Springer

计算机科学本科核心课程教材

Practical Distributed Processing

分布式处理实践

Phillip J. Brooke　Richard F. Paige　著

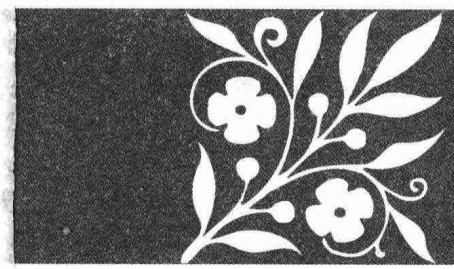
孙燕　等译

清华大学出版社

Springer

计算机科学本科核心课程教材

Practical Distributed Processing



分布式处理实践

清华大学出版社
北京

图书在版编目（CIP）数据

分布式处理实践 /（美）布鲁克（Brooke, P. J.），（美）佩奇（Paige, R. F.）著；孙燕等译．
--北京：清华大学出版社，2010.3
书名原文：Practical Distributed Processing
ISBN 978-7-302-21781-7

Ⅰ．①分…　Ⅱ．①布…②佩…③孙…　Ⅲ．①分布式处理系统　Ⅳ．①TP338

中国版本图书馆 CIP 数据核字(2010)第 001734 号

责任编辑：龙啟铭
责任校对：徐俊伟
责任印制：王秀菊

出版发行：清华大学出版社　　**地　　址**：北京清华大学学研大厦 A 座
http://www.tup.com.cn　　**邮　　编**：100084
社　总　机：010-6277017[illegible]　　**邮　购**：010-62786544
投稿与读者服务：010-62776969, c-service@tup.tsinghua.edu.cn
质　量　反　馈：010-62772015, zhiliang@tup.tsinghua.edu.cn

印 刷 者：北京季蜂印刷有限公司
装 订 者：三河市新茂装订有限公司
经　　销：全国新华书店
开　　本：185×230　　**印　　张**：14　　**字　　数**：239 千字
版　　次：2010 年 3 月第 1 版　　**印　　次**：2010 年 3 月第 1 次印刷
印　　数：1～3000
定　　价：29.00 元

译者序

在生活中，分布式系统应用已经越来越流行和普遍。分布式系统具有类似的模式：使用并发实现资源的共享和高性能；涉及同步，并协调活动时间；需要管理和减缓故障。在实际的分布式处理系统的构建中，不仅要涉及并发的常见内容（如竞争条件、信号量和互斥），还要涉及网络系统的实际实现，并考虑底层的操作系统。

本书从实用的角度讲解分布式处理的整个构建过程，以及在此过程中应用的工具、技术和原理。

根据所讲述的内容，本书大致分为如下几部分：

- 第 1 部分涉及第 1 章～第 6 章，讲解有关分布式处理的基本概念、并发的概念以及模型、操作系统的并发处理、进程通信、协议的使用。
- 第 2 部分的内容包括第 7 章～第 9 章，讲解工程问题，包括安全性和分布式处理的语言，并提供构建分布式处理的示例和案例研究。
- 第 3 部分包括第 10 章～第 11 章，第 10 章介绍一个游戏的分布式处理部分，从而将前面各章中所学的知识，特别是那些实用的内容，比如 BSD 套接字、TCP/IP 和 UDP，与复杂的分布式系统构建过程关联起来。第 11 章对全书内容加以概括总结。

本书的每一章都有相应的练习题，并在书后提供了参考答案。读者可以通过练习，巩固所学知识。

本书主要由孙燕翻译，参加翻译的还有韦笑、王雷、李志云、李晓春、陈安华、孙宏、赵成璧、侯佳宜、许伟、戴文雅、于樊鹏、刘朋、王嘉佳、李腾、邓卫、邓凡平、陈磊、李建锋、刘延军、魏宇等人。最后，祝广大读者从本书中挖掘出更多的宝藏。

前言

概述和目标

本书根据两位作者的讲稿和在分布式处理领域所做的研究而编写。在分布式处理的教学中，我们发现，不仅要涉及并发的常见内容（例如，竞争条件、信号量和互斥），还要涉及网络系统的实际实现，并考虑底层的操作系统。

关注并发原理、操作系统和编程的书有很多，但从实用的角度对分布式处理进行系统介绍的书几乎没有。因此，本书的目的是向读者清楚地展现分布式系统的整个构建过程，并介绍这个过程中应用的工具、技术和原理。该书旨在考虑这一工程过程中的重要问题，包括访问和分析分布式系统各部分时所用的模型及这些模型的实现技术（例如套接字）以及任何实际分布式系统必须考虑的有关协议和安全方面的问题。

结构和特点

本书分为 3 大部分：基础（包括原理和实际实现方面）、工程问题（包括安全性和语言方面）以及示例和案例研究。第 3 部分是前两部分的综合。

本书第 1 部分从第 1 章开始，该章对分布式处理进行了概述。分布式处理和并发的核心概念（像信号量、死锁和命名）是在第 2 章中讨论的。第 3 章提供了几种精确的并发模型，帮助读者理解第 2 章中核心概念背后的基本原理和实施方法；该章还介绍了在构建分布式系统时如何使用这些模型。第 4 章讨论操作系统以及它们在构建分布式系统中的地位。随后的第 5 章讲述进程间通信。第 6 章研究了协议，它们是构建实际分布式系统时要考虑的重要部分；该章还提供了协议示例以及设计协议时会遇到的各种挑战。

本书的第 2 部分转向工程问题。第 7 章考虑分布式系统安全性的常规问题，该章从工程的角度来探讨安全性问题。值得注意的是，安全性不仅仅是对通信进行加密，还需要对整个系统进行统筹考虑。第 8 章介绍语言的选

择，并着重于说明如何应用这些语言来构建分布式系统，以解释前面几章中出现的诸多概念，包括理论和实际两方面。

第3部分着重于示例和案例研究，旨在将前面各章结合起来。第9章通过工程生命周期来选择有关构建分布式系统的案例研究，但没有实现这些系统，目标是演示这些系统的构建过程，注重的是确定需求、甄选协议和进行设计。第10章给出了一个完整的示例：构建一个网络游戏。在该章中，首先确定了技术要求和商业要求，然后给出设计，讨论协议和安全方面，并构思实现，旨在将前面各章的内容有机地结合在一起。最后，第11章总结了本书给出了将来的一些工作、阅读资料和试验。

每一章都含有一个系统的总结，并专门给出了大量的练习题。有些练习题注重的是对当前章中问题的思考，还有一些练习题着重于构建分布式系统的一小部分内容。

本书还包含一个详尽的词汇表。

读者对象

本书针对有一定的(顺序)程序设计经验，但没有学过操作系统或并发的相关课程的学生。熟悉C语言编程对阅读和理解本书中的内容是有帮助的，但并不是必须的。那些没有太多编程经验的读者需要再学习有关一些特定编程语言的书籍(请参见参考书目)。

本书中的内容已向大二、大三的本科生传授过。通过扩充一些内容，特别是增加一个大的项目，本书也可以适用于软件工程专业的大四学生。

教师注意事项

本书的内容可以用作分布式系统的一学期课程或单元，也可以作为分布式系统要素占重要地位的有关软件工程长期课程的一部分。如果将本书用作一门单独课程，建议大致按照该书的顺序进行学习，当然，也可以在不影响整体方向的前提下，对相关章节的顺序进行调整。老师可以根据学生的背景和兴趣，有针对性地选择第3章中的某些正式模块，但强烈推荐包含状态图，因为它们非常重要，而且在第10章的完整性示例中会用到它们。对于以前已经学过操作系统相关课程的学生，可以跳过第2章中有关信号量的内容，但对于初学者来说，这部分内容是比较有用的。

作为阅读和描述项目，第9章中的案例研究是非常有价值的，它们是由课堂上的一部分学生提供的。

第10章的内容是分布式系统中较大项目的基础。实际上，我们已经基于此章中的内容来向本科生介绍研究和开发项目以及向硕士生授课。11.2.2小节给出了有关较大型和较长期项目调查研究方面的一些建议。

附录 A 包含许多练习题的概要解答或提示。示例代码，包括第 10 章中讨论的网络游戏，可以从本书的配套网站获得：http://www.scm.tees.ac.uk/p.j.brooke/dpb/。将来可能增加对第 10 章中给出的多人游戏的进一步扩充。欢迎大家提出意见和建议，并且可以直接向作者提出。

致谢

感谢英国提兹赛德大学（University of Teesside）和英国约克大学（University of York）（以及以前所在的英国普利茅斯大学[University of Plymouth]和加拿大约克大学）的同事们给出的建议、意见和有用的指导。还要感谢本书的审阅者们提出的及时、有用的建议。特别要感谢 Springer 的 Catherine Brett 和 Wayne Wheeler 在编辑和出版过程中所给的帮助以及 Ian Mackie 和 Helen Callaghan 对整个过程的启动。

本书的编写过程中使用了一些软件工具，没有这些工具，编写过程将要艰难得多。在此特别感谢 LATEX 和 Subversion 的开发者和维护者。

Phil 感谢 Christine、Rebecca 和 Alexander，他们在协助编写代码和文字方面付出了大量的精力和时间。

Rich 感谢 Angelika 对自己（和 Phil）编写、重写和测试过程的支持。

Phil 和 Rich 同时感谢审校者对编写错误的指出和改正。当然，对于未指出的错误，责任在我们作者。

Phil Brooke
Rich Paige

目录

第1章

什么是分布式处理

本章主要内容

- 分布式处理概述
- 计算和网络的发展
- 常见应用领域(数据库、文件系统、服务)
- 模型
- 移动代码

1.1 概　　述

我们一起来思考以下日常事件。假设用户站在自助银行提款机前,准备取一些钱。先是插入银行卡,输入有关信息(例如密码),选择所需的交易(例如,取款并打印凭条),接着选择要取的金额,几秒钟后,用户会收到钱和交易凭条。这是一个简单的日常事务,许多人都依赖这一过程,但现场背后的处理过程是非常复杂的。用户借助通信基础设施、计算机软件和安全机制,及时获得所需响应(以及所取现金)。

上面的银行系统就是一种分布式系统。这种系统包括网络计算机上的多个组件(可以是硬件、软件或两者都是)。这些组件通过消息进行通信,并通过消息协调活动和传递结果,例如用户所取的现金。

当今基于计算机的系统通常都是分布式系统。在日常生活中,有许多分布式系统的实例:

- Internet 就是分布式系统的一种典型实例:它将许多独立的计算机连接起来,并定义了运行在这些计算机上的软件进行通信的标准协议。标准应用程序,例如 Wed 浏览器、文件程序、对等文件共享应用程序和计算机游戏,都使用 Internet 进行操作。
- 当我们去一个旅行社预定假日行程,比如坐飞机旅行,旅行社会使用订票系统来查找两座城市之间的可用航班。这种系统称为SABRE[81],它搜索不同航空公

司的分布式数据库，找出不同类型的机票，并为顾客预订机票。整个过程用时很短。

- Search for Extraterrestrial Intelligence(SETI)是一种大型分布式系统，它利用个人计算机(连接到 Internet 并运行 SETI@home 软件)分析来自 Aercibo 射电望远镜的数据，查找某些模式，这些模式可以确定收到的是来自地球以外的射电信号，这些信号说明地球外有可能存在高智能的生命。这种分布式程序特别有意思的地方是它利用闲置的时钟周期，也就是闲置的计算资源。

本书关注的是构建分布式系统过程中会遇到的实际问题，特别是着重介绍构建实用分布式系统时需要的基本理论、原理和技术，本书较后的章节(例如第 9 章)中，通过构建一些有趣的重要系统，举例说明如何将这些理论、原理和技术结合起来。

1.2 计算和网络的发展

传统计算是单个机器运行由程序员和管理员提供的单机任务。在 20 世纪 70 年代，程序员开始认识到将计算机组成网络带来的价值(以及灵活性)。将计算机组成网络就是将机器通过通信基础设施(例如以太网电缆或无线网络)连接在一起。Tanenbaum 在谈及从集中式的大型机到分布式的而功能依然强大的小型机的计算机发展过程时，将这称为“计算机和通信的融合(merging of computers and communications)”[76]。

分布式系统和软件利用计算机网络：网络提供部分基础结构，供分布式系统在其上执行(在第 4 章中，我们将看到，操作系统提供部分的其他关键基础设施)。此外，计算机网络对用户来说，通常就是一组能够相互通信的计算设备。分布式系统通常提供一致的接口，使得用户可以将一组计算设备看作单个的统一实体。

虽然本书着重于介绍分布式系统，但是，阐明计算机网络的某些方面也是很重要的，特别是网络基础设施是如何用来实现和支持强大可靠的分布式系统的。

1.3 分布式处理

分布式系统有 3 项基本特点，这 3 项特点也是本书余下内容的关键章节。

- **并发**：在实际的分布式系统中，多个线程同时运行。比如，当你的网络浏览器正在从某个网络服务器下载内容时，可能也在确定如何将该内容显示在你的计算机屏幕上。同时，网络浏览器也正在从磁盘加载内容，以将它们通过网络发送给你。并发的复杂性的关键在于处理对资源的共享访问。例如，如果你和你的朋友同时要访问网上的某个文件，就需要有适当的机制来确保解决访问该文件时发生的冲突，譬如，你和你的朋友都同时要编辑文件。关于用来确保共享资源正确一致的

基本原理和实用方法，第 2 章中有详细的讨论。

- **在时间上同步**：在实际分布式程序中，组件和线程通常必须同步并协调它们的活动。例如，前面简略提到的 SABRE 系统，它允许旅行社为顾客查找和预订机票，显然，该系统必须有时间观念，使得如果伦敦的一家旅行社预定了格林尼治标准时间 09:00 的一张机票，虽然在巴黎该张机票的中部欧洲时间为 10:05，但巴黎的某家旅行社就不能预订这张机票。这个简单的例子说明活动的同步和协调通常依赖于活动发生的时间。当然，时钟不能做到完全精确，就像我们的手表一样，网络上同步的多个时钟也一样存在偏差。在分布式系统中，处理时间问题非常关键，后面的章节对此进行了简要的讨论。
- **失效**：在依赖于多个独立组件的分布式系统中，不幸的事实是这些组件会出现失效（例如，软件中的错误、硬件故障或外界的错误输入都可以导致出现失效），而且，整个系统对这些组件如何以及何时会失效几乎没有控制。因此，分布式系统设计者的责任是预见组件会出现失效，并纠正整个系统中的潜在失效。在许多领域中，拥有一个能够处理组件失效的分布式系统（也就是拥有一个可靠的分布式系统）非常重要，例如前面讨论的银行提款机系统。有关那些可以用来改善整体可靠性的技术，包括分布式系统的精确模型和冗余，本书会一直贯穿讨论。与失效有着特殊相关性的是安全，这部分内容将在第 7 章中详细讨论。

1.4　应用领域

我们在前面提到了几个分布式系统的例子。如今，分布式系统随处可见，而且通常使用在银行业（例如进行交易和清算账目）、电子商务、计算机游戏、嵌入式系统、生物信息（例如生物模型的分析）、移动设备（比如个人数字助理、手机和数码相机）和许多其他领域。

1.5　模　　型

虽然当今使用的分布式系统各式各样（而且我们已经讨论了一些重要的例子），但我们可以将这些系统归于几类模型。

- **客户-服务器模型**（也称为双层模型）最普遍，通常来说，使用的中央服务器和客户机可以是一个或多个，这些服务器维护共享信息，比如某个数据库，客户机则根据需要访问共享信息。实际上，服务器提供统一的公开界面，通过该界面提供服务，而客户机是根据需要使用这些服务。最为广泛的客户-服务器模型的应用是文件传输协议（File Transfer Protocol，FTP），它用于在两台计算设备之间传输文件。