

<<软件框架设计的艺术>>

图书基本信息

书名：<<软件框架设计的艺术>>

13位ISBN编号：9787115248497

10位ISBN编号：7115248494

出版时间：2011-3

出版时间：人民邮电出版社

作者：[捷] Jaroslav Tulach

页数：365

译者：王磊,朱兴

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>

<<软件框架设计的艺术>>

前言

对于《软件框架设计的艺术》，我有一种相见恨晚的感觉，相信很多读者在读后也会有同感。

多年的软件开发经验让我体会到了各种酸甜苦辣的滋味，很多开发人员对此都应感同身受。

除了这些滋味，最常出现的却是迷茫：碰到问题，却无法解决；解决问题，却无法避免同样的错误一犯再犯。

这一个个的迷茫，对每一位软件开发人员都会带来沉重的打击。

幸好，这个世界上总不乏先行者为我们点亮一盏盏明灯，也总有大师级人物为我们指点迷津。

阅读他们的著述，便有一种醍醐灌顶的顿悟，如Gamma等人的《设计模式》、Bloch的Effective Java，这些书常令我掩卷而叹：原来如此！

而此刻你手中拿到的也正是这样的一本书。

提到这本书，就不能不提到作者Jaroslav Tulach。

作为NetBeans的创始人，十多年来，他一直致力于NetBeans产品的开发，并赢得了开源社区的尊重和美誉。

这本书是他对自己十多年NetBeans开发的一个总结。

他将自己的心路历程如实记下，见证了NetBeans从IDE走向平台、从混乱的代码走向清晰的模块化架构这一不平凡的历程。

这一路走来并不平坦，但他终在每一次迷茫和困惑中找到了完善的解决方案。

在本书中，从API的设计思想、兼容性解决方案到编码的技术要点，他都娓娓道来，与我们一同分享。

读者也许会问：此书与市场上同类的设计书籍相比有何突出之处呢？

我们不妨想想，近十年蓬勃发展的开源运动，引无数英雄竞折腰。

而为什么有些开源框架或者系统能够快速拥有大量用户并维持下去，而自己开发的类库、软件却无法取得类似的成功呢？

我想告诉读者，答案尽在此书中。

这就是本书与其他设计书籍的迥异之处：它告诉读者的是一种大道，而非小技。

坦诚地说，本书不是写给初学者的，即使是有经验的开发人员，深读此书也并非易事。

但我仍然确信，不论是一位富有经验的架构师还是一位堪堪入门的初学者，只要他细心研读，就能从本书中有所收获。

本书不应读过后就束之高阁，而应常备于手边。

当遇到难思难解之事时，翻翻此书，必有意想不到之收获。

还可以将此书放于案头，闲暇之时读上几页，必有所得。

翻译本书决非一件轻松的工作，但于我，无论多苦多累，都是值得，应该说翻译本书带给我更多的是收获。

我也希望这本书可以为读者带来同样的收获。

人民邮电出版社图灵公司给了我足够的时间和自由度来翻译本书，对此致以我最真诚的谢意。

受限于自身的水平，书中若有翻译不到位或不妥之处，责任尽归于本人，烦请各位读者谅解一二。

王磊 2011年早春

<<软件框架设计的艺术>>

内容概要

本书帮助你解决API设计方面的问题，共分3个部分，分别指出学习API设计是需要进行科学的训练的、Java语言在设计方面的理论及设计和维护API时的常见情况，并提供了各种技巧来解决相应的问题。本书作者是NetBeans的创始人，也是NetBeans项目最初的架构师。相信在API设计中遇到问题时，本书将不可或缺。

本书适用于软件设计人员阅读。

<<软件框架设计的艺术>>

作者简介

Jaroslav

Tulach, NetBeans的创始人, 也是NetBeans项目最初的架构师, 有着丰富的项目开发经验, 一直致力于如何提高开发人员的设计技巧。

<<软件框架设计的艺术>>

书籍目录

目录

第一部分 理论与理由

第1章 软件开发的艺术

- 1.1 理性主义, 经验主义以及无绪
- 1.2 软件的演变过程
- 1.3 大型软件
- 1.4 漂亮, 真理和优雅
- 1.5 更好的无绪

第2章 设计API的动力之源

- 2.1 分布式开发
- 2.2 模块化应用程序
- 2.3 交流互通才是一切
- 2.4 经验主义编程方式
- 2.5 开发第一个版本通常比较容易

第3章 评价API好坏的标准

- 3.1 方法和字段签名
- 3.2 文件及其内容
- 3.3 环境变量和命令行选项
- 3.4 文本信息也是API
- 3.5 协议
- 3.6 行为
- 3.7 国际化支持和信息国际化
- 3.8 API的广泛定义
- 3.9 如何检查API的质量
 - 3.9.1 可理解性
 - 3.9.2 一致性
 - 3.9.3 可见性
 - 3.9.4 简单的任务应该有简单的方案
 - 3.9.5 保护投资

第4章 不断变化的目标

- 4.1 第一个版本远非完美
- 4.2 向后兼容
 - 4.2.1 源代码兼容
 - 4.2.2 二进制兼容
 - 4.2.3 功能兼容——阿米?变形虫效应
- 4.3 面向用例的重要性
- 4.4 API设计评审
- 4.5 一个API的生命周期
- 4.6 逐步改善

第二部分 设计实战

第5章 只公开你要公开的内容

- 5.1 方法优于字段
- 5.2 工厂方法优于构造函数
- 5.3 让所有内容都不可更改
- 5.4 避免滥用setter方法

<<软件框架设计的艺术>>

5.5 尽可能通过友元的方式来公开功能

5.6 赋予对象创建者更多权利

5.7 避免暴露深层次继承

第6章 面向接口而非实现进行编程

6.1 移除方法或者字段

6.2 移除或者添加一个类或者接口

6.3 向现有的继承体系中添加一个接口或者类

6.4 添加方法或者字段

6.5 Java中接口和类的区别

6.6 弱点背后的优点

6.7 添加方法的另一种方案

6.8 抽象类有没有用呢

6.9 要为增加参数做好准备

6.10 接口VS.类

第7章 模块化架构

.....

第三部分 日常生活

参考书目

<<软件框架设计的艺术>>

媒体关注与评论

这绝对是一本不容错过的好书，据我所知，市场上还不曾有哪本书在框架设计领域有如此深刻的阐述。

——亚马逊读者评论 这本书是作者对自己十多年NetBeans开发的一个总结。他将自己的心路历程如实记下，见证了NetBeans从IDE走向平台、从混乱的代码走向清晰的模块化架构这一不平凡的历程。

……本书与其他设计书籍的迥异之处：它告诉读者的是一种大道，而非小技。

——本书译者

<<软件框架设计的艺术>>

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>