<<OOD启思录>>

图书基本信息

书名：<<OOD启思录>>

13位ISBN编号：9787115265432

10位ISBN编号：7115265437

出版时间：2011-12

出版时间：人民邮电

作者：里尔

页数：358

译者：鲍志云

<<OOD启思录>>

内容概要

本书提供了改进面向对象设计的真知灼见。
全书共11章，总结出了60多条面向对象设计(OOD)的指导原则。
这些经验原则涵盖了从类到对象(主要强调它们之间的关系，包括关联、使用、包含、单继承、多继承)到面向对象物理设计的重要主题。
本书将帮助你理解经验原则和“设计模式”这一流行概念之间的相互作用。
你可以借助经验原则发现设计中所存在的某一方面的问题，而设计模式则提供了解决方案。

本书对各个层次的开发者都有价值，新手能借助本书走上通向面向对象编程的快车道，想提升自己的面向对象开发水准的老手则会受益于本书深具洞察力的分析。
本书提供了让你成为更好的软件开发者的途径。

作者简介

作者：(美国)里尔 (Arthur J.Riel) 译者：鲍志云

<<OOD启思录>>

书籍目录

<<OOD启思录>>

章节摘录

版权页：插图：因为，类的名字不仅意味着一组属性，还表示实体的行为。
这种数据和行为的双向联系是面向对象范型的基石之一。
一个对象一定会有如下4个重要方面：1.它自己的身份标识（可能只是它在内存中的地址）；2.它的类的属性（通常是静态的）和这些属性的值（通常是动态的）；3.它的类的行为（从实现者的角度看）；4.它的类的公开接口（从用户的角度看）。
将这一讨论置于软件开发的语境，类可以被实现为一个结构定义以及一组可以处理这个结构的操作。
在过程式语言中，任给一个函数，很容易找出数据依赖性。
只要检查函数实现并看一下所有参数、返回值以及局部变量声明的数据类型就可以了。
但是，如果你想要找出一个数据定义的函数依赖性，那你就不得不检查全部代码，寻找依赖于这个数据的函数。
而在面向对象模型中，两种依赖性（函数对数据的依赖性和数据对函数的依赖性）都现成摆明在那里了。
对象是类数据类型的变量。
它们的内部细节只对同它们的类关联的那组函数可见。
这种对内部细节的访问限制称作信息隐藏（informationhiding）。
在很多面向对象语言中，这种隐藏不是强制的，这样我们就有了第一条（也是最重要的一条）经验原则。
经验原则2.1 所有数据都应该隐藏在它所在的类内部。
违反这条经验原则意味着你不重视可维护性。
面向对象范型所带来的益处，大部分归因于在设计阶段和实现阶段始终确保信息隐藏。
如果你把数据设定为公有，那么就很难判断系统哪部分的功能依赖于这个数据。
事实上，这样一来，数据变动与函数的映射关系就和面向动作范型一模一样了。
我们不得不检查所有的函数以判断哪些函数依赖于公有数据。
有时开发者会争辩说，“我需要把这个数据设为公有，因为……”在这种情况下，开发者应该问自己，“我到底要用这个数据来做什么？
为什么不是类为我提供这个操作？
”在所有这类情况下，问题出在类缺少了一个必需的操作。
比如，考虑图2.2 中的File类。
开发者出人意料地认为，byteoffset数据成员应该是公有的，这样才能允许随机I／0访问。
但是，我们实际上需要的是执行随机访问任务的操作。
（如果你不是C程序员，那么我在这里补充说明一下：fseek和ftell和标准C库函数，用于执行文件的随机I／0访问。
）冒昧地认为“我们可以把这个数据设为公有，因为它永远也不会改变”的程序员请注意，Murphy关于编程的一条定理表明，这是第一个需要改变的数据。

<<OOD启思录>>

编辑推荐

《OOD启思录》被读者评价为“面向对象设计领域中的EffecliveC++”！
《OOD启思录》能帮助你迈入OoD殿堂：如果你是新手，《OOD启思录》将带你走上面向对象编程的快车道。
如果你是老手，《OOD启思录》深具洞察力的分析将进一步提升你的面向对象开发水准，使你成为更好的软件开发者。
提供改进面向对象设计的真知灼见！
总结61条富有启发意义的专家经验！

版权说明