

<<软件质量经济学>>

图书基本信息

书名：<<软件质量经济学>>

13位ISBN编号：9787111447443

10位ISBN编号：7111447441

出版时间：2014-1

出版时间：机械工业出版社

作者：（美）Capers Jones,（法）Olivier Bonsignour

译者：廖彬山,张永明,崔曼

版权说明：本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问：<http://www.tushu007.com>

<<软件质量经济学>>

内容概要

<<软件质量经济学>>

作者简介

【作者简介】

Capers Jones Namcook Analytics公司副总裁兼CTO、美国软件生产力研究所（SPR）的创办人及首席科学家，曾是Capers Jones & Associates公司的总裁兼CEO。

他是IEEE会员和IFPUG（国际功能点用户组）会员，专注于软件评估、软件测量、软件质量和软件项目管理，是软件工程领域最有影响的思想领袖之一，著有《软件评估、基准测试与最佳实践》、《Software Systems Failure and Success》和《Applied Software Measurement, Third Edition》等书。

Olivier Bonsignour CAST软件公司产品开发副总裁，负责研发与产品管理。

在加入CAST之前，Bonsignour是法国国防部武器装备总署（DGA）CIO。

他是分布式系统开发和面向对象开发领域的先驱，1996年成为最早采用CAST技术的人之一，并且随后加入了CAST。

他也是分布式系统和面向对象开发的早期的开拓者，拥有法国国立应用科学学院工程与计算机科学硕士学位。

在工作之余，他喜欢游泳、骑自行车、滑雪，以及沿法国海岸航行。

【译者简介】

廖彬山 先后就读于南京大学数学系和北京航空航天大学计算机科学与技术系。

现为美国CMMI主任评估师（SCAMPI Lead Appraiser）、北京国信普道科技有限公司首席咨询师、北京航空航天大学软件学院客座教授。

主要从事CMMI的培训、咨询和评估及软件工程理论与方法的研究工作，擅长的方向有软件项目管理、软件估算、功能点分析方法、软件度量、软件测试、同行评审、风险管理、PSP和TSP等。

张永明 毕业于北京航空航天大学计算机科学与技术系，多年在国内外大型IT企业从事软件开发、项目管理、质量保证以及运维管理等工作。

目前主要从事CMMI和GJB5000A方面的培训、咨询及相关理论与方法的研究工作，结合丰富的咨询实践经验，已经为几十家不同领域、不同规模企业的实际研发和项目管理提供了高效、有价值的改进。

崔曼 毕业于吉林大学软件学院，曾从事过软件开发、软件测试、质量保证及CMMI咨询等工作。

担任CMMI咨询师期间，为多家不同类型、不同规模、不同应用领域的企事业单位提供了基于CMMI的咨询和评估服务，有效解决了企业实际存在的问题，积累了丰富的实践经验。

<<软件质量经济学>>

书籍目录

对本书的赞誉

译者序

序

前言

第1章定义软件质量和经济价值 1

1.1引言 1

1.2定义软件质量 6

1.3定义经济价值和定义软件质量的价值 13

1.3.1软件和质量对构建自用内部软件的企业经济价值 14

1.3.2软件和质量对内部软件用户的经济价值 16

1.3.3软件和质量对商业软件厂商的经济价值 17

1.3.4软件和质量对COTS用户和客户的经济价值 18

1.3.5软件和质量对嵌入式软件公司的经济价值 20

1.3.6软件和质量对嵌入式软件设备用户的经济价值 21

1.3.7软件和质量对其他企业部门的经济价值 23

1.3.8同时担当多种角色 23

1.4总结和结论 23

第2章估算和度量软件质量 25

2.1引言 25

2.2使用功能点度量方法来量化潜在缺陷 27

2.3软件潜在缺陷 28

2.3.1软件需求的专门案例 32

2.3.2软件需求的起源 35

2.3.3软件需求的规模、结构和完整性 37

2.3.4最小化软件需求缺陷 39

2.3.5软件需求缺陷结论 45

2.3.6编码缺陷的专门案例 46

2.2估计软件缺陷预防 51

2.3估计软件缺陷检测和缺陷清除 53

2.4度量应用程序结构质量 55

2.4.1度量可靠性 56

2.4.2度量性能效率 57

2.4.3度量安全性 57

2.4.4度量可维护性 58

2.4.5度量规模 59

2.4.6应用程序结构质量度量属性总结 59

2.5结构质量评估案例 62

2.5.1绕过架构 63

2.5.2无法控制处理的数据量 64

2.5.3应用程序资源失衡 65

2.5.4安全弱点 66

2.5.5缺乏防御机制 67

2.5.6对评估结构质量的系统的要求 67

2.6歪曲软件经济学分析的三大问题 68

2.6.1软件历史数据的泄漏 69

<<软件质量经济学>>

| | | |
|--------|--------------------------|-----|
| 2.6.2 | LOC度量方法的经济学问题 | 74 |
| 2.6.3 | 单位缺陷成本度量方法的经济学问题 | 78 |
| 2.6.4 | 案例A：质量低劣 | 79 |
| 2.6.5 | 案例B：质量良好 | 79 |
| 2.6.6 | 案例C：零缺陷 | 80 |
| 2.6.7 | 预测软件潜在缺陷的有用经验法则 | 81 |
| 2.7 | 总结和结论 | 82 |
| 第3章 | 软件缺陷预防 | 84 |
| 3.1 | 引言 | 84 |
| 3.1.1 | 20世纪70年代IBM进行缺陷预防研究的早期历史 | 84 |
| 3.1.2 | 缺陷预防方法的协同组合 | 89 |
| 3.1.3 | 潜在缺陷和缺陷起源 | 90 |
| 3.1.4 | 缺陷预防、模式、认证的可重用材料 | 94 |
| 3.1.5 | 软件缺陷预防和应用程序规模 | 95 |
| 3.2 | 缺陷预防结果的分析 | 96 |
| 3.2.1 | 敏捷嵌入式用户 | 97 |
| 3.2.2 | 自动化的质量预测 | 97 |
| 3.2.3 | 软件质量数据的基准 | 98 |
| 3.2.4 | 能力成熟度模型集成 (CMMI) | 99 |
| 3.2.5 | 认证方案 | 100 |
| 3.2.6 | 单位缺陷成本度量 | 101 |
| 3.2.7 | 质量成本 | 104 |
| 3.2.8 | 圈复杂度度量 (及相关的复杂度度量) | 106 |
| 3.2.9 | 缺陷度量和缺陷跟踪 | 111 |
| 3.2.10 | 正式审查 | 113 |
| 3.2.11 | 功能点质量度量 | 116 |
| 3.2.12 | ISO 质量标准、IEEE质量标准及其他行业标准 | 121 |
| 3.2.13 | 质量功能展开 (QFD) | 124 |
| 3.2.14 | 风险分析 | 126 |
| 3.2.15 | 六西格玛 | 132 |
| 3.2.16 | 静态分析 | 133 |
| 3.3 | 总结和结论 | 135 |
| 第4章 | 测试前缺陷清除 | 136 |
| 4.1 | 引言 | 136 |
| 4.2 | 小型项目的测试前缺陷清除 | 139 |
| 4.3 | 大型系统的测试前缺陷清除 | 143 |
| 4.4 | 测试前缺陷清除活动的分析 | 148 |
| 4.4.1 | 个人桌面检查 | 148 |
| 4.4.2 | 非正式的同行评审 | 149 |
| 4.4.3 | 文档的自动化文本检查 | 150 |
| 4.4.4 | 正确性证明 | 156 |
| 4.4.5 | Scrum 会议 | 158 |
| 4.4.6 | Poka Yoke | 159 |
| 4.4.7 | Kaizen | 161 |
| 4.4.8 | 结对编程 | 164 |
| 4.4.9 | 规格说明书的客户评审 | 166 |
| 4.4.10 | 独立验证和确认 | 168 |

<<软件质量经济学>>

- 4.4.11SQA评审 169
- 4.4.12阶段评审 174
- 4.4.13审查(需求、架构、设计、代码及其他可交付成果) 176
- 4.4.14用户文档的编辑和校对 187
- 4.4.15源代码的自动静态分析 189
- 4.5总结和结论 195
- 第5章软件测试 196
- 5.1引言 196
- 5.1.1黑盒与白盒测试 205
- 5.1.2功能和非功能测试 206
- 5.1.3自动化和手动测试 206
- 5.1.4软件测试通用形式的讨论 207
- 5.1.5子例程测试 207
- 5.1.6PSP/TSP单元测试 208
- 5.1.7XP单元测试 208
- 5.1.8单元测试 209
- 5.1.9新功能测试 210
- 5.1.10回归测试 210
- 5.1.11集成测试 211
- 5.1.12系统测试 212
- 5.1.13专门的软件测试 213
- 5.1.14压力或容量测试 214
- 5.1.15性能测试 214
- 5.1.16病毒防护测试 215
- 5.1.17渗透测试 217
- 5.1.18安全性测试 218
- 5.1.19平台测试 219
- 5.1.20供应链测试 219
- 5.1.21净室测试 220
- 5.1.22诉讼测试 220
- 5.1.23云测试 221
- 5.1.24SOA测试 221
- 5.1.25独立测试 222
- 5.1.26本土化测试 222
- 5.1.27案例研究测试 223
- 5.1.28涉及用户或客户的测试形式 224
- 5.1.29敏捷测试 224
- 5.1.30可用性测试 224
- 5.1.31现场Beta测试 225
- 5.1.32实验室测试 226
- 5.1.33客户验收测试 226
- 5.1.34测试计划编制 227
- 5.1.35测试用例设计方法 227
- 5.1.36测试用例中的错误和bug 229
- 5.1.37软件项目的测试阶段数 229
- 5.1.38按行业和软件类型分类的不同测试模式 230
- 5.1.39按应用程序规模分类的不同测试模式 233

<<软件质量经济学>>

- 5.1.40在指控质量低劣的诉讼案中记录的测试阶段 234
- 5.1.41使用功能点估算测试用例数 235
- 5.1.42使用功能点估算测试人员的数量 238
- 5.1.43使用功能点估算测试工作量和成本 240
- 5.1.44由开发人员或由专业测试人员执行的测试 244
- 5.2总结和结论 245
- 第6章发布后缺陷清除 247
- 6.1引言 247
- 6.1.1发布后缺陷严重程度等级 248
- 6.1.2从结构质量的观点看严重程度等级 250
- 6.1.3软件的可维护性 255
- 6.1.4按软件应用程序用户分类的缺陷发现率 257
- 6.1.5无效缺陷报告 258
- 6.1.6发生在独特情况下的暂时搁置的缺陷 260
- 6.1.7很多客户报告的重复缺陷 260
- 6.1.8第一年缺陷发现率 261
- 6.1.9度量DDE和DRE 262
- 6.1.10发布后缺陷报告的变化 263
- 6.1.11报告软件缺陷的方法的变化 266
- 6.1.12缺陷在报告后由谁修复 269
- 6.1.13案例1: 开发人员负责缺陷修复 269
- 6.1.14案例2: 维护专家负责缺陷修复 270
- 6.1.15案例研究对比 270
- 6.1.16由低劣质量引起的诉讼 271
- 6.1.17发布后的缺陷修复成本模式 272
- 6.1.18参与缺陷修复的软件职业团队 273
- 6.1.19检查发布后缺陷修复的自变量 277
- 6.1.20按功能点计算的应用程序规模 278
- 6.1.21软件应用程序中容易出错模块 284
- 6.2来自发布后缺陷的用户和行业的成本 286
- 6.2.1安全漏洞对企业和政府机构的影响 289
- 6.2.2用于缺陷报告和修复安装的客户后勤 290
- 6.2.3案例1: 一个小公司的小型应用程序 292
- 6.2.4案例2: 大公司的大型应用程序 293
- 6.2.5在维护和发布后缺陷修复中的度量问题 296
- 6.3总结和结论 300
- 第7章软件质量的经济学分析 301
- 7.1引言 301
- 7.2软件的经济价值 302
- 7.2.1度量价值的方法 302
- 7.2.2资金投入批准和应用程序规模 308
- 7.2.3软件构建难度对软件质量的影响 309
- 7.2.4来自于软件的收入产生 313
- 7.2.5软件行业和其他行业之间的差异 315
- 7.2.6软件带来的成本降低 316
- 7.3低质量和高质量软件的经济影响 320
- 7.3.1软件开发和维护 321

<<软件质量经济学>>

- 7.3.2软件作为市场上销售的商品 321
- 7.3.3软件作为减少人力的一种方法 322
- 7.3.4软件和创新性的新型产品 322
- 7.3.5技术债务：度量软件质量对软件成本影响的方法 324
- 7.3.6量化商业价值的框架 327
- 7.3.7超越功能质量 330
- 7.3.8软件结构对质量的影响 330
- 7.3.9员工培训对质量的影响 331
- 7.3.10职业资格认证对质量的影响 332
- 7.3.11技术投资对质量的影响 332
- 7.3.12项目管理对质量的影响 333
- 7.3.13质量控制方法论和工具对质量的影响 334
- 7.3.14高质量和低质量对软件进度的影响 336
- 7.3.15高质量和低质量对软件人员配置的影响 337
- 7.3.16高质量和低质量对软件开发工作量的影响 338
- 7.3.17高质量和低质量对开发生产率的影响 339
- 7.3.18高质量和低质量对软件开发成本的影响 339
- 7.3.19高质量和低质量对每个功能点的开发成本的影响 340
- 7.3.20高质量和低质量对项目取消率的影响 341
- 7.3.21高质量和低质量对取消项目的时间安排的影响 341
- 7.3.22高质量和低质量对取消的项目工作量的影响 343
- 7.3.23高质量和低质量对相对平均项目的工作量的影响 343
- 7.3.24高质量和低质量对软件测试阶段的影响 344
- 7.3.25高质量和低质量对测试占开发百分比的影响 346
- 7.3.26高质量和低质量对每个功能点的测试用例的影响 347
- 7.3.27高质量和低质量对创建的测试用例数的影响 347
- 7.3.28高质量和低质量对测试覆盖的影响 348
- 7.3.29专业测试人员对高质量和低质量的影响 349
- 7.3.30高质量和低质量对软件潜在缺陷的影响 350
- 7.3.31高质量和低质量对总体软件缺陷的影响 352
- 7.3.32高质量和低质量对DDE的影响 352
- 7.3.33高质量和低质量对DRE的影响 353
- 7.3.34高质量和低质量对总缺陷清除的影响 354
- 7.3.35高质量和低质量对已交付给客户的缺陷的影响 355
- 7.3.36高质量和低质量对每功能点交付的缺陷的影响 355
- 7.3.37高质量和低质量对交付的缺陷严重程度等级的影响 355
- 7.3.38高质量和低质量对每功能点严重缺陷数的影响 357
- 7.3.39高质量和低质量对软件可靠性的影响 357
- 7.3.40高质量和低质量对维护和支持的影响 358
- 7.3.41高质量和低质量对维护和支持成本的影响 359
- 7.3.42高质量和低质量对维护缺陷容量的影响 360
- 7.3.43高质量和低质量对软件改进的影响 361
- 7.3.44高质量和低质量对改进成本的影响 362
- 7.3.45高质量和低质量对维护和改进人员配置的影响 363
- 7.3.46高质量和低质量对5年的总工作量的影响 363
- 7.3.47高质量和低质量对TCO的影响 366
- 7.3.48高质量和低质量对COQ的影响 369

<<软件质量经济学>>

| | | |
|--------|--------------------------|-----|
| 7.3.49 | 高质量和低质量对每个功能点的TCO和COQ的影响 | 372 |
| 7.3.50 | 高质量和低质量对应用程序使用寿命的影响 | 376 |
| 7.3.51 | 高质量和低质量对软件应用程序有形资产价值的影响 | 377 |
| 7.3.52 | 高质量和低质量对ROI的影响 | 378 |
| 7.3.53 | 高质量和低质量对于被取消项目的影晌 | 378 |
| 7.3.54 | 高质量和低质量对取消成本差异的影响 | 379 |
| 7.3.55 | 高质量、平均质量和低质量的软件项目的分布 | 380 |
| 7.4 | 总结和结论 | 381 |
| 7.4.1 | 对于10 000 个功能点的高质量结果 | 381 |
| 7.4.2 | 10 000个功能点的低质量软件结果 | 382 |
| | 参考文献和延伸阅读 | 384 |

<<软件质量经济学>>

版权说明

本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问:<http://www.tushu007.com>