

# 使用 McCabe IQ 提高测试质量的研究

黄华林

(广东女子职业技术学院, 广东 广州 511450)

**摘要:** 随着软件产业的急速发展, 应用软件系统规模不断增大, 企业对软件质量的重视程度越来越高, 软件企业对软件测试的投入也逐渐增加。探讨了 McCabe IQ 测试工具对软件测试质量和效益的提升。

**关键词:** 覆盖率; McCabe IQ

中图分类号: TP311.5

文献标识码: A

文章编号: 1674-7720(2012)03-0069-02

## Using McCabe IQ to improve the quality of test

Huang Hualin

(Guangdong Women's Professional Technical College, Guangzhou 511450, China)

**Abstract:** With the rapid development of software industry, as well as the increasing size of software project, companies focus on software quality, and software companies allocate more and more resources to software testing. This paper discusses the McCabe IQ test tools for improving software testing quality and efficiency.

**Key words:** coverage; McCabe IQ

随着软件产业的急速发展, 应用软件系统规模不断增大, 企业对软件质量的重视程度越来越高, 软件企业对软件测试的投入也逐渐增加。目前的商业环境下, 商业应用软件的测试着重考虑如下问题:

(1) 效益。怎样确保每个测试都有时间和经济方面的效益。(2) 彻底。测试到达怎样的程度可以合理地认为程序已经没有错误了。(3) 资源分配。测试资源的分配是否合理, 是否聚焦在软件的高危模块上, 核心功能部分是否得到足够的测试。

软件测试中, 覆盖率分析的方法越来越多地为测试人员使用。覆盖率是软件测试的一项重要指标, 通过达到良好的测试覆盖率, 可以保证软件测试的质量。

### 1 McCabe IQ 测试工具

McCabe IQ 是美国 McCabe Software 公司的软件质量管理解决方案, McCabe IQ 是一款基于 McCabe 圈复杂度进行质量度量 and 测试的工具, 为用户提供软件质量度量、软件结构分析及动态结构化测试的全面支持。McCabe IQ 提供客观的方法评估测试覆盖率, 具有一致性、重复性、客观解释、质量评估、优化测试及分配资源等强大功能, 能真正保证测试每个独立路径, 可以确定

测试过的路径和没测试过的路径。

圈复杂度是用来衡量一个模块判定结构的复杂程度, 数量上表现为独立现行路径条数, 即合理的预防错误所需测试的最少路径条数<sup>[1]</sup>。圈复杂度高说明代码质量可能很差, 难于测试和维护。根据经验, 程序的可能错误和圈复杂度高有着很大关系。圈复杂度显示了在测试一个单元时, 为保证软件质量而需要测试的基本路径的最小数目<sup>[2]</sup>。McCabe IQ 建议将圈复杂度不超过 10, 因为更高的复杂度使测试变得复杂并且可能会发生更多的错误。

### 2 使用 McCabe IQ 做覆盖率分析

McCabe IQ 的度量基于软件结构数学化的严格分析, 这种分析以 McCabe 图复杂度为基础, 明确地确定了高风险区域。McCabe IQ 就是一款基于 McCabe 圈复杂度进行质量度量和测试的工具。McCabe IQ 覆盖率分析使用了源代码插装技术, 能生成分支和路径覆盖报告。对于给定的程序, 所有的可测试路径的数目通常非常巨大, 找到一个有意义的可测试路径子集非常重要。被 McCabe IQ 标识出的路径虽不能代表程序中所有可能路径, 但最小的测试路径子集至少遍历每个一次判定,

## 技术与方法 Technique and Method

将使 McCabe IQ 的“圈复杂度”路径分析技术成为可能。

McCabe IQ 可以很方便地做到如下分析:

(1) 复杂度分析: McCabe IQ 提供了复杂度信息的即时访问, 测量不同的源代码特性, 标识出相关的复杂度信息和程序不同模块的结构化程度, 特别复杂或非结构化代码部分可以在覆盖率报告中突出出来, 使得测试工程师更容易识别出有缺陷的代码。

(2) 软件变更分析: 当一个程序被修改, 测试需要聚焦在修改的代码上, 以及那些被修改代码影响的代码, McCabe IQ 的软件变更分析不但可以标识出变更过的代码, 而且还可以标识出被修改所影响的模块。

### 2.1 McCabe IQ 提升功能测试

在功能测试中, McCabe IQ 从程序的需求中得到要测试的功能, 然后执行测试用例以验证程序是否按照预期运行。功能测试的目的是为了识别缺失的功能、不能正确运行的功能, 以及功能缺陷<sup>[1]</sup>。

#### 2.1.1 功能测试在执行过程中要解决如下问题:

(1) 隐含功能的丢失。由于需求定义隐含缺陷, 从需求文档得到的功能测试会丢失隐含功能。这样, 功能测试很难确保测试的完全性, 而且功能测试在功能合并阶段也会导致有问题的行为。

(2) 停止测试的时机。在功能测试中, 可测试功能组合的数量相当巨大, 最糟糕的情况是功能组合的数量将是 2 的功能需求次幂的形式, 显然测试所有的功能组合是不可能的。

#### 2.1.2 通过 McCabe IQ 的覆盖分析可以提升功能测试

(1) McCabe IQ 覆盖分析可以指出未测试的分支, 并突出为测试的分支, 隐含功能和明确的功能需求将同样被突出出来。

(2) McCabe IQ 覆盖分析可以识别出高风险模块块, 然后使用更为苛刻的路径覆盖进行分析。

使用路径覆盖分析可以遍历所有的关键功能组合, 包括隐含的功能和明确的功能。使用图复杂度路径作为危险模块的测试程度的索引, 可以追踪记录可累加的分支和路径执行的次数, 通过基于累加的分支和圈复杂度覆盖率决定可以接受的测试程度, 从而解决以上两个问题。

### 2.2 McCabe IQ 提升单元测试

单元测试是针对软件的最小模块进行正确性检验的测试工作, 是代码正确性验证的重要措施。这些最小模块是隔离出来的代码段, 称为单元。单元测试的最大挑战在于识别出需要运行的最小的测试的集合, 理想情况下, 每一个可能路径都要被测试, 但是这通常是不可能的。挑战在于分离出路径的子集来提供所有可测试路径的覆盖分析, 并且使得路径子集最小, 并且没有单元级的冗余<sup>[4]</sup>。

McCabe IQ 路径图精确的被设计为单元级的覆盖管

理, McCabe IQ 识别出的最小路径集合可以测试到所有的代码单元, 提供所有可测试路径的覆盖分析, 至少确认最危险模块已经被覆盖。

### 2.3 McCabe IQ 提升回归测试

回归测试是指测试程序开发过程中的修订版本, 这种测试验证新增的功能必须按照预期运行, 所做的修改达到了预定的目的, 错误得到改正, 未变更的功能没有被修改或变更所影响<sup>[5]</sup>。

#### 2.3.1 回归测试需要解决问题

(1) 程序修改后要找出程序中被变更影响的模块。这样, 需要做回归测试的只是受变更影响的模块, 提高了效率。

(2) 验证修改过的功能已经被测试。程序的变更通常会引入新的隐含功能, 没有黑盒测试的技术, 验证所有变更已经被测试是不可能的。

(3) 验证新的功能已经被测试。当做完全功能测试的时候, 测试新的功能, 要验证完整性测试, 隐含功能的测试及功能组合的测试<sup>[6]</sup>。

#### 2.3.2 通过 McCabe IQ 的覆盖分析可以提升回归测试

McCabe IQ 覆盖分析能精确指出程序中已经变更的分支和模块。覆盖分析可以限制在这个区域, 聚焦那些需要测试的地方。McCabe IQ 隔离出那些被变更影响的程序区域, 允许聚焦在需要做回归测试的部分, 而忽略那些和变更的程序无关联的模块。对于修改过的功能和新功能, McCabe IQ 与新的功能测试一样去对待。

软件的功能测试根据需求来验证软件功能的实现情况, 单元测试主要认证代码的正确性, 回归测试保证新做的修改不影响原有的功能需求, McCabe IQ 的覆盖率分析在这些方面都能提供很好的帮助。灵活运用 McCabe IQ 测试工具可以提高测试效率和测试质量, 挖掘出软件的潜在错误, 保证软件产品的质量。

#### 参考文献

- [1] 朱鸿, 金凌紫. 软件质量与保证[M]. 北京: 科学出版社, 1997.
- [2] 樊庆林, 吴建国. 提高软件测试效率的方法研究[J]. 计算机技术与应用, 2006, 16(10).
- [3] 叶言苓, 崔彦军. 软件测试管理的研究与应用[J]. 计算机应用与软件, 2003, 20(9).
- [4] 李秋英, 李海峰, 徐刚. 基于覆盖率信息的软件可靠性增长测试实践[J]. 计算机应用研究, 2010, 27(7).
- [5] 高海昌, 贺晓红, 冯博琴, 等. 软件结构测试自动化关键技术研究[J]. 微电子学与计算机, 2005, 22(02).
- [6] 郑艳. 一种支持软件测试的测试工具研究[J]. 无锡职业技术学院学报, 2008(4).

(收稿日期: 2011-08-12)

#### 作者简介:

黄华林, 男, 1981 年生, 讲师, 硕士, 主要研究方向: 计算机技术应用, 软件测试。

《微型机与应用》2012 年第 31 卷第 3 期