College of Computer Science, Chongqing University

Software Engineering

A Practitioner's Approach Seventh Edition

21 软件质量保证

zmqmail@cqu.edu.cn 13708390417

重庆大学 计算机学院 COLLEGE OF COMPUTER SCIENCE

软件质量保证是一种普适性活动

软件质量保证是适用于整个软件过程的一种普适性活动
SQA is an umbrella activity

软件质量保证（SQA）

- 软件质量保证（SQA ）内容:
  - 1.SQA过程
  - 2.具体的质量保证和质量控制任务（包括技术评审和多层测试策略）
  - 3.有效的软件工程实践（方法和工具）
  - 4.对所有软件工作产品及其变更的控制
  - 5.保证符合软件开发标准的规程（在使用的情况下）
  - 6.测量和报告机制

21.2 软件质量保证内容 Elements of SQA(Software Quality Assurance)

提出一组活动将有助于保证每个软件工程工作产品表现出高质量。

- Standards (标准)
- Reviews And Audits(评审和审核)
- Testing (测试)
- Error/Defect Collection and Analysis (错误/缺陷的收集与分析)
- Change Management (变更管理)
- Education (教育)
- Vendor Management (供应商管理)
- Security Management (安全管理)：软件保护系统数据的安全
- Safety (安全)：软件缺陷带来的影响可能是灾难性的（例如飞机控制系统）
- Risk Management (风险管理)

这些事情都在做质量保证

21.3 Role of the SQA Group-I

从事质量保证策划、监督、记录、分析和报告。

质量保证任务

- 编制项目质量保障计划

  Prepares an SQA **plan** for a project.

  - The plan identifies
    - Evaluations to be performed
    - Audits and reviews to be performed
    - Standards that are applicable to the project
    - Procedures for error reporting and tracking
    - Documents to be produced by the SQA group
    - Amount of feedback provided to the software project team
- 参与项目的软件过程描述的编写

  Participates in the development of the project's software process description.

  - The SQA group reviews the process description for compliance with organizational policy, internal software standards, externally imposed standards (e.g., ISO-9001), and other parts of the software project plan.

21.3 Role of the SQA Group-II

从事质量保证策划、监督、记录、分析和报告。

质量保证任务

- 评审软件工程活动，以验证是否符合规定的软件过程

  Reviews software engineering activities to verify compliance with the defined software process. Identifies, documents, and tracks deviations from the process and verifies that corrections have been made.

- 审核指定的软件工作产品以验证是否遵守作为软件过程一部分的那些规定

  Audits designated software work products to verify compliance with those defined as part of the software process.

  - Reviews selected work products; identifies, documents, and tracks deviations; verifies that corrections have been made
  - Periodically reports the results of its work to the project manager.

21.3 Role of the SQA Group-III

从事质量保证策划、监督、记录、分析和报告。

质量保证任务

- 确保根据文档化的规程记录和处理软件工作和工作产品中的偏差
  Ensures that deviations in software work and work products are documented and handled according to a documented procedure.
- 记录各种不符合项并报告给高层管理人员
  Records any noncompliance and reports to senior management.

21.4 SQA Goals (see Figure 21.1)

- **需求质量 Requirements Quality**.
  - The correctness, completeness, and consistency of the requirements model will have a strong influence on the quality of all work products that follow.
- **设计质量 Design Quality**.
  - Every element of the design model should be assessed by the software team to ensure that it exhibits high quality and that the design itself conforms to requirements.
- **代码质量 Code Quality**.
  - Source code and related work products (e.g., other descriptive information) must conform to local coding standards and exhibit characteristics that will facilitate maintainability.
- **质量控制效率 Quality Control Effectiveness**.
  - A software team should apply limited resources in a way that has the highest likelihood of achieving a high quality result.

目标	属性	度量
需求质量	歧义	引起歧义地方的修改数量（例如：许多、大量、与人友好）
	完备性	TBA, TBD的数量
	可理解性	节/小节的数量
	易变性	每项需求变更的数量
		变更所需要的时间（通过活动）
	可追溯性	不能追溯到设计/代码的需求数
	模型清晰性	UML模型数
		每个模型中描述文字的页数，
		UML错误数
设计质量	体系结构完整性	是否存在现成的体系结构模型
	构件完备性	追溯到结构模型的构件数
		过程设计的复杂性
	接口复杂性	挑选一个典型功能或内容的平均数
		布局合理性
	模式	使用的模式数量
代码质量	复杂性	环路复杂性
	可维护性	设计要素（第8章）
	可理解性	内部注释的百分比
		变量命名约定
	可重用性	可重用构件的百分比
	文档	可读性指数
质量控制效率	资源分配	每个活动花费的人员时间百分比
	完成率	实际完成时间与预算完成时间之比
	评审效率	参见评审度量（第14章）
	测试效率	发现的错误及关键性问题数
		改正一个错误所需的工作量
		错误的根源

Java 开发中必须遵守的军规

Cookies 程序员头号内推圈 2019-06-14

程序员头号内推圈报道

编辑：Cookies

来源：程序员共成长

让软件像数学一样证明?

形式验证一般被称为形式化验证方法，是相对于传统的验证（模拟、仿真和测试）而言的。形式化验证方法的主要思路就是使用数学的公式、定理和系统来验证一个系统的正确性等。

21.5 软件质量保证的形式化方法 (Formal approaches to SQA)

- Formal approaches to SQA

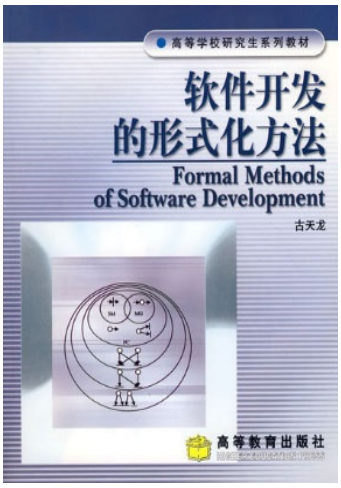
  软件质量保证的形式化方法

- Over the past three decades, a small, but vocal, segment of the software engineering community has argued that a more formal approach to software quality assurance is required.

  一个计算机程序可以看作是一个数学对象，每一种程序设计语言都有一套定义严格的语法和语义，而且软件的需求规格说明也有严格的方法(第21 章) .如果需求模型(规格说明)和程序设计语言都以严格的方式表示，就可以采用程序正确性证明来说明程序是否严格符合它的规格说明.

软件形式化方法包括以下几个主要研究方向

- 1. 基础概念：
- 2. 形式化方法与面向对象方法的结合：
  - 例如：
    - Statecharts、**Petri网**、Z语言、VDM等.
- 3. 工具开发：



21.6 软件质量保证的统计学方法 (Statistical SQA)

- For software, statistical quality assurance implies the following **steps**:
  - 1.Information about software errors and defects is collected and categorized.
    收集软件的错误和缺陷信息，并进行分类。
  - 2.An attempt is made to trace each error and defect to its underlying cause (e.g., non-conformance to specifications, design error, violation of standards, poor communication with the customer).
    追溯每个错误和缺陷形成的根本原因(例如，不符合规格说明、设计错误、违背标准、缺乏与客户的交流)。

21.6 软件质量保证的统计学方法 (Statistical SQA)

- For software, statistical quality assurance implies the following **steps**:
  - 3.Using the Pareto principle (80 percent of the defects can be traced to 20 percent of all possible causes), isolate the 20 percent (the vital few).
    使用Pareto原则(80%的缺陷可以追溯到所有可能原因中的20%)，将这20%（“重要的少数”)原因分离出来。
  - 4.Once the vital few causes have been identified, move to correct the problems that have caused the errors and defects.
    一旦找出这些重要的少数原因，就可以开始纠正引起错误和缺陷的问题。

不完整或错误的规格说明(IES)

与客户交流中所产生的误解(MCC)

错误原因

数据表示有错(EDR)

	Total		Serious		Moderate		Minor	
Error	No.	%	No.	%	No.	%	No.	%
IES	205	22%	34	27%	68	18%	103	24%
MCC	156	17%	12	9%	68	18%	76	17%
IDS	48	5%	1	1%	24	6%	23	5%
VPS	25	3%	0	0%	15	4%	10	2%
EDR	130	14%	26	20%	68	18%	36	8%
ICI	58	6%	9	7%	18	5%	31	7%
EDL	45	5%	14	11%	12	3%	19	4%
IET	95	10%	12	9%	35	9%	48	11%
[illegible]	36	4%	2	2%	20	5%	14	3%
[illegible]	60	6%	15	12%	19	5%	26	6%
HCI	28	3%	3	2%	17	4%	8	2%
MIS	56	6%	0	0%	15	4%	41	9%
Totals	942	100%	128	100%	379	100%	435	100%

错误原因

	Total		Serious		Moderate		Minor	
Error	No.	%	No.	%	No.	%	No.	%
IES	205	22%	34	27%	68	18%	103	24%
MCC	156	17%	12	9%	68	18%	76	17%
IDS	48	5%	1	1%	24	6%	23	5%
VPS	25	3%	0	0%	15	4%	10	2%
EDR	130	14%	26	20%	68	18%	36	8%
ICI	58	6%	9	7%	18	5%	31	7%
EDL	45	5%	14	11%	12	3%	19	4%
IET	95	10%	12	9%	35	9%	48	11%
IID	36	4%	2	2%	20	5%	14	3%
PLT	60	6%	15	12%	19	5%	26	6%
HCI	28	3%	3	2%	17	4%	8	2%
MIS	56	6%	0	0%	15	4%	41	9%
Totals	942	100%	128	100%	379	100%	435	100%

53% IES、MCC和EDR是造成所有错误的53%的"重要的少数'原因

六西格玛

产业界应用最广泛

基于统计的质量保证策略

Σ

- 20世纪80年代在摩托罗拉公司最先普及，六西格玛策略是严格且规范的方法学。
- “六西格玛”一词来源于6个标准偏差。
  - 每百万个操作发生3 .4个偏差(缺陷)，它意味着非常高的质量标准。
- 运用数据和统计分析，通过识别和消除制造以及服务相关过程中的‘缺陷’来测量和改进企业的运转状况。

软件工程中的六西格玛 (Six-Sigma for Software Engineering)

- The Six Sigma methodology defines these **Core Steps**:
  - **定义**：通过与客户交流的方法来定义客户需求、可交付的产品及项目目标。
    Define customer requirements and deliverables and project goals via well-defined methods of customer communication
  - **测量**：测量现有的过程及其产品，以确定当前的质量状况(收集缺陷度量信息)。
    Measure the existing process and its output to determine current quality performance (collect defect metrics)
  - **分析**：分析缺陷度量信息. 并挑选出重要的少数原因。
    Analyze defect metrics and determine the vital few causes.
  - **改进**：通过消除缺陆根本原因的方式来改进过程。
    Improve the process by eliminating the root causes of defects.
  - **控制**：控制过程以保证以后的工作不会再引入缺陷原因。
    Control the process to ensure that future work does not reintroduce the causes of defects.

21.6.1 Software Reliability

- **MTBF：** A simple measure of reliability is mean-time-between-failure (**MTBF**)(平均故障间隔时间), where

  MTBF = MTTF + MTTR

  The acronyms MTTF(修复前平均时间)and MTTR(平均维护时间)are mean-time-to-failure and mean-time-to-repair, respectively.

- Software availability is the probability that a program is operating according to requirements at a given point in time and is defined as

  Availability = [MTTF/(MTTF + MTTR)] x 100%

21.6.2 Software Safety

- 识别和评估可能对软件产生负面影响并促使整个系统失效的潜在灾难。

  Software safety is a software quality assurance activity that focuses on the identification and assessment of potential hazards that may affect software negatively and cause an entire system to fail.

- If hazards can be identified early in the software process, software design features can be specified that will either eliminate or control potential hazards.
  如果能够在软件过程的早期阶段识别出这些灾难，就可以指定软件设计特性来消除或控制这些潜在的灾难。

附：几个标准对过程的定义

- IEEE-STD-610
  - 过程是针对确定的目的所实施的序列步骤，例如软件开发过程。
- ISO 9000 : 2000
  - 过程是使用资源将输入转化为输出的活动的系统。
- ISO/IEC 12207
  - 过程是把输入转换为输出的一组彼此相关的活动。

Summary

Quality Assurance

↓

Standard

工作产品（体现）

建模、编码阶段

技术评审的输出(第15章)

测试阶段(第17-20章)

制定的测试计划和测试规程；

也可能产生其他与过程改造相关的工作产品。

为了制定软件团队的SQA 策略，需要建立“软件质量保证计划”

Question?

Q&A?

College of Computer Science, Chongqing University

Software Engineering

A Practitioner's Approach Seventh Edition

16 Software Quality Assurance

zmqmail@cqu.edu.cn 13708390417

重庆大学 计算机学院 COLLEGE OF COMPUTER SCIENCE