



分布式 Java应用

基础与实践

林昊 著



電子工業出版社
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY
<http://www.phei.com.cn>



分布式 Java应用

基础与实践

林昊 著

电子工业出版社

Publishing House of Electronics Industry

北京 • BEIJING

内 容 简 介

本书介绍分布式 Java 应用涉及的知识点,分为基于 Java 实现网络通信、RPC;基于 SOA 实现大型分布式 Java 应用;编写高性能 Java 应用;构建高可用、可伸缩的系统 4 个部分,共 7 章内容。作者结合自己在淘宝网的实际工作经验展开论述,既可作供初学者学习,也可供同行参考。

未经许可,不得以任何方式复制或抄袭本书之部分或全部内容。
版权所有,侵权必究。

图书在版编目(CIP)数据

分布式 Java 应用:基础与实践 / 林昊著. —北京:电子工业出版社, 2010.6

ISBN 978-7-121-10941-6

I. ①分… II. ①林… III. ①JAVA 语言—程序设计 IV. ①TP312

中国版本图书馆 CIP 数据核字(2010)第 093814 号

策划编辑:徐定翔

责任编辑:杨绣国

印 刷:北京市天竺颖华印刷厂

装 订:三河市鑫金马印装有限公司

出版发行:电子工业出版社

北京市海淀区万寿路 173 信箱 邮编 100036

开 本:787×1092 1/16 印张:18.5 字数:350 千字

印 次:2010 年 6 月第 1 次印刷

印 数:5 000 册 定价:49.80 元

凡所购买电子工业出版社图书有缺损问题,请向购买书店调换。若书店售缺,请与本社发行部联系,联系及邮购电话:(010) 88254888。

质量投诉请发邮件至 zltz@phei.com.cn, 盗版侵权举报请发邮件至 dbqq@phei.com.cn。

服务热线:(010) 88258888。

献给我想共度一生的爱人宗伟

分布 & 分享

分布式计算不是一门年轻的技术，早在上个世纪 70 年代末便已是计算机科学的一个独立分支了；它也不是一门冷僻的技术，从 C/S 模式到 P2P 模式，从集群计算到网格计算，乃至风靡当下的云计算，都是其表演的舞台。另一方面，Java 作为一门应网络而生的语言，对分布式计算有着天然的友好性，同时也是当今最流行的编程语言。然而令人稍感意外的是，以“分布式 Java 应用”为专题的书籍并不多见，佳作则更少。至于国人所著者，请恕在下孤陋，尚未得见。不过细想之下，其实不足为奇。要开发一个高质量的分布式 Java 应用，以达到高性能、可伸缩、高可用、低延迟的要求，同时保证一致性、容错性、可恢复性和安全性，是何等的不易啊。它对程序开发的挑战在于：不仅需要对 Java 语言、类库、各种框架及相关工具极为熟悉，还需要对底层的 JVM 机制、各种网络协议有足够的了解；它对分析设计的挑战在于：不仅需要熟悉传统的应用架构模式，还需要掌握适用于分布式应用的架构模式和算法设计。此外，分布式应用对数据库管理员、测试分析人员等也提出了更高的要求。一本书若想涵盖这么多的内容，其难度可想而知。作者不仅需要具备起码的理论素养，更需要有丰富的实践经验。如果仅仅是纸上谈兵，对读者是无甚裨益的。

正因如此，半年多前收到林昊先生的部分书稿时，便窃感欣喜，认定这是一个很好的选题。作为淘宝网的系统架构师，他有着令人艳羡的得天独厚的实践机会。尤其难得的是，林昊亲身经历了淘宝网的快速成长和转型期，饱尝其间的成败甘苦，从中也获得了许多宝贵的第一手经验。如今，他选择将这些经验以书籍的形式与众共享，对有志于分布式应用开发的读者而言无疑是一大福音。

本书的基础部分介绍了分布式 Java 应用的基本实现方式（重点是 SOA）、相关的 JDK 类库和第三方框架，并对 JVM 的基本机制进行了深入解析；实践部分则关注于高性能、高可用和可伸缩系统的构建等。全书文风朴实，并附有大量的代码、数据和图表，比较符合大多数程序员的口味，也非常具有实践指导意义。如果用挑剔的眼光看，本书在深度、广度和高度上还有继续改进的余地，比如：对关键性的并发设计和算法设计可以介绍得更深入些；尚未涉及分布式应用中的安全问题；在性能调优一章中对 Java 源码级别的优化介绍似嫌不足；未能高屋建瓴地总结和提炼出分布式应用独有的编程和设计原则、架构和思维模式，等等。当然，对于这样一本上乏经典可参、下需躬身实践的书籍来说，以上多属苛求。事实上，本人在审稿过程中也是获益良多。

分布式 Java 应用：基础与实践

另值一提的是，林昊先生利用业余时间写作，已是经年有余，其间数易其稿，且从善如流，充分体现了技术人员的求道精神。本书的主题在明是**分布**，在暗则是**分享**——分享一段成长经历、分享一份宝贵经验，无论对作者还是读者，都是善莫大焉。

郑晖

<http://blog.zhenghui.org/>

《冒号课堂——编程范式与 OOP 思想》作者

2010 年 5 月于广州

Get Architecture Done

提起诸如“高性能”、“高可用性”、“大规模并发”、“可扩展性”这些词汇，我相信多数技术人的心情都是激动而稍有点复杂的，当然，也或许是不屑一顾。毕竟不是谁都有机会面对这些富有挑战的技术场景，也不是每个架构师在面对这些挑战之前都能做好技术上的准备。那些意外故障总是不期而至，疲于奔命地解决问题的场景回顾起来对架构师来说犹如一场噩梦。

本书阐述当一个面向数以亿计用户的网站经过几年高速发展，技术团队不得不面临大规模、高并发、高扩展性等挑战带来的技术困境的时候，一个出色的架构师经过多年一线实践后累积的经过时间考验的解决方案以及宝贵的实战经验。在这本书里，你会看到作者在解决一些关乎 Web 应用问题的指导原则、实践方法、多重工具的综合运用以及作者本人的感悟。要强调的是，本书讲述的内容是一个 Web 应用从小到大过程中遇到的棘手问题的解决之道，并非宏观解析，亦非屠龙之技。无论您面对的站点是大是小，皆会有参考作用，毕竟大站点会越来越复杂，而小站点总有一天也将变大。

如今到计算机书店里走一下，会发现 Java 架构相关的技术图书虽已不少，但仍有理由相信本书内容填补了在 Java 架构实战方面的空白。在互联网应用大行其道的今天，有些名义上主题为 Java 架构的图书，要么单从 Java 本身阐述，缺乏整体应用的大局观；要么是高屋建瓴，从编程思想的高度坐而论道，缺乏实践性；要么是闭门造车之作，缺乏验证性。本书作者林昊多年来致力于推动 OSGi 在国内的发展，不乏理论功底，而后加盟淘宝网(Taobao.com)的几年间奋战在架构一线，爬摸滚打积累了丰富的实践心得。所以，本书是一本不折不扣的“理论结合实践”之作。

考虑国内的技术图书出版环境以及必须尽力适应读者的预期，写书本身是一件十分耗费心力的事情，但能将知识传递给更多人无疑又是让人快乐的。现在，经过作者近两年的梳理与总结，这本书即将出版，相信读者在研读本书之后会有所收获并可运用到所面对的 Web 应用上，也期待将来有更多朋友能够分享架构实践经验，一展才华，不亦快哉！

冯大辉

<http://dbanotes.net>

2010 年 5 月于杭州贝塔咖啡

分布式 Java 应用：基础与实践

实践是最好的成长 发表是最好的记忆

—— 作者序

分布式 Java 应用需要开发人员掌握较多的知识点，通常分布式 Java 应用的场景还会对性能、可用性以及可伸缩有较高的要求，而这就意味着开发人员需要掌握更多的知识点。我刚进淘宝的时候，曾经一直苦恼对于一个这样的分布式 Java 应用，我到底需要学习些什么。

随着在淘宝工作的不断开展，我的眼前终于慢慢呈现了高性能、高可用以及可伸缩的 Java 应用所需知识点的全景，这张知识点的全景图现在已经演变成了本书的目录。当看到自己整理出的知识点的全景图时，很惊讶地发现其中有些知识点其实是我之前已经学习过的，但到了真正需要使用的时候有些是完全遗忘了，有些则是在使用时碰到了很多的问题，从这里我看到，当学习到的知识不去经过实践检验时，这些知识就不算真正属于自己。

幸运的是，在淘宝我得到了分布式 Java 应用的绝佳实践机会，于是所学习到的网络通信、高性能、高并发、高可用以及可伸缩的一些知识，有机会在实践中得到验证。正是这样的机会，让我对这些知识点有了更深刻本质的理解，并能将其中的一些知识真正吸收，变为自己的经验，所以我一个很真切的体会就是：**实践是最好的成长**。

经历了这段艰难的成长，自己也希望不要忘却在这个过程中的收获，胡适先生曾说：“发表是最好的记忆”（这句话也长期放在台湾技术作家侯捷老师的网站上），于是萌生了写这本书的想法，一方面想梳理自己通过实践所获得的成长，另一方面也希望与正在从事分布式 Java 应用的技术人员分享一些实践的心得，同时给将要或打算从事分布式 Java 应用的技术人员提供一些参考。

从 2008 年 11 月确认要写这本书，到 2010 年 5 月完成这本书，历时一年半，过程可谓波折不断，前 3 个月的写作一帆风顺，顺利完成了第 1 章和第 3 章的撰写。

到了 2009 年 3 月底后，由于投入到了《OSGi 原理与最佳实践》一书的编写中，停顿了将近 3 个

月的时间。

2009 年 8 月重新开始了这本书的撰写，在 2009 年 10 月下旬前按计划完成了第 2 章、第 4 章和第 5 章，但随后由于项目进入冲刺阶段、忙于校园招聘以及迁居等事情，再度停顿了本书的撰写。

记得在刚开始写这本书的时候，周筠老师就告诉我，要坚持写，就算每天只写一点也是好的，千万不要停顿！确实如此，在停顿了两次后，就很难再找到继续写这本书的动力和激情了，得感谢徐定翔编辑在之后给我的鼓励和督促，终于在 2010 年 3 月我又开始了写作。待完成了第 6 章的编写后，由于剩下的第 7 章中的部分知识点和自己的工作联系不是非常紧密，导致了不断的拖延，这时周筠老师和徐定翔编辑给我的一个建议起到了关键作用，就是先放下第 7 章，先做之前完成的六章的定稿。

回到自己熟悉的前 6 章，终于再次有了写作的动力，随着工作中对自己书中所涉及的知识点的不断实践，此时再回头看自己半年前甚至一年前写的初稿时，发现其中有不少的错误以及条理不够清晰的地方，于是进行了大刀阔斧的改动，实践所获得的积累此时起了巨大的推动作用，这 6 章定稿除了第 3 章以外的章节，完成得较为顺利。

6 章定稿提交后，由白爱萍编辑带领的编辑小组再次对书稿进行细致的“田间管理”，给出了非常多的修改建议，印象中几乎平均每页都至少有两到三处需要修改的地方，正是他们的认真和专业，使得定稿中很多语言上以及技术上的错误得以纠正，感谢武汉博文的编辑们。

最后，在第 3 章定稿的修改过程中，得到了同事莫枢（<http://rednaxelafx.javaeye.com>）非常多的建议和帮助，在此非常感谢他的支持。

全书在编写的过程中，初稿、定稿也提交给了一些业内专家帮忙评审，主要有：郑晖、霍炬、曹祺、刘力祥，他们给出的很多意见一方面纠正了书中一些技术上的错误，另一方面也让书的条理性更加顺畅，衷心感谢他们的辛勤付出。尤其郑晖老师，在承诺为本书写推荐序时，又花时间把全稿通读一遍，他的耐心和专业精神，让我感佩不已。

书的撰写过程是如此漫长，每天晚上下班后、周末、假期，甚至过年期间，都成了写书的时间，感谢家人给我的包容、支持和理解，最要感谢的是我明年春节就将迎娶的准老婆：宗伟，感谢你忍受了我不断忘记买家里需要的各种东西，感谢你独立完成了新家的装修，更要感谢你允许我这么多的周末、假日都无法陪伴你，没有你的支持和鼓励，这本书是无法完成的。

回顾整个编写过程，从开始编写，到提交完全部终稿，经历了 15 个月的时间，写作时间大概为 11 个月，经过这 11 个月对这些知识点的不断实践、回顾和总结，它们在我的脑海中刻下了深深的烙印，的确，发表是最好的记忆。

林昊

2010 年 5 月 20 日晚于杭州家中

高效写作，敏捷出版

—— 出版人感言

和林昊认识了五六年了。他还在深圳的时候，来武汉出差，我们俩一起吃一大盘船帮鱼头，憨憨的他，话不多，就知道笑。

说话间，他的第二本书就要上市了。这本书，我更多地放手让编辑去做，当然，我扮演的还是黑头黑面的教练。编辑们在手记里都提到了被我“强力推动”做某事。

天下无难事，天下无易事。难，是因为问题没有被细分到可以去解决的程度；易，是因为找到了解决问题的办法。

很多初涉写作的朋友，因为读书多，读一流书还不少，眼界一高，就看不上自己初出茅庐的文笔。这种追求完美的心态，导致很多人的写作没效率。其实，谁刚开始学走路就姿势优美呢？当编辑这么多年，不知见过多少作者毁在这追求一步走到金字塔塔尖的误区中了。可惜当时的我不明就里，没有带给他们切实的帮助。

写作，其实可以是高效的，前提是：学会接受自己刚试手时的不完美，每天都不能停止写。

高效的写作取决于高效的时间管理，高效的时间管理离不开专注与灵活这两种能力。所有与我们合作成功的作者，都是很忙的人，但他们忙得有效率，既专注又灵活。专注，让他们懂得抓主要矛盾；灵活，让他们知道借力，能随时理解并配合出版社的各项要求。

写作是件很辛苦的事情，而且看起来似乎金钱上的回报未必高（除了某些畅销书）。但是，所有在博文出书的作者，他们都深谙写作的好处。有哪些好处呢？

1. 写作犹如教书，作者与读者教学相长。因为想把道理写给人看，要写明白，就会发现其实自己脑子里还有好多模糊点没搞清楚。因为要教读者，所以只能自己搞得更清楚。为了给读者一碗水，本来只有半桶水的作者，不得不让自己变得拥有一桶水。您瞧，这是谁赚了？

2. 写作，光教人怎么干不算高明，那是授人以鱼；还要告诉人为什么要这么干，这是授人以渔。上升到规律层面，就得要求作者有较扎实的理论功底，能把一件事说圆了。而在这个过程中，不少作者会痛苦地发现自己的系统思维不足、理论功底不扎实。这就迫使他们不断思考，直到能把事情说得让

自己和读者都信服。这就很好地训练了作者的系统思维能力。系统思维能力的提升，让一个人的视野更开阔，站得更高，看得更远。试想，又是谁赚了？

3. 现在，已经进入了个人品牌时代。想拿张名片就让人对你刮目相看，土！至少得有个博客吧。看看现在，IT 圈的朋友们，有几个没博客的？但，有几个人能写书咧？物以稀为贵。写了一本好书的人，声名远播，个人价值会放大。所以，现在愿意写书的人越来越多。

好处明白了，怎样才能做到高效写作？怎样才能做到敏捷出版？我们和作者们进行了一些探讨，虽然这方面的认识积累还不够，但，有一说一。比如，在和出版社共同确定了图书产品的市场定位后（需求分析的过程也颇熬人，编辑手记里有细述），作者可以尝试：

1. 对自己的写作进行难度分级管理。忙的时候写难度低的，闲下来就写难度高的。这样就能保持写作的平稳状态，不至于忙的时候完全不写，闲的时候写一大堆，起伏太大就不易坚持。难度分级越细，越容易对写作进行模块化管理。写好一个模块就丢在那里，等着日后拼装。书也是个产品，把各个模块拼装成一本书的过程，和拼装宜家的家具没啥两样。

2. 每天得坚持写，给自己规定每日最低写作量。我们通常希望作者的每日写作量不要低于 500 字。区区 500 字，也就是三篇微博的字数之和。能坚持每日为自己的主题写 500 字的，就一定能把这个最低量渐渐提高到 1000 字。

3. 每个模块需要有写作的时间量限制，比如每 5 天必须完成一个小节，每 15 天必须完成一章中的一到两个核心小节，等等。

上述讲的是原创书的写作，其实在翻译和制作外版书时，这种模块化、敏捷运作的思路同样可以借鉴。我们用老办法做出来的产品，以往常常存在着周期和质量问题。最近我们出版的《编程之魂》就因为翻译质量挨骂了，这是因为这本书的制作还是采用的老模式，编辑和译者都单枪匹马，得不到及时的交流和帮助。挨骂不可怕，谁做产品会一帆风顺？重要的是通过思考，找到正确的方法。挨骂还是因为方法不对，方法不对，还是因为思考没到位。

林昊的这本书，我们还有不少工作没做到位，只是尽可能把团队现有的能力都贡献出来了，没有偷懒。这是进步。以往尽管也有能力不足之处，但偷懒也处处可见。

做林昊的书，再次体会到淘宝文化的魅力。开放的淘宝、重视知识积累和传承的淘宝，让我们淘到了不少林昊这样的“宝贝”作者，他们心态开放，接纳批评、消化批评的能力很强。我多次和林昊说，他的谦虚，他的开朗，他的认真，让我和同事们受益匪浅。

闻道有先后，写作有专攻。期待着更多的朋友和我们一起分享“高效写作，敏捷出版”带来的愉悦——快乐因创造价值而生。

有创造，就会有真正的成长。

周筠

2010 年 5 月 21 日于武汉

前言

软件系统得到用户认可后，访问量通常会产生爆发性的增长，互联网网站，例如淘宝、豆瓣等更是如此。在不断完善功能和多元化发展的同时，如何应对不断上涨的访问量、数据量是互联网应用面临的最大挑战。

对于用户而言，除了功能以外，网站访问够不够快，网站能否持续提供服务，也是影响用户访问量的重要因素，因此如何保证网站的高性能，及高可用也是我们要关注的重点。

为了支撑巨大的访问量和数据量，通常需要海量的机器，例如现在的 google 已经拥有了百万台以上的机器，这些机器耗费了巨大的成本（硬件采购成本、机器的电力成本、网络带宽成本等）。网站规模越大，运维成本就越高，这意味着商业公司所能获得的利润越低，而通常这会导致商业公司必须从多种角度关注如何降低网站运维成本。

本书的目的是从以上几个问题出发，介绍搭建高性能、高可用以及可伸缩的分布式 Java 应用所需的关键技术。

目标读者

本书涵盖了编写高性能、高可用以及可伸缩的分布式 Java 应用所需的知识点，适合希望掌握这些知识点的读者。

在介绍各个知识点时，作者尽量结合自己的工作，分享经验与心得，希望能够对那些有相关工作经验的读者有所帮助。

内容导读

本书按照介绍的知识点分为五个部分：第一部分介绍基于 Java 实现系统间交互的相关知识，这些知识在第 1 章中进行介绍；第二部分为基于 SOA 构建大型分布式 Java 应用的知识点，这些在第 2 章中介绍；第三部分为高性能 Java 应用的相关知识，这些在第 3、4、5 章中介绍；第四部分介绍高可用 Java

应用的相关知识，这些在第 6 章中介绍；第五部分介绍可伸缩 Java 应用，这些在第 7 章介绍，读者也可根据自己的兴趣选择相应的章节进行阅读。

网站业务多元化的发展会带来多个系统之间的通信问题，因此如何基于 Java 实现系统间的通信是首先要掌握的知识点，在本书的第 1 章中介绍了如何基于 Java 实现 TCP/IP (BIO、NIO)、UDP/IP (BIO、NIO) 的系统间通信、以及如何基于 RMI、Webservice 实现系统间的调用，在基于这些技术实现系统间通信和系统间调用时，性能也是要考虑的重点因素，本章也讲解了如何实现高性能的网络通信。

在解决了系统之间的通信问题后，多元化的发展带来的另外一个问题是多个系统间出现了一些重复的业务逻辑，这就要将重复的业务逻辑抽象为一个系统。这样演变后，会出现多个系统，这些系统如何保持统一、标准的交互方式是要解决的问题，SOA 无疑是首选的方式，在本书的第 2 章中介绍了如何基于 SOA 来实现统一、标准的交互方式。

高性能是本书关注的重点，Java 程序均运行在 JVM 之上，因此理解 JVM 是理解高性能 Java 程序所必须的。本书的第 3 章以 Sun Hotspot JVM 为例讲述了 JVM 执行 Java 代码的机制、内存管理的机制，以及多线程支持的机制。执行代码的机制包含了 Sun hotspot 将 Java 代码编译为 class 文件，加载 class 文件到 JVM 中，以解释方式执行 class，以及 client 模式和 server 模式编译为机器码方式执行 class 的实现方式；内存管理的机制包含了 Sun hotspot 内存分配以及回收的机制，内存分配涉及的主要为堆上分配、TLAB 分配以及栈上分配。内存回收涉及的有常见的垃圾回收算法、Sun Hotspot JVM 中新生代可用的 GC、旧世代可用的 GC 及 G1；多线程支持的机制包含了多线程时资源的同步机制，以及线程之间交互的机制。

除 JVM 外，在编写分布式 Java 应用时，通常要使用到一些 Sun JDK 的类库。如何合理地根据需求来选择使用哪些类，以及这些类是如何实现的，是编写高性能、高可用的 Java 应用必须掌握的。在本书的第 4 章中介绍了 Sun JDK 中常用的集合包的集合类、并发包中的常用类（例如 ConcurrentHashMap、ThreadPoolExecutor）、序列化/反序列化的实现方式，同时对这些类进行了基准的性能测试和对比。

掌握 JVM 和使用到的类库是编写高性能 Java 应用的必备知识，但除了编写之外，通常还会面临对已有系统进行调优。调优是个非常复杂的过程，在本书的第 5 章中介绍了寻找系统性能瓶颈的一些方法以及针对这些瓶颈常用的一些调优方法。寻找性能瓶颈的方法主要是根据系统资源的消耗寻找对应问题代码的方法，常用的一些调优方法主要是降低锁竞争、降低内存消耗等。

除了高性能外，高可用也是大型 Java 应用要关注的重点，在本书的第 6 章中介绍了一些构建高可用系统常用的方法，例如负载均衡技术、构建可容错的系统、对资源使用有限制的系统等。

在面对不断访问的访问量和数据量时，最希望能做到的是仅升级硬件或增加机器就可支撑，但要达到这个效果，在软件上必须付出巨大的努力。本书的第 7 章介绍了构建可伸缩系统的一些常用方法，主要包括支持垂直伸缩时常用的降低锁竞争等方法；以及支持水平伸缩时常用的分布式缓存、分布式文件系统等方法。

关于本书的代码

书中大部分代码只列出了关键部分或伪代码，完整的代码请从 <http://bluedavy.com> 下载。

关于本书的反馈

每次重看书稿，都会觉得这个知识点尚须补充，那个知识点尚须完善，但书不可能一直写下去。在提交终稿的时候，我不特别兴奋，倒是有一点遗憾和担心，遗憾书里的知识点还没有足够讲开，担心自己对某些知识点理解不够，误导了大家。为此，我在自己的网站（<http://bluedavy.com>）上开辟了一个专区，用于维护勘误，并将不断地完善书中涉及的知识点，欢迎读者反馈和交流。

联系博文

您可以通过如下方式与本书的出版方取得联系。

读者信箱: reader@broadview.com.cn

投稿信箱: bvtougao@gmail.com

北京博文视点资讯有限公司（武汉分部）

湖北省 武汉市 洪山区 吴家湾 邮科院路特 1 号 湖北信息产业科技大厦 1402 室

邮政编码: 430074

电 话: 027-87690813

传 真: 027-87690595

欢迎您访问博文视点官方博客: <http://blog.csdn.net/bvbook>

目 录

前言	1
第 1 章 分布式 Java 应用	1
1.1 基于消息方式实现系统间的通信	3
1.1.1 基于 Java 自身技术实现消息方式的系统间通信	3
1.1.2 基于开源框架实现消息方式的系统间通信	10
1.2 基于远程调用方式实现系统间的通信	14
1.2.1 基于 Java 自身技术实现远程调用方式的系统间通信	14
1.2.2 基于开源框架实现远程调用方式的系统间通信	17
第 2 章 大型分布式 Java 应用与 SOA	23
2.1 基于 SCA 实现 SOA 平台	26
2.2 基于 ESB 实现 SOA 平台	29
2.3 基于 Tuscany 实现 SOA 平台	30
2.4 基于 Mule 实现 SOA 平台	34
第 3 章 深入理解 JVM	39
3.1 Java 代码的执行机制	40
3.1.1 Java 源码编译机制	41
3.1.2 类加载机制	44
3.1.3 类执行机制	49
3.2 JVM 内存管理	63

3.2.1	内存空间	63
3.2.2	内存分配	65
3.2.3	内存回收	66
3.2.4	JVM 内存状况查看方法和分析工具	92
3.3	JVM 线程资源同步及交互机制	100
3.3.1	线程资源同步机制	100
3.3.2	线程交互机制	104
3.3.3	线程状态及分析	105
第 4 章	分布式 Java 应用与 Sun JDK 类库	111
4.1	集合包	112
4.1.1	ArrayList	113
4.1.2	LinkedList	116
4.1.3	Vector	117
4.1.4	Stack	118
4.1.5	HashSet	119
4.1.6	TreeSet	120
4.1.7	HashMap	120
4.1.8	TreeMap	123
4.1.9	性能测试	124
4.1.10	小结	138
4.2	并发包 (java.util.concurrent)	138
4.2.1	ConcurrentHashMap	139
4.2.2	CopyOnWriteArrayList	145
4.2.3	CopyOnWriteArraySet	149
4.2.4	ArrayBlockingQueue	149
4.2.5	AtomicInteger	151
4.2.6	ThreadPoolExecutor	153
4.2.7	Executors	157