일 모듈에서 update 메소드와 getValue 메소드는 재사용 메트릭에 의해 재사용하기에 적합한 모듈로 식별된다. 빈도수를 확인해 보면 update 메소드는 4번, getvalue 메소드는 1번 호출된다. 따라서, 호출 횟수가 더 많은 update 메소드가 재사용에 더 적합하다는 것을 알 수 있다. 복합 모듈은 재사용에 적합한 단일 모듈과 해당 모듈과 결합도가 있는 다른 모듈을 연결하여 재사용한다. 복합 모듈에서 initInverterData 메소드는 재사용 메트릭에 의해 재사용에 적합한 모듈로 식별된다. 이 메소드는 setErrorData 메소드와 높은 결합도를 갖는다. 따라서 initInverterData 메소드를 재사용할 때 이 메소드를 연결하여 복합 모듈로 재사용해야 효율적이다.



#### 4.2.2 클래스 레벨 가시화

그림 7은 클래스 레벨에서 가시화한 그래프이다. 재사용에 적합하지 않은 낮은 응집도를 가진 클래스 내의 메소드와 높은 결합도의 클래스 간의 호출은 빨간색으로 표시하여 가시성을 높였다. 단일 모듈은 해당 모듈 내부의 응집도와 외부 모듈과의 결합도를 확인하여 식별할 수 있다. CRC16Modbus 클래스는 재사용 메트릭에 의해 재사용하기에 적합한 모듈로 식별된다. 복합 모듈의 경우에 WillingProtocolParser 클래스는 결합되어 있는 InverterData 클래스를 13번 호출한다. 결합도가 높고 빈도수가 많은 두 메소드를 연결하여 복합 모듈로 재사용이 가능하다.

### 5. 결론 및 향후 연구

본 논문은 소프트웨어 품질을 높이기 위한 다양한 방법 중에 재사용성에 대한 연구했다. 기존에는 재사용성 측정의 대표적 요소인 응집도와 결합도를 CK메트릭을 사용했다. 하지만 이 메트릭은 다양한 응집도와 결합도의 종류를 구분하지 않고 대략적인 수치만을 제공했다. 이 문제를 해결하기 위해 실제 응집도와 결합도의 종류별 수치를 측정하는 재사용 메트릭을 정의한다. 이를 기반으로 레거시 시스템에 역공학 기법을 적용하여 메소드/클래스 레벨에서의 재사용 모듈을 자동식별하고 가시화 하는 시스템을 구축했다. 이를 통해서 레거시 시스템 모듈을 확장 시스템 및 유사한 시스템 개발 시에 재사용할 수 있고, 이를 통해 품질을 향상 시키고, 개발 시간과 비용을 절감 시킬 수 있다. 향후에는 동적 분석을 통하여 재사용 메트릭을 개선하고, 재사용 레벨을 상위 레벨로 확장하여 연구할 예정이다.

### References

- [1] NIPA, "SW Development Quality Management Manual," 2013.
- [2] Stephen H. Kan, "Metrics and models in software Quality Engineering," Addison-Wesley Professional, 2003.
- [3] W Frakes and C. Terry, "Software reuse: metrics and models," *ACM Computing Surveys*, Vol. 28, No. 2, pp. 415-435, Jun. 1996.
- [4] Bojana Koteska and Goran Velinov, "Component-Based Development: A Unified model of Reusability Metrics," *ICT Innovations 2012*, pp. 335-344, 2012. (in Berlin)
- [5] Vijai Kumar, Arun Sharma, Rajesh Kumar, and P.S. Grover, "Quality aspects for component based systems: A metrics based approach," *Software : Practice and Experience*, Vol. 42, No. 12, pp. 1531-1548, Dec. 2012.

- [6] V. Lee, "Automated source code measurement environment for software quality," *Doctoral, Auburn University, Auburn, USA*, Dec. 2007.
- [7] S. I. Zahara, M. Ilias, and T. Zia, "A study of comparative analysis of regression algorithms for reusability evaluation of object oriented based software components," *International Conference on Open Source Systems and Technologies*, Dec. 2013.
- [8] A. Shri, P. S. Sandhu, V. Gupta, and S. Anand, "Prediction of reusability of object oriented software systems using clustering approach," *World Acad. Sci. Eng.*, Vol. 4, No. 7, pp. 853-856, 2010.
- [9] P. Sandhu and H. Singh, "A Neuro-Fuzzy Based Software Reusability Evaluation System with Optimized Rule Selection," *ICET '06 International Conference on Emerging Technologies*, Vol. 2, pp. 664-669, Nov. 2006.
- [10] Chidamber Shyam and kemerer Chris, "A metrics suite for object oriented design," *IEEE Transactions on Software Engineering*, Vol. 20, No. 6, Jun. 1994.
- [11] Marko Mijac and Zlatko Stapic, "Reusability Metrics of Software Components: Survey," *Central European Conference on Information and Intelligent Systems*, pp. 221-231, Sep. 2015.
- [12] Eun Young Byun, Bo Kyung Par, Woo Sung Jang and R. Young-Chul Kim, "Constructing software Modernization System based on Open source software for a valuable module identification," *KIISE*, pp. 404-406, Dec. 2016.
- [13] Hyun Seung Son and R. Young Chul Kim, "Modeling a Photovoltaic Monitoring System based on Maintenance Perspective for New&Renewable Energy," *International Joint Conference on Convergence*, pp. 144-147, Jan. 2016.



변 은 영

2016년 홍익대학교 컴퓨터정보통신(학사)  
2016~현재 홍익대학교 소프트웨어공학 석사과정. 관심분야는 소프트웨어 자동화 도구 개발, 소프트웨어 프로세스 및 가시화, 소프트웨어 역공학, 소프트웨어 품질 및 재사용성



박 보 경

2008~2012년 홍익대학교 컴퓨터정보통신(학/석사). 2012년~현재 홍익대학교 일반대학원 소프트웨어공학 박사과정. 관심분야는 소프트웨어공학, 요구공학, 역공학, 테스트 성숙도 모델(TMM)



장 우 성

2011년 홍익대학교 소프트웨어공학(석사)  
2015년~현재 홍익대학교 소프트웨어공학  
박사과정. 관심분야는 소프트웨어 공학, 임  
베디드 소프트웨어, 소프트웨어 테스트, 테  
스트 성숙도 모델(TMM), 소프트웨어 역  
공학, 클라우드 시스템 개발



김 영 철

2000년 LG산전 중앙연구소 Embedded  
System 부장. 2010년~현재 홍익대학교  
컴퓨터정보통신공학 교수. 관심분야는 테  
스트 성숙도 모델(TMM), 임베디드 소프  
트웨어 개발 방법론, 모델 기반 테스트,  
메타모델, 비즈니스 프로세스 모델, 신재

생에너지 통합관리 시스템



손 현 승

1999년~2015년 홍익대학교 소프트웨어공  
학(학/석/박사). 2015년~현재 홍익대학교  
메카트로닉스연구센터 박사후연구원. 관심  
분야는 임베디드 소프트웨어 자동화 도구  
개발, 소프트웨어 프로세스 및 가시화, 메  
타모델 설계 및 모델 변환, 모델 검증 기법

연구, 한국형 테스트 성숙도 모델, 시큐어 코딩