Advanced Engineering Forum *Online: 2012-09-26*
ISSN: 2234-991X, Vols. 6-7, pp 333-336
doi:10.4028/www.scientific.net/AEF.6-7.333

Improved CK Metrics for Object Oriented Design Measurement

Zhenqi Wang[1, a], Dankai Zhang[1,b]

[1]Information & Network Management Center ,North China Electric Power University,Baoding, China

[a]w-zhenqi@ncepubd.edu.cn, [b]zhangdankai@163.com

Keywords: software metrics; object-oriented; Java

Abstract. In order to understand and control the software development process better, improve the quality of software. At the same time, traditional methods of measurement in the field of object-oriented is no longer fit for some of the unique features for object-oriented software, this makes us accelerate the speed of research of object-oriented software metrics, and considerable progress has been made.

This paper describes object-oriented software measurement method proposed by Chidamber and Kemerer-- C&K , for its lack, we improved on method of C&K, combined with the features of the Java language, applied software measurement tools that you already have on software measurement indicators for implementation and help developers guide software development processes to meet the needs of users better.

Introduction

With the rapid development of software development technology, software metrics take more and more people's attention. Object-oriented software methodologies introduced many new features, such as inheritance, encapsulation, polymorphism and so on. But due to the large number of measurement methods and object-oriented software development, and required metrics are not the same, so object-oriented software metrics is a challenge.

Current methods of object-oriented software metrics including C&K, MOOD, LK, and forecast objects, and so on. C&K method is the most commonly used method which have 6 metrics, these six indicators more fully describe the features of object-oriented technology, WMC, CBO measurement class complexity, DIT and NOC class inheritance of measurement, metrics provide basis for reuse of program, RFC measure such responses to messages, LCOM metrics of cohesion of the class. Although these six indicators are making considerable contributions, there is insufficient, there are areas requiring improvement.

C&K METRIC SUITE

（1）the synopsis of C&K

metric 1：Weighted Methods per Class，WMC

Class C has methods M_1 M_2...M_n,every method" set a complexity" as C_1 C_2...C_n ,then the metric formula is defined as:

$$WMC = \sum_{i=1}^{n} C_i \qquad (1)$$

value of WMC larger the methods of a class, the more greater complexity, time required to develop and maintain such difficulty and more.

metric 2： Depth of Inheritance Tree，DIT

$$DIT = \begin{cases} 0, no \quad inherit \\ \max(DIT(I_k)) + 1, \ others \end{cases}, I_k = \text{the number of ancestor of the class} \qquad (2)$$

DIT may affect the number of ancestor of the class. DIT larger values, methods may inherit several more, more reuse, design and more complex. In the Java language, there are only single inheritance, a class can have only one parent class.

metric 3： Number Of Children，NOC

NOC indicates that a number of such direct successors. Larger values of the NOC, the greater the number of their children, and reusable as possible, abstract, the weak, the greater impact on the whole system.

metric 4： Coupling Between Object classes，CBO

Set class C1= 〈x p(x)〉 C2= 〈y p(y)〉 ， p(x)=$\{M_X\} \cup \{I_X\}$ p(y)=$\{M_Y\} \cup \{I_Y\}$

$\{M_X\}$， $\{I_X\}$ are the method and property members of objects C_1.Any $\{M_X\}$effect on$\{M_Y\}$or$\{I_Y\}$ constitute coupled relationship. Larger values of the CBO, the other class larger, greater independence, and poor maintenance. In a Java program, coupling between object classes, refers to a class called child class inherits and implements the interface.

metric 5： Response For a Class，RFC

RFC indicates class size of response set. Larger values of the RFC, such methods more, more complex, more complex testing. In order to measure the complexity class, RFC response collection also includes Java programs used by the custom class to another custom method to a class method and system.

metric 6： Lack of Cohesion in Methods，LCOM [1]

Set class C have methodsM1M2......M_n ,Mi use istance of class variable sets I_i :

$$P = \{I_i， I_j | I_i \cap I_j = \varnothing, i \neq j\}; \quad Q = \{I_i， I_j | I_i \cap I_j \neq \varnothing, i \neq j\};$$

Or

$$LCOM = \begin{cases} |P| - |Q| & |P| > |Q| \\ 0 & others \end{cases} \qquad (3)$$

cohesion of the class is a description by "partial methods associated with a local instance variable in the class how close" to describe. LCOM value larger the subclasses of class description can be split into more.

（2）Research and analyse of the improvement C&K

C&K measurement methods is measured against the class layers, usually class layers with 5 layers: coupled systems, use, inheritance, classes, methods, study finds C&K measurement methods have the following issues were not taken into account:

①Not covered instances of the class. Actually only instantiated as objects that can be used, and number of instantiated times of a class is different. Clearly weighted method does not take into account the number of classes.

②Does not involve such polymorphism. Polymorphism is one of the important characteristics of a class, however C&K method without measuring polymorphism of a class make the appropriate measurements.

Therefore, introduces the following two measures improved C&K method which does not take into account some of the issues.

metric7：Complexity per Instance of Class,CIC

Set of instance class has methods M1 M2...Mn, corresponding method of "complexity" as C_1 C_2..C_n, instantiate a number N, the metric formula is defined as:

$$C_{CIC} = \begin{cases} \beta\sum_{i=1}^{n} M_i + \gamma\sum_{i=1}^{n} C_i, n = 1 \\ \alpha N + \beta\sum_{i=1}^{n} M_i + \gamma\sum_{i=1}^{n} C_i, n > 1 \end{cases} ; \quad \alpha + \beta + \gamma = 1， \text{are adjustment coefficient} \qquad (4)$$

Take full account of the number of times of a instantiated class, the times of methods and complexity, link instantiated class with the program's complexity, length, and other properties overall; adjustment coefficient could reflect the differences in effects on the length and complexity of the program; as one of the members of the reference number, you can increase the measure of nature of intuitive and detailed.

The greater complexity of a instantiated class, the longer the procedure is more complex, more difficult to develop and maintain. Greater complexity, the possibility of error is also bigger, the focus of the class should be used as test.

metric 8： Polymorphism of System，PS

In the Java , System polymorphism is the maximum number of classes of polymorphism of the actual number and of the possible number.

N_1={ polymorphism of the actual number} N_2={ polymorphism of the possible number}

$PS = \alpha \cdot N_1 + \beta \cdot N_2$, α，$\beta \in [0,1]$ are adjustment coefficient，first set $\alpha = \beta = 0.5$ （5）

Polymorphism is another feature of object-oriented, it allows a name has multiple semantics. Program can send a message to an object to complete a series of actions, does not need to be concerned about how to implement, in favour of software implementation and modify. Polymorphism performance object has more than one class, so it is related to the static type and associated with the dynamic type, too many dynamic is not conducive to the process of understanding and reliability. Use polymorphism is another important element of object-oriented software metrics.

Realize the metric tool through improved C&K

Software is running on platform Windows XP, select Java launched in 1995 by Sun Microsystems Corporation to develop measurement tools.

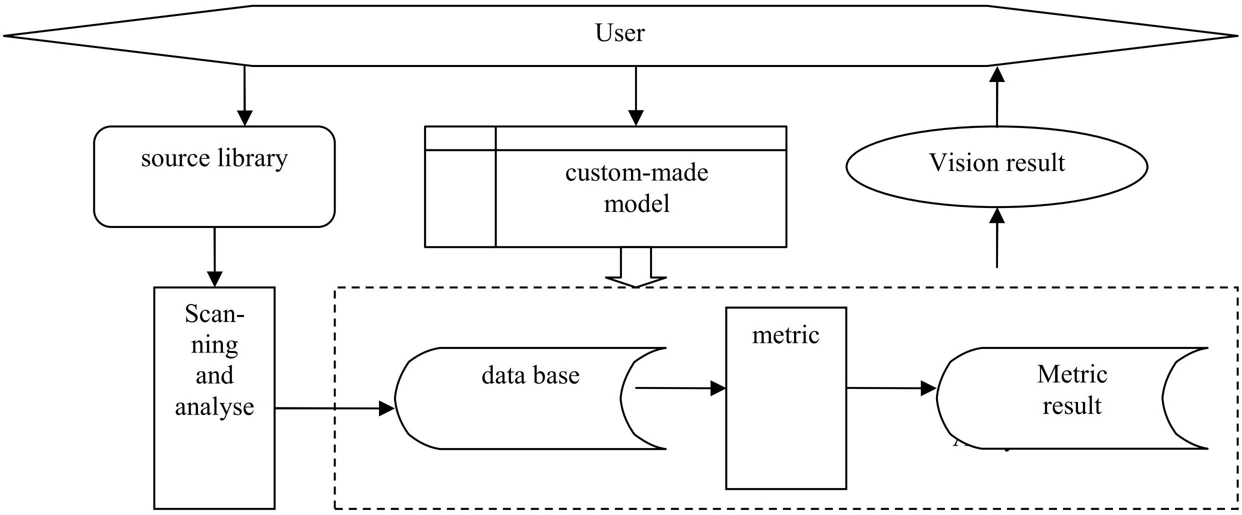

Fig 1. Metrics structure

The structure of software measurement system can be divided into six parts, as shown in Figure1. Scan analysis, databases, models, measurement analysis, results, view display.

Select a source programs or a set programs, scanning analysis, database, dynamic description of schedule progress, prompting after the end of analysis, then modeling, and general support "factor-criteria-metrics" model, calculated metrics[3], results into a repository. According to user's needs, choose metrics, such as forms, source code chart shows a list of results. Main steps:

（1）Scan analysis layer in the front, formed by the lexical analyzer and parser, function: ① analyze source code; ② gather information related metrics; ③ put information to the database.

（2）Source code information for the database is used to store calculated metrics, change different object-oriented language source of useful information into a unified form. Using XML as a database[3], you must put measure information which you needs into XML repository[3]: ① Select program directory, extract the .jaya file name. ②obtained .jaya's output stream (BufferedReader), incoming parser returns the List collection classes, store XClass object the collection, information extracted from the object class. ③ XClass incoming a class XML create (XmlFileCreate), the class give back a certain format Document by passing information, use jdom to build the database. ④ put the Document into the XML file, stored in the program directory xmlfile's folder. ⑤ historical file management.

（3）Analysis of measurement is based on the metric formula, reads necessary information from the database, calculates the correspond measure index, then put the value into metrics database.

Analysis of Measurement Date

Use traditional methods of C&K and improved C&K method, respectively scan the source program, calculate metrics, and calculate the correspond average. As shown in Figure 2.

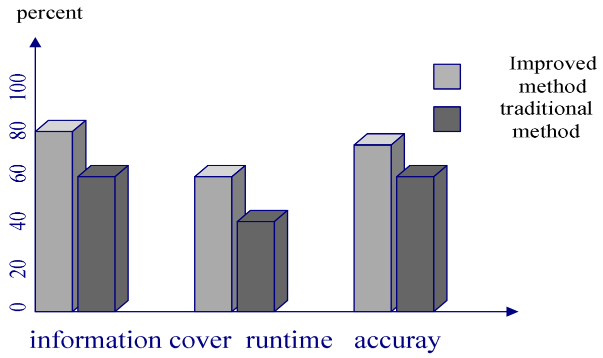

Fig. 2 Traditional methods of C&K and improved C&K method by comparison

From the top figure, information on cover and data accuracy, improve C&K is significantly higher than traditional C&K method, but the improved C&K method needs significantly longer time than the traditional method of C&K. Study on object-oriented software metrics has a broad application prospects, but currently, it is only in the discussion phase, there is no theoretical and practical value of the measurement methods and standards through mining features of object-oriented software. Above metrics only evaluated about some aspects of object-oriented technology in the measurement of a specific internal properties, they cannot be linked with the complexity of the entire system.

References

[1] Mr. U. L. Kulkarni, Mr. Y. R. Kalshetty, Ms. Vrushali G. Arde. Validation of CK metrics for Object Oriented Design Measurement. Third International Conference on Emerging Trends in Engineering and Technology.October 2011 pp:646-651

[2] Zhangyan,Linying,Wanghong song. Metrics for Object-Oriented Systems.Computer and digitial engineering. 2009 .

[3] Li Xiao, Wu Jing. Research on object-oriented software metrics and tool design. Wanfang data.