

## <<Java Bug模式详解>>

### 图书基本信息

书名：<<Java Bug模式详解>>

13位ISBN编号：9787302074434

10位ISBN编号：7302074437

出版时间：2003-11

出版时间：清华大学出版社

作者：艾伦 (Allen Eric)

页数：186

字数：254

译者：王蕾

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>

## <<Java Bug模式详解>>

### 内容概要

本书特点：

把软件开发的调试过程与极限编程方法融为一体

对软件实例进行深层次的剖析

为常见的调试问题提供便捷的参考清单

本书着重讨论和日常编程工作息息相关的JavaBug模式。

Bug模式是一种非常实用的概念，它能提高用户检测和修正代码错误的能力。

本书共分为三个部分：第I部分是有关Bug模式的理论基础，介绍Bug模式的基本概念及应用这种方法的意义所在；第II部分是对最为常见的13种Bug模式的详细讲解，从中可以学会如何识别、预防这些典型Bug的方法；第III部分通过表格的形式对全书内容进行了总结。

本书适合于希望通过利用Bug模式来提高代码质量和效率的开发工程师和编程爱好者。

本书是关于计算机程序诊断和调试的方法论，它把调试过程描述为科学方法的一种完美体现。

调试是人们容易轻视的一项任务。

调试技能完全独立于其他编程技能，如程序的扩展性设计和软件重用。

Eric Allen专注于调试的理论，研究调试与软件开发周期的其他阶段的相关性。

他特别强调了单元测试对于实际调试过程的重要性。

同时，Allen还指出，虽然测试和调试往往同步执行，但它们却是截然不同的两项工作。

本书的第一部分奠定了全书的理论基础。

第二部分详述在编程过程中会频繁出现的各种类型的bug模式。

对于每一种模式，Allen都对它的特征、处理方法和预防措施进行了讲解。

Bug模式出现的频率越高，被诊断和有效消除的可能性就越大。

本书的最后部分为可能出现的问题和解决方案提供了一个参考清单，并为Java开发人员提供了各种资源，包括Web站点、设计工具和参考书目。

## &lt;&lt;Java Bug模式详解&gt;&gt;

## 书籍目录

第1章 混乱环境下的灵活方法 1.1 软件设计.实现和维护的趋势 1.1.1 对于稳定.安全系统的需求增加 1.1.2 传统软件工程技术的局限性 1.1.3 开放源代码的软件项目的可利用性 1.1.4 对于跨平台语言的需求 1.2 在快节奏的社会中学习 1.3 bug模式简述 1.4 小结 第2章 Bug.规范和实现方案 2.1 bug的概念 2.2 一体性规范 2.2.1 C++ 2.2.2 Python 2.2.3 ML 2.2.4 Pascal 2.3 规范的好处 2.4 实现方案与规范的差异 2.5 利用素材建立经济有效的规范 2.5.1 通过测试来排除规范错误 2.5.2 单元测试的缺陷 2.6 小结 第3章 调试和开发过程 3.1 将调试当作科学试验 3.1.1 逐步规范化.整合并发行软件 3.1.2 在设计上尽可能保持简单 3.1.3 结对编程 3.1.4 及时的客户反馈 3.1.5 所有开发人员共享程序代码 3.1.6 对任何可能产生问题的代码进行测试 3.2 将调试测试程序并入到单元测试集 3.3 展望:面向测试的语言 3.4 小结 第4章 调试和测试过程 4.1 可测试的设计模式 4.1.1 在模型中而不是视图中保管代码 4.1.2 使用静态类型检查发现错误 4.1.3 使用中介器封装跨越断层线的功能 4.1.4 编写带有简短签名和默认参数的方法 4.1.5 使用不修改内存状态的存取器 4.1.6 通过接口定义程序外组件 4.1.7 优先编写测试程序 4.2 GlobalModel接口 4.3 小结 第5章 科学的调试方法 5.1 软件是永不磨损的机器 5.1.1 软件有多重 5.1.2 小异常引起大问题 5.2 Bug模式可以加快诊断bug的速度 5.3 小结 第6章 关于bug模式 6.1 了解bug模式的重要性 6.2 选择bug模式的原因 6.3 如何组织bug模式 6.4 Bug诊断的快速参考 第7章 RogueTile模式 7.1 Rogue Tile bug模式简述 7.1.1 症状 7.1.2 起因.解决方法和预防措施 7.2 提取代码的其他障碍 7.2.1 通用类型 7.2.2 面向方面的编程技术 7.3 小结 第8章 随处可见的空指针 8.1 空指针异常不提供任何信息 8.2 难以捉摸的空指针 第9章 Dangling Composite 模式 9.1 Dangling Composite bug模式简述 9.1.1 症状 9.1.2 起因 9.1.3 解决方法和预防措施 9.2 小结 第10章 Null Flag模式 10.1 Null Flag bug模式简述 10.1.1 症状 10.1.2 起因 10.1.3 解决方法和预防措施 10.2 健壮性和诊断证据的缺乏 10.2.1 在更好的位置处理异常 10.2.2 处理老式代码 10.3 小结 第11章 Double Descent模式 11.1 Double Descent bug模式简述 11.1.1 症状 11.1.2 起因 11.1.3 解决方法和预防措施 11.1.4 快速但不完善的修正方法 11.1.5 真正的修正方法 11.2 小结 第12章 Liar View模式 12.1 Liar View bug模式简述 12.1.1 症状 12.1.2 起因 12.1.3 解决方法和预防措施 12.2 Liars并非仅出现在GUI程序 12.3 小结 第13章 Saboteur Data模式 13.1 Saboteur Data bug模式简述 13.1.1 症状 13.1.2 语法原因 13.1.3 语义原因 13.1.4 解决办法和预防措施 13.2 小结 第14章 Broken Dispatch模式 14.1 Broken Dispatch bug简述 14.1.1 症状 14.1.2 起因 14.1.3 解决方法和预防措施 14.2 小结 第15章 Impostor Type模式 15.1 Impostor Type bug模式简述 15.1.1 症状 15.1.2 起因 15.1.3 解决方法和预防措施 15.2 混合模式 15.3 小结 第16章 Split Cleaner模式 16.1 Split Cleaner bug模式简述 16.1.1 症状 16.1.2 起因 16.1.3 解决方法和预防措施, 16.2 小结 第17章 Fictitious Implementation模式 17.1 Fictitious Implementation bug模式简述 17.1.1 症状 17.1.2 起因 17.1.3 检测Fictitious Implementation 17.1.4 解决方法和预防措施 17.2 小结 第18章 Orphaned Thread模式 18.1 Orphaned Thread bug模式简述 18.1.1 症状 18.1.2 起因 18.1.3 解决方法和预防措施 18.2 Orphaned Thread和GUI 18.3 小结 第19章 Run-on Initializatiier模式 19.1 Run-On Initializatiier bug模式简述 19.1.1 症状和起因 19.1.2 解决方法和预防措施 19.2 修正bug 19.3 小结 第20章 Platform-Dependent模式 20.1 Platform-Dependentbug模式简述 20.1.1 与供应商相关的bug 20.1.2 与版本相关的bug 20.1.3 与操作系统相关的bug 20.2 小结 第21章 诊断清单 21.1 基本概念 21.2 模式清单 第22章 用于调试的设计模式 22.1 最大化静态类型检查 22.1.1 尽可能设置final字段 22.1.2 将不可能被改写的方法设为final 22.1.3 包括作为默认值的类 22.1.4 利用已检查异常确保所有客户端程序可处理异常情况 22.1.5 定义新的异常类型来精确区分各种异常情况 22.1.6 利用特定State类 22.1.7 将类型转换和instanceof测试降至最少 22.1.8 使用Singleton设计模式帮助最小化instanceof的使用 22.2 将引入bug的可能降至最低 22.2.1 提取通用代码 22.2.2 尽可能实现纯功能性方法 22.2.3 在构造函数中初始化所有字段 22.2.4 出现异常情况时立即抛出异常 22.2.5 出现错误时立刻报告错误消息 22.2.6 尽早发现错误 22.2.7 在代码中置入断言 22.2.8 尽可能在用户可观察到的状态下测试代码 22.3 征程尚未结束 第23章 参考资料 附录 String-parsing列表构造函数 术语表



<<Java Bug模式详解>>

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>