

<<软件开发的创新思维>>

图书基本信息

书名：<<软件开发的创新思维>>

13位ISBN编号：9787505383784

10位ISBN编号：7505383787

出版时间：2003-1

出版时间：电子工业出版社

作者：库伯

页数：293

字数：233

译者：刘瑞挺

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>

<<软件开发的创新思维>>

内容概要

这不是一本教你如何编程的书，但却是软件工程师不可不读的书。

本书作者Cooper被誉为“VB之父”，他针对基于电脑的产品难学、难用且难以理解的弊病，提出了软件设计的新理念、新思维、新方法，以及交互设计、目标导向、角色分析和脚本提纲等一系列创新概念。

本书还对软件文化、团队组织和进程管理做了精辟而生动的论述，书中通过大量的商务案例，并用幽默的语言，向读者透彻地描述了软件业面临的挑战，指导我们如何回应并取得成功。

本书观点独特，描述生动，实例丰富，分析深刻，方法具体而有效，是软件项目经理、软件工程师及开发人员，通信及电子产品设计师，IT界、工业界管理人员以及大专院校师生值得一读的好书。

<<软件开发的创新思维>>

作者简介

艾兰·库伯：在数字化的人群中，艾兰·库伯以“ Visual basic之父 ” 而著称。1992年，艾兰等人共同创建了库伯交互设计公司，以公司为基地，这位具有洞察力的软件设计师与编程家，肩负的使命是，阐明当前软件开发过程中的“设计差距”，由于对使用产品的实际用户考虑不周，开发过程经常在接近完成的时候归于失败。

1994年，比尔·盖茨授予艾兰视窗先锋奖，以表彰他为软件产业所作的贡献。

<<软件开发的创新思维>>

书籍目录

第1篇 电脑的逆向文化

第1章 信息时代之谜 3

1.1 电脑与飞机失事 4

1.2 我不会用数字相机 5

1.3 把电脑闹钟扔出窗外 7

1.4 保时捷汽车熄火的故事 9

1.5 自动取款机拒绝我 9

1.6 当电脑与军舰“杂交”时，会得到什么？
10

1.7 电脑容易造成麻烦 11

1.8 商务软件也受到损害 13

1.9 技术的愤怒 14

1.10 业界受到否定 15

1.11 本书的缘起 16

第2章 认知的摩擦 19

2.1 与物理力没有联系的行为 20

2.2 交互设计与程序设计 21

2.3 程序员与设计师的关系 23

2.4 大多数软件是意外设计的 23

2.5 “交互”设计与“界面”设计的比较 24

2.6 瑞士小军刀与遥控汽车钥匙 25

2.7 跳舞的熊 27

2.8 功能的成本 29

2.9 发烧友与报怨者 31

2.10 我们如何应对认知摩擦 35

2.11 消费势力的民主化 36

2.12 责备用户 37

2.13 软件的等级隔离 39

第2篇 耗费大量时间

第3章 金钱的浪费 45

3.1 最后期限管理 46

3.2 “完成了”的软件是什么样？
47

3.2.1 帕金森定律 48

3.2.2 从未交付的产品 49

3.3 推迟交货并无危害 50

3.4 功能表的讨价还价 51

3.4.1 程序员的控制 53

3.5 功能不尽人意 53

3.6 重复及市场不可预测的神话 55

3.7 坏软件的隐性成本 58

3.7.1 写坏软件比写软件更昂贵 59

3.7.2 机会的价值 60

3.8 原型法的价值 61

第4章 跳舞的熊 67

<<软件开发的创新思维>>

- 4.1 如果有问题，不能立刻解决掉吗？
68
- 4.2 消费类电子产品的受害者 68
- 4.3 电子邮件程序如何失败 70
- 4.4 调度程序如何失败 71
- 4.5 日记程序如何失败 71
- 4.6 大规模的Web歇斯底里症 72
- 4.7 软件出了什么错？
74
 - 4.7.1 软件的忘性 74
 - 4.7.2 软件的懒惰 74
 - 4.7.3 软件对信息的吝啬 75
 - 4.7.4 软件不灵活 75
 - 4.7.5 软件责备用户 76
 - 4.7.6 软件不负责任 76
- 第5章 消费者的忠诚 79
 - 5.1 可期望性 80
 - 5.2 比较 83
 - 5.3 市场时机 87
- 第3篇 用叉子喝汤
- 第6章 作茧自缚 91
 - 6.1 越俎代庖 92
 - 6.2 孕育悲剧的故事 94
 - 6.3 电脑与人的比较 99
 - 6.4 把狗教成猫 101
- 第7章 逻辑人 107
 - 7.1 登机之路 108
 - 7.2 电脑程序员心理学 110
 - 7.3 程序员用可控性来换取简单性 111
 - 7.4 程序员只想求知而不要成功 112
 - 7.5 程序员注重可能性而不是合理性 115
 - 7.6 程序员像捣蛋鬼 117
- 第8章 过时的文化 121
 - 8.1 编程的文化 122
 - 8.2 代码复用 123
 - 8.3 共同的文化 127
 - 8.3.1 微软的编程文化 128
 - 8.4 文化的孤立性 132
 - 8.5 问题的关键 133
 - 8.5.1 稀缺性思维 135
 - 8.6 是过程而不是技术使人丧失人性 137
- 第4篇 交互设计好商机
- 第9章 为欢乐而设计 141
 - 9.1 角色 142
 - 9.2 只为一个人设计 143
 - 9.2.1 带轱辘的手提箱和可粘贴的便条 144
 - 9.3 弹性用户 145

<<软件开发的创新思维>>

- 9.4 特定性 146
- 9.5 假设 148
- 9.6 精确, 而不准确 148
- 9.7 对技术水平的实际了解 149
- 9.8 角色终结功能的辩论 151
 - 9.8.1 设计师和程序员都需要角色 153
- 9.9 是用户角色, 而不是买方角色 154
- 9.10 角色的预测 155
- 9.11 首要角色 157
- 9.12 案例分析: 索尼传媒公司的“P@ssport” 158
 - 9.12.1 传统的解决办法 159
 - 9.12.2 角色 162
 - 9.12.3 为克利维斯设计 165
- 第10章 为效能而设计 169
 - 10.1 目标是我们完成任务的缘由 170
 - 10.2 任务不是目标 171
 - 10.2.1 从事任务导向设计的程序员 172
 - 10.3 目标导向的设计 173
 - 10.3.1 目标导向的电视新闻 174
 - 10.3.2 目标导向的课堂管理程序 175
 - 10.4 个人的与实际的目标 176
 - 10.4.1 相称尽力的原理 177
 - 10.5 个人目标 178
 - 10.6 公司目标 179
 - 10.7 实际目标 180
 - 10.8 虚假目标 181
 - 10.9 电脑也是人 182
 - 10.10 为彬彬有礼而设计 183
 - 10.10.1 什么是礼貌? 184
 - 10.11 使软件有什么礼貌? 185
 - 10.11.1 礼貌软件对我关心 186
 - 10.11.2 礼貌软件对我尊重 187
 - 10.11.3 礼貌软件对我友好 187
 - 10.11.4 礼貌软件通情达理 188
 - 10.11.5 礼貌软件能预知我的需要 188
 - 10.11.6 礼貌软件善于应对 189
 - 10.11.7 礼貌软件对其个人问题沉默寡言 189
 - 10.11.8 礼貌软件又要信息灵通 190
 - 10.11.9 礼貌软件应有感知 190
 - 10.11.10 礼貌软件应有自信 191
 - 10.11.11 礼貌软件能坚持集中 191
 - 10.11.12 礼貌软件能规避问题 192
 - 10.11.13 礼貌软件立即令人满足 194
 - 10.11.14 礼貌软件值得信赖 194
 - 10.12 案例研究: ES公司的“鼓声” 195

<<软件开发的创新思维>>

- 10.12.1 调查研究 196
- 10.12.2 谁为谁服务 198
- 10.12.3 设计方案 200
- 10.12.4 退一步 201
- 10.12.5 其他话题 202
- 第11章 为人而设计 205
 - 11.1 脚本 206
 - 11.2 日常使用型脚本 207
 - 11.3 必需使用型脚本 207
 - 11.4 边缘情况的脚本 208
 - 11.5 调节界面 209
 - 11.6 永久的中间程度 210
 - 11.7 自诩神奇 213
 - 11.8 词汇表 214
 - 11.8.1 语言的突破 215
 - 11.9 侧向思维 216
 - 11.10 案例研究：Logitech公司的“孔雀” 217
 - 11.10.1 麦尔考，Web勇士 219
 - 11.10.2 马启迪，小男孩 219
 - 11.10.3 曼努姆，DPI 219
 - 11.10.4 扮演自诩神奇 221
 - 11.10.5 一流的修整 223
 - 11.10.6 一流的图像缩放 224
 - 11.10.7 一流的图像重新定向 226
 - 11.10.8 一流的结果 228
 - 11.11 连接软硬件的桥梁 228
 - 11.12 少些才会更好些 230
- 第5篇 恢复主控权
- 第12章 不顾一切找用途 235
 - 12.1 调整时间顺序 236
 - 12.2 用户测试 238
 - 12.2.1 用户在编程之前测试 239
 - 12.2.2 在测试过程中加入可用性测试 239
 - 12.3 多学科团队 240
 - 12.4 程序员做设计 241
 - 12.5 你怎么知道的？
242
 - 12.6 风格指南 243
 - 12.6.1 利益冲突 244
 - 12.7 专题组 244
 - 12.8 可视设计 245
 - 12.9 工业设计 247
 - 12.10 好酷的新技术 248
 - 12.11 重复 248
- 第13章 进程的管理 251

<<软件开发的创新思维>>

- 13.1 谁真正拥有最大的影响力？
252
- 13.1.1 顾客驱动的死亡螺线 252
- 13.1.2 概念的完整性是核心能力 254
- 13.1.3 一个浮士德式的交易 255
- 13.1.4 要有一个长远观点 257
- 13.1.5 要有责任感 257
- 13.1.6 要保证足够的时间 257
- 13.1.7 要把握控制权 258
- 13.2 认清基本事实 258
- 13.2.1 要知道删减哪些功能 259
- 13.3 向电影制作学习 259
- 13.4 解决方案 262
- 13.4.1 便于制作的文档设计 263
- 13.4.2 设计影响代码 264
- 13.4.3 设计文档有益于程序员 265
- 13.4.4 设计文档有益于市场销售 267
- 13.4.5 设计文档有益于软件说明和技术支持 268
- 13.4.6 设计文档有益于管理 269
- 13.4.7 设计文档有益于整个公司 269
- 13.5 谁对产品的质量负责？
270
- 13.6 建立友好的设计进程 271
- 13.6.1 交互设计师从哪里来？
272
- 13.6.2 组建设计小组 273
- 第14章 效能和欢愉 275
- 14.1 一个运行良好的项目实例 277
- 14.2 全公司的设计意识 279
- 14.3 转变的好处 280
- 14.4 让他们吃蛋糕 281
- 14.4.1 改变这个过程 283
- 中英文名词、术语对照表 287
- 参考文献 295
- 编辑手记 297

<<软件开发的创新思维>>

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>