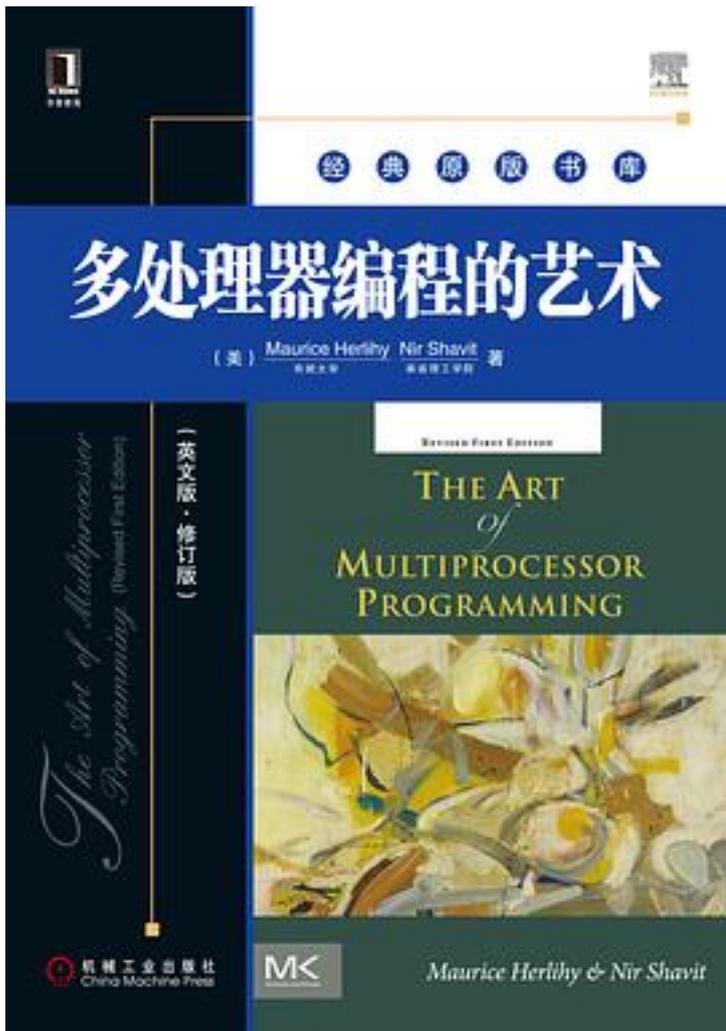


# 多处理器编程的艺术



[多处理器编程的艺术\\_下载链接1](#)

著者: (美) Maurice Herlihy

出版者:机械工业出版社

出版时间:2013-2

装帧:

isbn:9787111412335

工业界称为多核的多处理器机器正迅速地渗入计算的各个领域。多处理器编程要求理解

新型计算原理、算法及编程工具，至今很少有人能够精通这门编程艺术。

现今，大多数工程技术人员都是通过艰辛的反复实践、求助有经验的朋友来学习多处理器编程技巧。这本最新的权威著作致力于改变这种状况，作者全面阐述了多处理器编程的指导原则，介绍了编制高效的多处理器程序所必备的算法技术。了解本书所涵盖的多处理器编程关键问题将使在校学生以及相关技术人员受益匪浅。

本书特色

本修订版结合2008年第1版出版以来课堂教学和读者反馈的勘误和修改意见，对全书进行了多方面的修订和更新。

循序渐进地讲述共享存储器多线程编程的基础知识。

详细解释当今多处理器硬件对并发程序设计的支持方式。

全面考察主流的并发数据结构及其关键设计要素。

从简单的锁机制到最新的事务内存系统，独立、完整地阐述了同步技术。

利用Java并发工具包编写的可完全执行的Java实例。

附录提供了采用其他程序设计语言和包（如C#、C及C++的PThreads库）进行编程的相关背景知识以及硬件基础知识。

作者介绍:

Maurice Herlihy

哈佛大学的数学学士和麻省理工学院的计算机科学博士，目前为美国布朗大学计算机科学系教授，曾工作于卡内基-梅隆大学和DEC剑桥实验室。他是美国ACM会士，2003年分布式计算领域Dijkstra奖获得者。

Nir Shavit

以色列希伯来大学的计算机科学博士，目前为麻省理工学院电子工程和计算机科学系教授、以色列特拉维夫大学计算机科学系教授。1999~2011年期间，他担任Sun实验室的技术人员。

两位作者在2004年获得了理论计算机领域最高奖——哥德尔奖（Gödel Prize），2012年他们共享了分布式计算领域的Edsger W. Dijkstra奖，40多年来他们一起合作，从事并行和分布式计算教学和研发工作。

目录:

[多处理器编程的艺术 下载链接1](#)

标签

多核

编程

并发

计算机科学

并行计算

计算机

并行

多核编程

## 评论

懂了许多

---

[多处理器编程的艺术\\_下载链接1](#)

## 书评

TL;DR

在看这书之前，我以为这书会讲多核结构啊缓存一致性啊一类的东西，再不济讲一下各种 concurrency model 啥的，结果看完前几章后我又扫了一下目录，这才发现坑爹了。这书的关键词是锁 (locking)，是的，你没看错，就是锁，而且还是用户态的锁。以锁为核心，这书可以分...

---

深入学习一个领域最好的方法就是读这个领域权威人物写的教材，最好是每章后面带有

chapter notes的那种，共享内存并发同步领域的The Art of Multiprocessor Programming这本书就是如此，作者M. Herlihy本身就提出了并发同步相关的很多重要理论和方法。全书以一致的风格介绍领域...

-----  
This book is very new. I'm still wondering why the authors titled this book multprocessors instead of much more eye-attractive multicore. Maybe they thought they are actually the same unsolved problems. Parallelization came from very specific domains like O...

-----  
这本书是我们的课本。不知道为什么这么多人竟然说很好。个人感觉读过以后仍然一头雾水，不得不去翻翻原来的paper看，通过读由来的paper以后清楚很多。比如说第三章讲linearizable，就很不清楚。这本书只能当目录来看，里面有很多不详细的内容，还有很多错误的地方，都需要去翻...

-----  
极好的书，详尽的讲解了多线程的知识。可是，可是。我对翻译的要求不高，可是他自己可没有读懂就算了，还到处是错别字。主要是多线程的理论知识，解释了很多背后的方法，还有各种手段。但此书极好，相见恨晚，恨不得自己翻译一把。

-----  
现在在读这本书，已经看了ch.1-5, ch.8--10,有了一点发言权，谈谈自己的看法。首先，这本书挺对不起书名的，其中的错误挺多的，有些证明挺不严密的。例如，ch.2中关于mutual exclusion的证明，其隐含的前提没有指出，读起来很费解；ch.8中的关于read-write lock 代码...

-----  
可以更加深入理解操作系统调度和并发等问题，无需深刻的计算理论知识即可读懂。我是在研一花了整整十天看完的！建议后来者多结合图示理解文字算法，不懂再去看代码，这样快一些！前六章的计算理论不分不必追究细节，后面阅读时可以慢慢体会！其实学过操作系统和计算机组成完全...

-----  
[多处理器编程的艺术\\_下载链接1](#)