

# Java性能权威指南



[Java性能权威指南\\_下载链接1](#)

著者:奥克斯 (Scott Oaks)

出版者:人民邮电出版社

出版时间:2016-3-1

装帧:平装

isbn:9787115413765

市面上介绍Java的书有很多，但专注于Java性能的并不多，能游刃有余地展示Java性能优化难点的更是凤毛麟角，本书即是其中之一。通过使用JVM和Java平台，以及Java语言 and 应用程序接口，本书详尽讲解了Java性能调优的相关知识，帮助读者深入理解J

Java平台性能的各个方面，最终使程序如虎添翼。

通过阅读本书，你可以：

运用四个基本原则最大程度地提升性能测试的效果

使用JDK中自带的工具收集Java应用的性能数据

理解JIT编译器的优缺点

调优JVM垃圾收集器以减少对程序的影响

学习管理堆内存和JVM原生内存的方法

了解如何最大程度地优化Java线程及同步的性能

解决Java EE和Java SE应用程序接口的性能问题

改善Java驱动的数据库应用程序的性能

作者介绍:

Scott

Oaks是Oracle公司的一位架构师，专注研究Oracle中间件软件的性能。加入Oracle之前，他曾于Sun

Microsystem公司任职多年，在多个技术领域都有建树，包括SunOS的内核、网络程序设计、Windows系统的远程方法调用（RPC）以及OPEN

LOOK虚拟窗口管理器。1996年，Scott成为Sun公司的Java布道师，并于2001年加入Sun公司的Java性能小组——从那时起他就一直专注于Java的性能提升。此外，Scott也在O'Reilly出版社出版了多部书籍，包括Java Security、Java Threads、JXTA in a Nutshell和Jini in a Nutshell。

目录: 推荐序 xi

前言 xii

第1章 导论 1

1.1 概述 2

1.2 平台版本约定 2

1.3 全面的性能调优 4

1.3.1 编写更好的算法 4

1.3.2 编写更少的代码 4

1.3.3 老调重弹的过早优化 5

1.3.4 其他：数据库很可能就是瓶颈 6

1.3.5 常见的优化 7

1.4 小结 8

第2章 性能测试方法 9

2.1 原则1：测试真实应用 9

2.1.1 微基准测试 9

2.1.2 宏基准测试 13

2.1.3 介基准测试 15

2.1.4 代码示例 16

2.2 原则2：理解批处理流逝时间、吞吐量和响应时间 19

2.2.1 批处理流逝时间 19

2.2.2 吞吐量测试	20
2.2.3 响应时间测试	20
2.3 原则3：用统计方法应对性能的变化	23
2.4 原则4：尽早频繁测试	26
2.5 小结	28
第3章 Java性能调优工具箱	29
3.1 操作系统的工具和分析	29
3.1.1 CPU使用率	29
3.1.2 CPU运行队列	32
3.1.3 磁盘使用率	33
3.1.4 网络使用率	34
3.2 Java监控工具	35
3.2.1 基本的VM信息	36
3.2.2 线程信息	39
3.2.3 类信息	39
3.2.4 实时GC分析	39
3.2.5 事后堆转储	39
3.3 性能分析工具	39
3.3.1 采样分析器	40
3.3.2 探查分析器	41
3.3.3 阻塞方法和线程时间线	42
3.3.4 本地分析器	44
3.4 Java任务控制	45
3.4.1 Java飞行记录器	46
3.4.2 开启JFR	52
3.4.3 选择JFR 事件	54
3.5 小结	56
第4章 JIT编译器	58
4.1 JIT编译器：概览	58
4.2 调优入门：选择编译器类型（Client、Server或二者同用）	61
4.2.1 优化启动	62
4.2.2 优化批处理	63
4.2.3 优化长时间运行的应用	64
4.3 Java和JIT编译器版本	64
4.4 编译器中级调优	67
4.4.1 调优代码缓存	67
4.4.2 编译阈值	68
4.4.3 检测编译过程	70
4.5 高级编译器调优	73
4.5.1 编译线程	73
4.5.2 内联	74
4.5.3 逃逸分析	75
4.6 逆优化	76
4.6.1 代码被丢弃	77
4.6.2 逆优化僵尸代码	78
4.7 分层编译级别	79
4.8 小结	80
第5章 垃圾收集入门	81
5.1 垃圾收集概述	81
5.1.1 分代垃圾收集器	83
5.1.2 GC算法	84
5.1.3 选择GC算法	87
5.2 GC调优基础	92
5.2.1 调整堆的大小	92

- 5.2.2 代空间的调整 95
- 5.2.3 永久代和元空间的调整 96
- 5.2.4 控制并发 97
- 5.2.5 自适应调整 98
- 5.3 垃圾回收工具 99
- 5.4 小结 102
- 第6章 垃圾收集算法 103
- 6.1 理解Throughput收集器 103
- 6.2 理解CMS收集器 109
  - 6.2.1 针对并发模式失效的调优 113
  - 6.2.2 CMS收集器的永久代调优 116
  - 6.2.3 增量式CMS垃圾收集 117
- 6.3 理解G1垃圾收集器 118
- 6.4 高级调优 126
  - 6.4.1 晋升及Survivor空间 126
  - 6.4.2 分配大对象 129
  - 6.4.3 AggressiveHeap标志 136
  - 6.4.4 全盘掌控堆空间的大小 137
- 6.5 小结 138
- 第7章 堆内存最佳实践 140
- 7.1 堆分析 140
  - 7.1.1 堆直方图 141
  - 7.1.2 堆转储 142
  - 7.1.3 内存溢出错误 146
- 7.2 减少内存使用 149
  - 7.2.1 减少对象大小 149
  - 7.2.2 延迟初始化 152
  - 7.2.3 不可变对象和标准化对象 156
  - 7.2.4 字符串的保留 157
- 7.3 对象生命周期管理 160
  - 7.3.1 对象重用 160
  - 7.3.2 弱引用、软引用与其他引用 165
- 7.4 小结 175
- 第8章 原生内存最佳实践 176
- 8.1 内存占用 176
  - 8.1.1 测量内存占用 177
  - 8.1.2 内存占用最小化 178
  - 8.1.3 原生NIO缓冲区 178
  - 8.1.4 原生内存跟踪 179
- 8.2 针对不同操作系统优化JVM 182
  - 8.2.1 大页 182
  - 8.2.2 压缩的oop 185
- 8.3 小结 187
- 第9章 线程与同步的性能 188
- 9.1 线程池与ThreadPoolExecutor 188
  - 9.1.1 设置最大线程数 189
  - 9.1.2 设置最小线程数 192
  - 9.1.3 线程池任务大小 193
  - 9.1.4 设置ThreadPoolExecutor的大小 193
- 9.2 ForkJoinPool 195
- 9.3 线程同步 201
  - 9.3.1 同步的代价 202
  - 9.3.2 避免同步 205
  - 9.3.3 伪共享 208

- 9.4 JVM线程调优 211
  - 9.4.1 调节线程栈大小 211
  - 9.4.2 偏向锁 212
  - 9.4.3 自旋锁 212
  - 9.4.4 线程优先级 213
- 9.5 监控线程与锁 213
  - 9.5.1 查看线程 214
  - 9.5.2 查看阻塞线程 214
- 9.6 小结 217
- 第10章 Java EE 性能调优 218
  - 10.1 Web容器的基本性能 218
  - 10.2 线程池 222
  - 10.3 EJB会话Bean 223
    - 10.3.1 调优EJB对象池 223
    - 10.3.2 调优EJB缓存 225
    - 10.3.3 本地和远程实例 226
  - 10.4 XML和JSON处理 227
    - 10.4.1 数据大小 227
    - 10.4.2 解析和编组概述 229
    - 10.4.3 选择解析器 230
    - 10.4.4 XML验证 235
    - 10.4.5 文档模型 237
    - 10.4.6 Java对象模型 240
  - 10.5 对象序列化 241
    - 10.5.1 transient字段 241
    - 10.5.2 覆盖默认的序列化 241
    - 10.5.3 压缩序列化数据 244
    - 10.5.4 追踪对象复制 246
  - 10.6 Java EE网络API 248
  - 10.7 小结 250
- 第11章 数据库性能的最佳实践 251
  - 11.1 JDBC 251
    - 11.1.1 JDBC驱动程序 252
    - 11.1.2 预处理语句和语句池 253
    - 11.1.3 JDBC连接池 255
    - 11.1.4 事务 256
    - 11.1.5 结果集的处理 262
  - 11.2 JPA 264
    - 11.2.1 事务处理 264
    - 11.2.2 对JPA的写性能进行优化 267
    - 11.2.3 对JPA的读性能进行优化 268
    - 11.2.4 JPA缓存 271
    - 11.2.5 JPA的只读实体 276
  - 11.3 小结 277
- 第12章 Java SE API技巧 278
  - 12.1 缓冲式I/O 278
  - 12.2 类加载 280
  - 12.3 随机数 284
  - 12.4 Java原生接口 285
  - 12.5 异常 287
  - 12.6 字符串的性能 290
  - 12.7 日志 291
  - 12.8 Java集合类API 292
    - 12.8.1 同步还是非同步 293

- 12.8.2 设定集合的大小 294
- 12.8.3 集合与内存使用效率 295
- 12.9 AggressiveOpts标志 296
  - 12.9.1 替代实现 296
  - 12.9.2 其他标志 297
- 12.10 Lambda表达式和匿名类 297
- 12.11 流和过滤器的性能 300
- 12.12 小结 302
- 附录A 性能调优标志摘要 303
- 作者简介 312
- 关于封面 312
- • • • • (收起)

[Java性能权威指南\\_下载链接1](#)

标签

Java

性能优化

JVM

java

性能

计算机

java性能优化

编程

评论

失望了，内容虽然新，但都是皮毛，纵观全书，没有一个点讲的特别深入个透彻。还有

内容组织上，用让人感觉前后颠倒。最后说说翻译，不知道译者有没有自己读读句子，我小学语文不好，真的没法段这冗长不通的句子，是谷歌翻译吗？

-----  
可能自己水平较差,讲解的内容有点泛泛,阅读过程中,走神较严重。过段时间再看第二遍

-----  
翻译的真是太差了

-----  
我觉得根据目录自己上网搜资料可能更好点

-----  
不要过早优化？！

-----  
内容覆盖的不少，但大都太简单了

-----  
看了前面八章，后面关于应用层面及调用层面的调优遇到瓶颈或用到再看。后陆陆续续看完，有些用不到长长见识，重点还是在前八章

-----  
读起来不顺畅，比较浅。前面几章还不错，gc和线程池介绍的比较详细。

-----  
补漏了一些知识点，有些点还可以讲得更深入一点，整体收获一般

-----  
内容相对较新

-----  
最核心的地方还是垃圾回收器以及堆内存优化.建议看完<深入解析jvm虚拟机>相关理论部分,再看这本书相关调优实战部分~

-----  
虽然比较旧，还是能看，非常详细，以后慢慢实践了。

-----  
满满干货

-----  
相比平时只使用Java编程，借助本书，可以更细致地了解JVM的一些特性。作者对调优陈述了多次“是艺术”。一下子看到有很多参数可配，但又万事无绝对。若不是以前了解过一些粗浅的机器学习方法论，仅以做业务项目的经历可能会惊诧于“模棱两可”的说明。想到现如今运维已有AIOps，也许以后JVM调优也会有基于AI的解决方案。扯远了。调优的内容不比新学习一门语言上手写代码那样见效快。细心地去发现问题，关联书中所介绍的工具，动手实践验证，才会真正有所收获吧。

-----  
挺好的，读优化学原理系列

-----  
G1 部分还行..填补了G1中文书的空白..

-----  
对jvm有进一步了解

-----  
很高深啊，关于jvm和gc的部分，还是能够理解的，会再读！

-----  
至少详细讲了G1就比其他好

-----  
书的内容很好，比较新，翻译也不错，



-----  
[Java性能权威指南\\_下载链接1](#)

## 书评

市面上介绍Java的书有很多，但专注于Java性能的并不多，能游刃有余地展示Java性能优化难点的更是凤毛麟角，本书即是其中之一。通过使用JVM和Java平台，以及Java语言 and 应用程序接口，本书详尽讲解了Java性能调优的相关知识，帮助读者深入理解Java平台性能的各个方面，最终使程序...

-----  
[Java性能权威指南\\_下载链接1](#)