

基于量化指标分析的软件质量度量方法

程 诺 万 琳 张 威

(装甲兵工程学院信息工程系, 北京 100072)

摘 要: 使用定量的软件质量度量能够为质量管理人员的决策提供数据的基础,并减少定性评价或者主观评价的片面性。本文基于 ISO/IEC9126 软件质量标准,在分析软件质量度量的量化指标的基础上,进行了软件质量度量方法的研究,完善了软件质量综合特性和定量指标的度量模型。

关键词: 软件质量; 度量模型; 度量指标模型

中图分类号: TP311.52

引 言

软件度量工作自 1958 年 Rubey 和 Hurtwick 首次提出软件度量学概念^[1]至今,大体经历了三个阶段。1976 年,Boehm 等人从可移植性、可使用性、可维护性三个方面来定量的评价软件质量^[2]为第一阶段;第二阶段以 1978 年 McCall 等人从“产品运行、产品修正和产品转移”提出的质量度量模型即“质量要素——评价准则——度量”为标志。但是实践证明通过该模型方式不同缺陷的反馈信息可能相同,以致指标的制定及其定量的结果难以评价^[3]。随后,1984 年 Basili 提出了 GQM (Goal-Question-Metric) 范例度量模型,将测量与项目和过程的总体目标联系在一起,但 GQM 没有涉及测量标度、客观性和可行性等问题^[1]。国际标准化组织 (ISO)^[4]于 1985 年和 1993 年先后提出的多项关于软件质量度量技术的工作报告,并提出的软件质量度量模型将软件度量工作带入了第 3 个发展阶段。

1 软件度量过程

软件度量过程大体可以分为 4 步:确定度量需求,设计度量,执行软件度量,分析度量结果。确定度量需求是有效构造软件质量和客观评价软件质量的前提。确定度量需求首先要确定度量的目的,软件质量度量的目的通常是把软件产品的质量与用户的质量需要相比较。其次确定产品的类型,最后选

择相应的软件质量模型。

度量设计包括:选择度量类型、确定定量定级水平、确定评价准则。由于软件的类型不同,对软件的质量要求就不同,所以在度量设计时必须选择与软件相应的度量类型。软件的质量特性是无法直接度量的,因此需定义每个质量特性要求的级别(评定等级),每个质量特性都要标识应用的度量,这就是确定定量定级水平。评价准则是用来确定一个特定软件产品的总体质量是否能够被接受的已经定义成文的规则和条件的集合。

执行软件度量是收集指标数据和进行数据计算的过程。采集的数据应该基于正确定义的度量和模型,从而保证数据的正确性、准确性和精度;因此,在收集数据之前,应当设定数据采集的目标,并且定义有意义的问题。同时必须详细规定使用的数据分类计划和精确度,避免脱离度量采集过多无用信息。

收集到有关的数据后,必须选择合适的度量指标,采用一种适当的方式对数据进行分析处理,得出度量结果。

2 度量指标

选择合适的指标并使其量化是软件测试与质量评估的关键。度量指标可以分为定性指标和定量指标两种。从理论上讲,选择定量指标,能够更科学客观地反映软件的质量特征。但是也不能忽视定性指标,因为有效的质量评价是建立在定量数据和定性评估两者的发现和观察结果的交叉验证基础之上的。

在选取评估指标时,应该注意把握如下几项原则:针对性、可测性、简明性、完备性、客观性。另外,

收稿日期: 2007-04-30

第一作者: 女,1982 年生,硕士生

E-mail: cheng5109@163.com

选择的度量指标不是越多越好,关键在于指标在评估中所起的作用的大小。否则,过多指标不仅增加结果的复杂性,有时甚至会影响评估的客观性。指标的确定一般是采用自顶向下的方法,逐层分解,并且需要在动态过程中反复综合平衡。

2.1 度量指标领域模型

软件质量度量的实施对象是要测量的各个度量元,但随着具体的软件产品、项目及其软件过程和服务对象的不同,对软件质量需求的指标也不同,但是所有的度量指标模型有个共性部分,即最上层的度量指标领域模型,包括:技术满意度、扩展与稳定性、开发性能、资源与费用、计划与进度、产品质量、客户满意度^[5]。如图 1 所示。

2.2 度量指标模型

本文基于上述的软件度量指标的分析,分析软

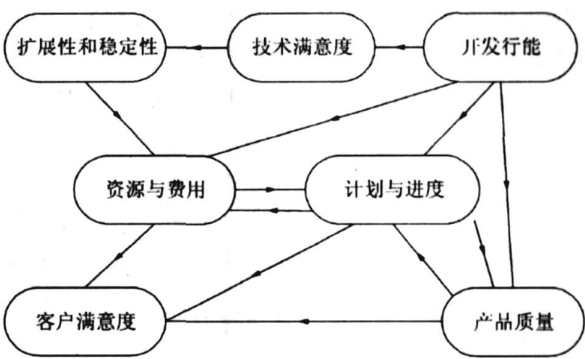


图 1 度量指标领域模型
Fig. 1 The field model of metric index

件质量标准 ISO/ IEC9126 的质量模型,并改进了质量度量指标模型。质量度量指标模型具有分层的体系结构,如图 2 所示。

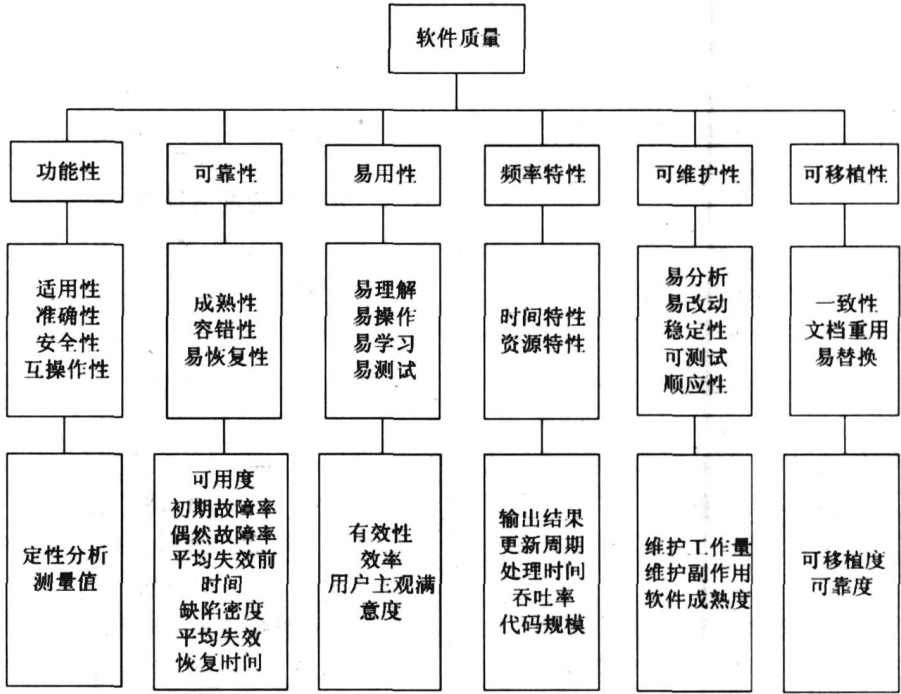


图 2 软件质量综合特性和定量指标度量模型
Fig. 2 Composite characteristics of software quality and mesurable index metric model

该定量指标度量模型是在软件内部和外部质量特性模型的基础上改进得到的。该模型分为三层:第一层是面向管理的质量特性,每一个质量特性是用以描述和评价软件质量的一组属性,代表软件质量的一个方面;第二层质量子特性是上层质量特性的细化,一个特定的质量特性可以对应若干个质量子特性;第三层是度量特性,对于度量特性只要按照相应的标准进行度量,就可为质量特性和质量子特

性提供定量值。

其中各定量指标是由软件工程工作人员在长期的工作经验中得出的,直观明了,具有可操作性、实用性、可行性和有效性。

2.3 定量的度量方法

通过该度量指标模型定义的度量元,进行软件质量的定量度量。

2.3.1 明确度量需求 在软件开发前期或初期进

行定义度量需求,即在具体应用的特定环境下的软件产品质量的定量要求,它是有效构造软件质量和客观评价软件质量的前提。由于软件质量特性和子特性描述的软件度量需求常常无法直接测量,需要进一步确定相关的度量元,将所有的度量和模型都应该和终端产品质量联系起来。根据应用环境为软件开发的各个阶段和其最终产品分别确定适当的度量元,建立度量元、质量子特性、质量特性的映射模型,确定合理的评估准则。

2.3.2 明确采集数据对象 基于正确定义的度量和模型采集数据。采集数据前详细规定使用的数据分类计划和精确度,此交付项用作要实施的度量和测量过程的输入,即所要收集的特性及其权重。

2.3.3 规范度量执行过程 软件质量的度量流程为:度量元指标——子特性指标——特性指标。

有了切实可行的度量模型,按照度量元、质量子特性、质量特性的层次顺序执行度量。

(1) 对评价模型中的度量元进行适用性选择,并根据适用度量元的计分方式进行计分,从而获得所有适用度量元的测量值;指标权重的计算可参考文献[8]通过 PHP 方法计算;

(2) 计算子特性所辖的全部适用度量元的测量值的算术平均值,从而获得子特性的测量值;

(3) 根据分配给特性所辖的子特性的权值,计算特性所辖的子特性的测量值的加权平均值,从而获得特性的测量值;

(4) 根据分配给特性的权值,计算特性测量值的加权平均值,从而获得质量测量值。

3 实例分析

利用图 2 的指标模型,模拟某一软件的可靠性特性的度量方法。按照上述度量方法的度量步骤如下。

3.1 度量元测量值计算

选择子特性的适用度量元,并分别计算各度量元的测量值,如表 1 所示。

3.2 子特性权重、测量值计算

通过判断矩阵确定子特性成熟性、容错性和易恢复性的权重,参照文献[9]得到子特性的测量值和权重:

$$\begin{bmatrix} 1 & 1/3 & 1 \\ 3 & 1 & 3 \\ 1 & 1/3 & 1 \end{bmatrix}$$

根据特征根法,计算得到各子特性的权重与测量值。如表 2 所示。

表 1 子特性测量值表

Table 1 Sub-characteristic values

特性	度量元	测量值
可靠性	可用度 C21	
	初期故障率 C22	
	偶然故障率 C23	
	平均失效前时间 C24	
	平均失效间隔时间 C25	
	缺陷密度 C26	
	平均失效恢复时间 C27	

表 2 子特性权重

Table 2 Sub-characteristic powers

子特性	测量值	权重
成熟性	0.68	0.6
容错性	0.5	0.2
易恢复性	0.7	0.2

3.3 可靠性度量值计算

通过子特性的测量值和权重综合计算可靠性的度量之 V_{C2} :

$$W_{2j} = 1$$

W_{2j} 是可靠性的第 j 个子特性的权值;

$$V_{C2} = \sum_{j=1}^3 (V_{2j} \times W_{2j}) =$$

$$(0.68 \times 0.6 + 0.5 \times 0.2 + 0.7 \times 0.2) = 0.648$$

V_{2j} 是可靠性的第 j 个子特性的测量值。

同样方法,计算软件的其他特性的度量值。

3.4 综合特性度量值计算

按照质量特性的测量值和权重的求和公式计算软件的综合特性度量值。

$$V = \sum_i (V_{Ci} \times W_{Ci})$$

$$W_{Ci} = 1$$

V_{Ci} 、 W_{Ci} 分别是软件的第 i 个特性的测量值和权值。

4 结束语

软件质量度量活动的关键与难点在于明确了业务目标后进行度量指标设计,而设计的核心部分就是度量指标模型。由于软件质量的度量具有其特有

的规范和要求,其度量指标涉及面广、不确定性因素较多、量化困难,至今还没有统一的标准。通过建立科学合理的软件质量度量指标体系,充分考虑到软件的特殊性,借鉴其他学科的质量评估理论,有可能全面客观地度量软件质量。

参考文献:

- [1] FENTON N E, PFLIEGER S L. 软件度量[M]. 2版. 北京:机械工业出版社,2004.
- [2] BOEHM BROWN B W, KASPAR H, et al. Characteristics of software quality. TRW series of software technology Vol1[M]. New York:North Holland,1978.
- [3] CALL J Mc, RICHARDS P, WALTERS G. Factors in software quality[R]. Technical Report CDRL A003, US Rome Air Development Centre,1977.
- [4] ISO/IEC 9126-1. Software engineering-product quality (Part 1):quality model[M]. 2001-06-15.
- [5] 张红延,黄涛,蒋明青. 软件质量度量及其模型指标的研究[J]. 计算机应用研究,2003,10:20.
- [6] 吴源俊. 软件质量要求及评价的共性问题[J]. 软件工程及标准化,2004,2:41-43.
- [7] 梁成才,汤伟. 软件质量的定量评定方法[J]. 计算机工程,2003,29(14):97.
- [8] 陈建明,王海峰. 软件质量模型及其评价[J]. 微电子学与计算机,2003(5):65-67.
- [9] 肖红民. 软件质量评价的一种定量方法[J]. 西北师范大学学报,2000,36(4):33-35.

Software quality metric based on analysis of quantitative index

CHENG Nuo WAN Lin ZHANG Wei

(Department of Information Engineering, Academy of Armored Force Engineering, Beijing 100072, China)

Abstract: Software quality evaluation with quantitative method can help the managers' decision-making, and reduces the unilateralism caused by qualitative evaluation or subjective evaluation. After analyzing quantitative index of software quality with quantitative method, the paper discusses quantitative methods for software quality evaluation, and improves the combined character of software quality and the metric model of quantitative index based on the software quality standard ISO/IEC9126.

Key words: software quality; software metric model; metric index model

(上接第 125 页)

An auxiliary regression test tool for changed nodes coverage

YU XiuShan DONG Xin JIANG XiuFeng

(Institute of China Electronic Systems Equipment Company, Beijing 100039, China)

Abstract: Regression test case design is an important factor in regression test. Starting from the application point of view and based on the program flow graph produced by commercial software testing tools, an algorithm that can cover all paths of changed nodes was proposed and implemented. The application experiments showed that this tool can aid regression test effectively.

Key words: software test; regression test; test cases selection