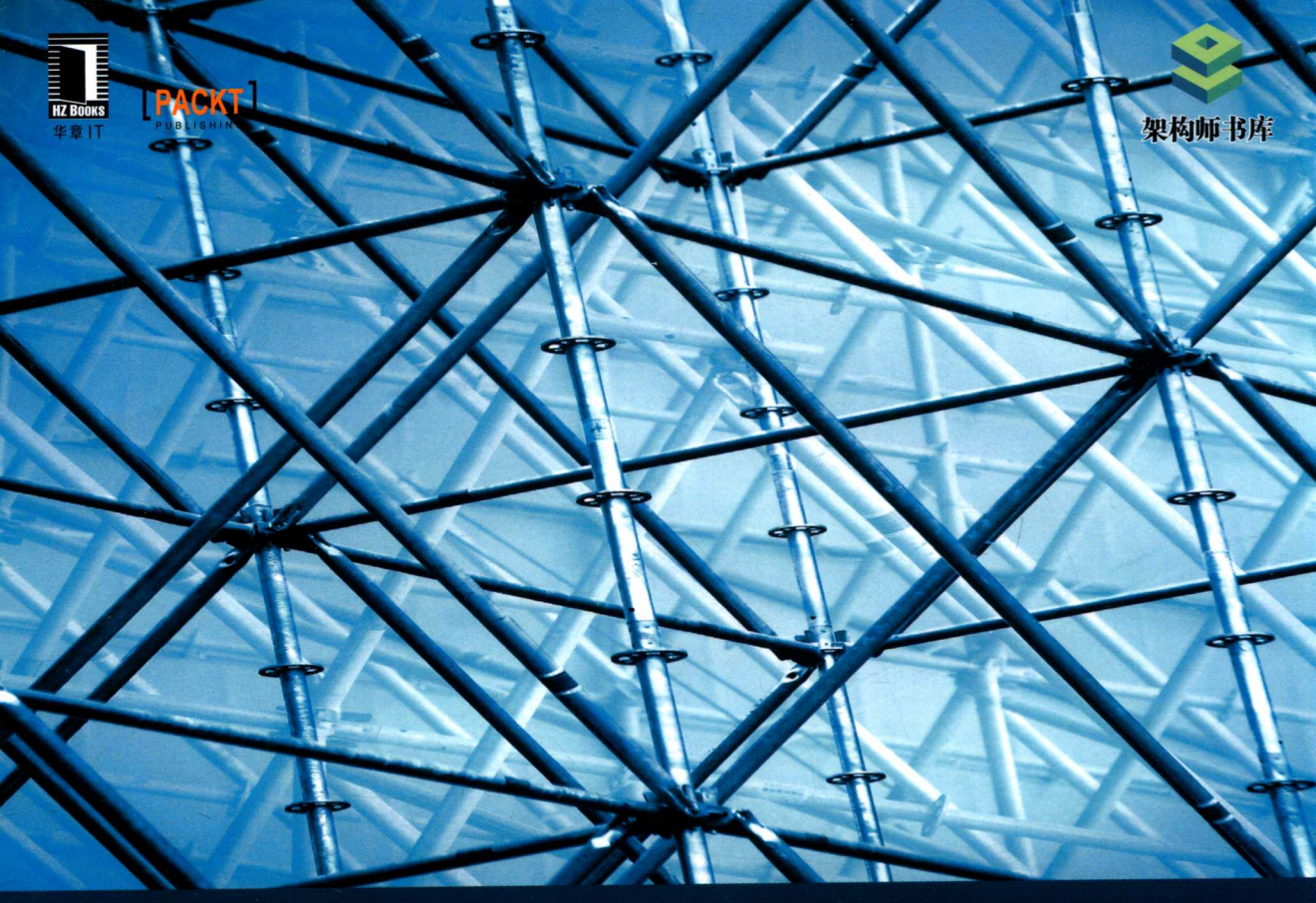

架构师书库

软件架构

Python语言实现

[印度] 阿南德·巴拉钱德拉·皮莱（Anand Balachandran Pillai）著

李必信 廖力 王璐璐 周颖 等译

SOFTWARE ARCHITECTURE
WITH PYTHON

如何用Python语言设计和构建高度可扩展、健壮、干净和高性能的软件架构产品

机械工业出版社
China Machine Press

架构师书库

SOFTWARE ARCHITECTURE
WITH PYTHON

软件架构

Python语言实现

[印度] 阿南德·巴拉钱德拉·皮莱（Anand Balachandran Pillai）著
李必信 廖力 王璐璐 周颖 等译

机械工业出版社
China Machine Press

图书在版编目（CIP）数据

软件架构：Python 语言实现 /（印）阿南德·巴拉钱德拉·皮莱著；李必信等译 . —北京：机械工业出版社，2018.2
（架构师书库）
书名原文：Software Architecture with Python

ISBN 978-7-111-59094-1

I. 软… II. ① 阿… ② 李… III. 软件工具 - 程序设计 IV. TP311.561

中国版本图书馆 CIP 数据核字（2018）第 025273 号

本书版权登记号：图字 01-2017-7518

软件架构：Python 语言实现

出版发行：机械工业出版社（北京市西城区百万庄大街 22 号 邮政编码：100037）
责任编辑：张梦玲 责任校对：李秋荣
印 刷：北京市荣盛彩色印刷有限公司 版 次：2018 年 3 月第 1 版第 1 次印刷
开 本：186mm×240mm 1/16 印 张：25
书 号：ISBN 978-7-111-59094-1 定 价：79.00 元

凡购本书，如有缺页、倒页、脱页，由本社发行部调换
客服热线：（010）88379426 88361066 投稿热线：（010）88379604
购书热线：（010）68326294 88379649 68995259 读者信箱：hzit@hzbook.com

·· 译 者 序 ··

近年来，人们对软件架构的关注度和重视程度越来越高了，具体表现在：一方面，几乎所有高等院校的计算机学院、软件学院，甚至物联网学院、网络安全学院等都在开设软件架构相关课程，而且很多高校和研究院所投入大量的人力从事软件架构的研究和开发；另一方面，大型软件企业、互联网企业和电商企业也开始非常重视软件架构的设计、实现和演化，很多著名的企业（例如，华为、腾讯、阿里巴巴、微软、IBM、谷歌等）都成立了专门的软件架构部门，甚至成立了研究院。每年的世界架构师大会、各种类型的软件架构研讨会、工作会议等，把全世界从事软件架构相关工作的人聚集在一起，为软件架构的知识体系添砖加瓦，为软件架构的未来发展出谋划策。

遗憾的是，现有的很多软件架构书籍要么只注重理论阐述、内容太过抽象，缺少软件架构实践内容；要么融合了很多与软件架构核心内容不太相关的其他知识点，使得大家在学习时无法掌握重点，甚至是无法正确地理解软件架构。由此带来的后续问题是，学生虽然学习了软件架构课程，但对软件架构的认识仍然不够深入，仍然缺乏软件架构的实践经验，走入社会后也很难承担软件架构方面的工作。

作为一本软件架构方面的书，本书的定位准确，且全面、系统地介绍了软件架构的核心内容。本书共 10 章，在阐述软件架构各个层面的知识点的同时，又重点突出了软件架构的特点和重要性，如何设计好的软件架构，软件架构对系统性能、可靠性和安全性的影响，以及软件架构与其他各种软件质量属性（例如，可测试性、可维护性等）的关系等。另外，本书的一个主要特色是，结合 Python 语言阐述了设计和实现良好软件架构的过程，是一本难得的既有软件架构的理论阐述，又有软件架构的实践案例的好书。

本书的用途和读者对象如下：①作为高等院校的教科书，面向高年级本科生和研究生；②作为软件架构研究人员的参考书，本书打穿了软件架构从设计到实现再到演化的各个环节，为实验和仿真研究提供了很好的基础；③作为软件架构师及其他工程技术人员的工具书，本书中的软件架构设计和实现案例为解决实践中遇到的架构问题具有很好的借鉴作用。

参加本书翻译工作的人员主要是来自东南大学软件工程研究所、东南大学计算机科学与工程学院的教师和部分高年级研究生，包括廖力、王璐璐、周颖、宋启威、韩伟娜、

李慧丹、谢仁松、杨安奇、杜鹏程、尹强、宋震天、葛丹薇、孔祥龙、王桐、刘辉辉、熊壬浩、邱建鹏、汪小飞、苏晓威、段鹏飞、王家慧、汤立辉等。在翻译过程中，还得到了倪巍伟、周晓宇、戚晓芳、汪鹏、张祥等老师的帮助。在此，对他们的辛勤劳动表示衷心的感谢，也特别感谢机械工业出版社张梦玲编辑的无私帮助。

限于水平，对内容的理解和中文语言表达难免存在不当之处，在此敬请读者批评指正。但无论怎样，本书是一本非常优秀的软件架构读物，本人也十分荣幸能够向读者推荐，认真地阅读本书一定使你受益匪浅。

李必信

2017 年于南京九龙湖

•• 关于作者 ••

Anand Balachandran Pillai 是一名工程技术专家，在软件企业有 18 年以上的工作经历，在产品工程、软件设计、架构设计和相关研究方面具有非常丰富的经验。

他曾获得印度理工学院机械工程专业的学士学位。曾在 Yahoo!、McAfee 和 Infosys 等公司任职，担任产品开发团队的首席工程师。

他的主要兴趣在于软件性能工程、高可扩展性架构、安全和开源社区等方面。他也经常在 Startups 工作，担任首席技术专家或顾问。

他还是班加罗尔 Python 用户联盟的奠基人和 Python 软件协会（PSF）的会士。Anand 现在是 Yegii 公司的首席架构师。

•• 关于评审人 ••

Mike Driscoll 从 2006 年开始使用 Python。他喜欢写一些关于 Python 的博客，见 http://www.blog.pythonlibrary.org/。他曾合著了《the Core Python refcard for DZone》一书，并参与了《Python 3 Object Oriented Programming》《Python 2.6 Graphics Cookbook》《Tkinter GUI Application Development Hotshot》的评审工作和其他几本书的撰写工作。他最近刚完成《Python 101》的编写，目前正在写作他的下一本书。

感谢我的妻子 Evangeline 一如既往的支持，感谢我的朋友和家人对我的无私帮助。

前　言

软件架构，可以说是为特定的应用软件创建一个蓝图设计。软件架构中存在两大挑战：首先，软件架构与需求必须保持一致，对尚未发现的需求或者发生演化的需求都是如此；其次，尽管常常发生架构实现的变更，但软件架构与其对应的架构实现必须保持一致。

本书包含很多示例和用例，通过这种直观的方法来帮助你获取成为一名成功的软件架构师所需的一切。本书将帮助你了解 Python 的来龙去脉，以便可以用 Python 来构建和设计高度可扩展的、健壮的、简洁的、性能强大的应用程序。

主要内容

第 1 章介绍了软件架构的核心思想，简要介绍了架构质量属性和一些隐含的原理。这将使你能够在软件架构原理和基本属性方面拥有良好的知识基础。

第 2 章包括开发中软件架构的可修改性和可读性。它将帮助你深入理解架构的可维护性等质量属性，并获得用 Python 编写代码来测试应用程序的各种技巧和策略。

第 3 章帮助你理解软件架构的可测试性，以及如何为 Python 应用程序构建架构以满足可测试性。你还将了解可测试性和软件测试的各个方面，以及 Python 中可用的各种库和模块，以便编写各种可测试的应用程序。

第 4 章讨论了在编写 Python 代码过程中关于性能的方方面面。你不仅可以学习架构性能的基本知识，还可以掌握在何时何地需要进行性能优化。例如，你会学习到何时进行 SDLC 的性能优化。

第 5 章不仅阐述了编写可扩展应用程序的重要性，还讨论了实现应用程序可扩展性的各种不同方法，并论述了如何利用 Python 来实现各种可扩展性技术。你不仅能学到可扩展性的理论方面的知识，还能学到业界的最佳实践。

第 6 章讨论了架构安全性的方方面面，并使你掌握一些最佳实践和技巧来编写安全性高的应用程序。你会了解在 Python 架构应用程序中可能出现的各种不同的安全问题，以及 Python 是如何从头开始保障安全性的。

第 7 章从程序员实用性的角度，简要论述了 Python 中出现的各种设计模式以及每个

模式的理论背景。这些设计模式对程序员来说是非常实用的。

第 8 章从较高抽象层次角度介绍 Python 中现有的架构模式，同时给出了几个示例，用来说明如何利用 Python 库和框架来实现基于这些模式的高层次架构问题的解决方法。

第 9 章讨论如何正确地在远程环境中或云上使用 Python 轻松部署代码的方方面面。

第 10 章讨论了一些 Python 代码调试技术，包括最简单实用的打印语句、日志记录和系统调用跟踪机制等，这些对程序员来说都是非常容易获得的，也有助于系统架构师指导他的团队。

阅读本书需要准备什么

为运行本书中展示的大部分代码示例，需要在系统中安装 Python 3。其他的预备知识会在相应的实例中提到。

本书的读者对象

本书适用于有经验的 Python 开发人员，他们渴望成为企业级应用程序的架构师；本书也适用于软件架构师，他们希望利用 Python 的特长来创建更有效的应用程序蓝图。

约定

书中的代码块设置如下：

```
class PrototypeFactory(Borg):
    """ A Prototype factory/registry class """

    def __init__(self):
        """ Initializer """

        self._registry = {}

    def register(self, instance):
        """ Register a given instance """

        self._registry[instance.__class__] = instance

    def clone(self, klass):
        """ Return clone given class """

        instance = self._registry.get(klass)
        if instance == None:
            print('Error:',klass,'not registered')
        else:
            return instance.clone()
```

当希望重点关注代码块的某个特定部分时，会将相关的行（line）或项（item）设置成

粗体：

```
[default]
exten => s,1,Dial(Zap/1|30)
exten => s,2,Voicemail(u100)
exten => s,102,Voicemail(b100)
exten => i,1,Voicemail(s0)
```

任何命令行输入或输出写成如下形式：

```
>>> import hash_stream
>>> hash_stream.hash_stream(open('hash_stream.py'))
'30fbc7890bc950a0be4eaa60e1fee9a1'
```

新术语和重要词汇以粗体形式表示。

示例代码下载

可以从 http://www.packtpub.com 下载本书的示例代码文件（需要 Packt 账户）。也可以访问 http://www.packtpub.com/support 并注册账户，Packt 将通过 email 把文件直接发送给你。

可以按照以下步骤下载代码文件：

（1）使用你的电子邮件地址和密码注册或登录到我们的网站。

（2）将鼠标指针悬停在顶部的 SUPPORT 选项卡上。

（3）单击 Code Downloads & Errata 选项。

（4）在 Search 框中输入图书的名称。

（5）选择要下载代码文件的书。

（6）从下拉菜单中选择本书的购买渠道。

（7）单击 Code Download 下载。

你还可以单击 Packt Publishing 网站中图书页面上的 Code Files 按钮下载代码文件，也可以通过在 Search 框中输入图书的名称来访问此页面。请注意，你需要首先登录 Packt 账户。

下载文件后，请确保使用这些最新版软件来解压缩文件或文件夹：

- WinRAR / 7-Zip for Windows
- Zipeg / iZip / UnRarX for Mac
- 7-Zip / PeaZip for Linux

该书的代码包也由 GitHub 托管在 https://github.com/PacktPublishing/Software-Architecture-with-Python。你还可从 Packt 提供的图书和视频目录中获取其他代码包，网址为 https://github.com/PacktPublishing/。

目　录