

# 《C++并发编程实战》

## 图书基本信息

书名：《C++并发编程实战》

13位ISBN编号：978711538732X

出版时间：2015-5

作者：Anthony Williams

页数：360

译者：周全,梁娟娟,宋真真,许敏

版权说明：本站所提供下载的PDF图书仅提供预览和简介以及在线试读，请支持正版图书。

更多资源请访问：[www.tushu111.com](http://www.tushu111.com)

# 《C++并发编程实战》

## 内容概要

《C++并发编程实战》是一本基于C++11新标准的并发和多线程编程深度指南。内容包括从std::thread、std::mutex、std::future和std::async等基础类的使用，到内存模型和原子操作、基于锁和无锁数据结构的构建，再扩展到并行算法、线程管理，最后还介绍了多线程代码的测试工作。本书的附录部分还对C++11新语言特性中与多线程相关的项目进行了简要的介绍，并提供了C++11线程库的完整参考。

《C++并发编程实战》适合于需要深入了解C++多线程开发的读者，以及使用C++进行各类软件开发的开发人员、测试人员。对于使用第三方线程库的读者，也可以从本书后面的章节中了解到相关的指引和技巧。同时，本书还可以作为C++11线程库的参考工具书。

## 书籍目录

### 目录

- 第1章 你好，C++并发世界 1
  - 1.1 什么是并发 2
    - 1.1.1 计算机系统中的并发 2
    - 1.1.2 并发的途径 3
  - 1.2 为什么使用并发 5
    - 1.2.1 为了划分关注点而使用并发 5
    - 1.2.2 为了性能而使用并发 6
    - 1.2.3 什么时候不使用并发 7
  - 1.3 在C++中使用并发和多线程 8
    - 1.3.1 C++多线程历程 8
    - 1.3.2 新标准中的并发支持 9
    - 1.3.3 C++线程库的效率 9
    - 1.3.4 平台相关的工具 10
  - 1.4 开始入门 11
  - 1.5 小结 12
- 第2章 管理线程 13
  - 2.1 基本线程管理 13
    - 2.1.1 启动线程 14
    - 2.1.2 等待线程完成 16
    - 2.1.3 在异常环境下的等待 17
    - 2.1.4 在后台运行线程 19
  - 2.2 传递参数给线程函数 20
  - 2.3 转移线程的所有权 23
  - 2.4 在运行时选择线程数量 26
  - 2.5 标识线程 28
  - 2.6 小结 29
- 第3章 在线程间共享数据 31
  - 3.1 线程之间共享数据的问题 32
    - 3.1.1 竞争条件 33
    - 3.1.2 避免有问题的竞争条件 34
  - 3.2 用互斥元保护共享数据 35
    - 3.2.1 使用C++中的互斥元 35
    - 3.2.2 为保护共享数据精心组织代码 36
    - 3.2.3 发现接口中固有的竞争条件 38
    - 3.2.4 死锁：问题和解决方案 44
    - 3.2.5 避免死锁的进一步指南 46
    - 3.2.6 用std::unique\_lock灵活锁定 51
    - 3.2.7 在作用域之间转移锁的所有权 52
    - 3.2.8 锁定在恰当的粒度 54
  - 3.3 用于共享数据保护的替代工具 56
    - 3.3.1 在初始化时保护共享数据 56
    - 3.3.2 保护很少更新的数据结构 59
    - 3.3.3 递归锁 61
  - 3.4 小结 62
- 第4章 同步并发操作 63
  - 4.1 等待事件或其他条件 63

- 4.1.1 用条件变量等待条件 65
- 4.1.2 使用条件变量建立一个线程安全队列 67
- 4.2 使用future等待一次性事件 71
  - 4.2.1 从后台任务中返回值 72
  - 4.2.2 将任务与future相关联 74
  - 4.2.3 生成(std::)promise 77
  - 4.2.4 为future保存异常 79
  - 4.2.5 等待自多个线程 80
- 4.3 有时间限制的等待 82
  - 4.3.1 时钟 83
  - 4.3.2 时间段 84
  - 4.3.3 时间点 85
  - 4.3.4 接受超时的函数 86
- 4.4 使用操作同步来简化代码 88
  - 4.4.1 带有future的函数式编程 88
  - 4.4.2 具有消息传递的同步操作 92
- 4.5 小结 96
- 第5章 C++内存模型和原子类型上操作 97
  - 5.1 内存模型基础 98
    - 5.1.1 对象和内存位置 98
    - 5.1.2 对象、内存位置以及并发 99
    - 5.1.3 修改顺序 100
  - 5.2 C++中的原子操作及类型 100
    - 5.2.1 标准原子类型 101
    - 5.2.2 std::atomic\_flag上的操作 103
    - 5.2.3 基于std::atomicbool的操作 105
    - 5.2.4 std::atomicT\*上的操作：指针算术运算 107
    - 5.2.5 标准原子整型的操作 108
    - 5.2.6 std::atomic初级类模板 109
    - 5.2.7 原子操作的自由函数 111
  - 5.3 同步操作和强制顺序 112
    - 5.3.1 synchronizes-with关系 114
    - 5.3.2 happens-before关系 114
    - 5.3.3 原子操作的内存顺序 116
    - 5.3.4 释放序列和synchronizes-with 133
    - 5.3.5 屏障 135
    - 5.3.6 用原子操作排序非原子操作 137
  - 5.4 小结 138
- 第6章 设计基于锁的并发数据结构 140
  - 6.1 为并发设计的含义是什么 141
  - 6.2 基于锁的并发数据结构 142
    - 6.2.1 使用锁的线程安全栈 142
    - 6.2.2 使用锁和条件变量的线程安全队列 145
    - 6.2.3 使用细粒度锁和条件变量的线程安全队列 149
  - 6.3 设计更复杂的基于锁的数据结构 160
    - 6.3.1 编写一个使用锁的线程安全查找表 160
    - 6.3.2 编写一个使用锁的线程安全链表 165
  - 6.4 小结 169
- 第7章 设计无锁的并发数据结构 170

- 7.1 定义和结果 171
  - 7.1.1 非阻塞数据结构的类型 171
  - 7.1.2 无锁数据结构 172
  - 7.1.3 无等待的数据结构 172
  - 7.1.4 无锁数据结构的优点与缺点 172
- 7.2 无锁数据结构的例子 173
  - 7.2.1 编写不用锁的线程安全栈 174
  - 7.2.2 停止恼人的泄漏：在无锁数据结构中管理内存 178
  - 7.2.3 用风险指针检测不能被回收的结点 182
  - 7.2.4 使用引用计数检测结点 189
  - 7.2.5 将内存模型应用至无锁栈 194
  - 7.2.6 编写不用锁的线程安全队列 198
- 7.3 编写无锁数据结构的准则 209
  - 7.3.1 准则：使用std::memory\_order\_seq\_cst作为原型 210
  - 7.3.2 准则：使用无锁内存回收模式 210
  - 7.3.3 准则：当心ABA问题 210
  - 7.3.4 准则：识别忙于等待的循环以及辅助其他线程 211
- 7.4 小结 211
- 第8章 设计并发代码 213
  - 8.1 在线程间划分工作的技术 214
    - 8.1.1 处理开始前在线程间划分数据 214
    - 8.1.2 递归地划分数据 215
    - 8.1.3 以任务类型划分工作 219
  - 8.2 影响并发代码性能的因素 222
    - 8.2.1 有多少个处理器 222
    - 8.2.2 数据竞争和乒乓缓存 223
    - 8.2.3 假共享 225
    - 8.2.4 数据应该多紧密 225
    - 8.2.5 过度订阅和过多的任务切换 226
  - 8.3 为多线程性能设计数据结构 226
    - 8.3.1 为复杂操作划分数组元素 227
    - 8.3.2 其他数据结构中的数据访问方式 228
  - 8.4 为并发设计时的额外考虑 230
    - 8.4.1 并行算法中的异常安全 230
    - 8.4.2 可扩展性和阿姆达尔定律 237
    - 8.4.3 用多线程隐藏延迟 238
    - 8.4.4 用并发提高响应性 239
  - 8.5 在实践中设计并发代码 241
    - 8.5.1 std::for\_each的并行实现 241
    - 8.5.2 std::find的并行实现 243
    - 8.5.3 std::partial\_sum的并行实现 248
  - 8.6 总结 256
- 第9章 高级线程管理 258
  - 9.1 线程池 259
    - 9.1.1 最简单的线程池 259
    - 9.1.2 等待提交给线程池的任务 261
    - 9.1.3 等待其他任务的任务 265
    - 9.1.4 避免工作队列上的竞争 267
    - 9.1.5 工作窃取 269

- 9.2 中断线程 273
  - 9.2.1 启动和中断另一个线程 274
  - 9.2.2 检测一个线程是否被中断 275
  - 9.2.3 中断等待条件变量 276
  - 9.2.4 中断在std::condition\_variable\_ any上的等待 279
  - 9.2.5 中断其他阻塞调用 281
  - 9.2.6 处理中断 281
  - 9.2.7 在应用退出时中断后台任务 282
- 9.3 总结 284
- 第10章 多线程应用的测试与调试 285
  - 10.1 并发相关错误的类型 285
    - 10.1.1 不必要的阻塞 286
    - 10.1.2 竞争条件 286
  - 10.2 定位并发相关的错误的技巧 288
    - 10.2.1 审阅代码以定位潜在的错误 288
    - 10.2.2 通过测试定位并发相关的错误 290
    - 10.2.3 可测试性设计 291
    - 10.2.4 多线程测试技术 292
    - 10.2.5 构建多线程的测试代码 295
    - 10.2.6 测试多线程代码的性能 297
  - 10.3 总结 298
- 附录A 附录A C++11部分
  - 语言特性简明
  - 参考 299
- 附录B 并发类库

## 精彩短评

- 1、翻译书中，至今看过的最烂的翻译，语句前后不通，估计译者英文水平只有初中。
  - 2、in action系列的书籍我都很喜欢，简明易懂，读起来很有效率。不过这本书的翻译的确是有问题，这么一本书搞了4个翻译者，明显是分派任务，大家都一副很不情愿的样子，很多地方句子根本不通顺，严重影响了这本书的效果。
  - 3、翻译差得气得到我三更半夜找回多年没回过的豆瓣账号专门上来评论，真是醉了，中国银行合肥支行你跟原著有仇么，毁了一本好书，英语差的人自认倒霉
  - 4、被翻译毁掉的一本书
  - 5、如果英文还不错，就直接读英文。如果觉得还是习惯读中文，就两本书一起看，看到中文意思狗屁不通的时候，就看英文。在看了一些内容以后，确实感觉很多地方翻译有问题，比如，pipeline在计算机中是流水线的意思，而不是管道的意思。译者确实犯了一些很低级的错误。
  - 6、一星完全是因为原书的面子。
  - 7、有时间还是看英文原版吧
  - 8、写的不透彻，atomic一章完全看不懂，查了各种文档网页才知道在讲什么。大概扫一遍就行了。
  - 9、书是好书，但翻译水准，校准都差。
  - 10、翻译差
  - 11、主要介绍C++11并发API，翻译超烂
  - 12、读的Gitbook上翻译的一个版本，翻译还不错。
  - 13、书是好书，翻译真是硬伤。
  - 14、翻译的巨烂，奉劝任何一个想在并发编程上精进的程序员都别买这本译作。我翻了几章，觉得原书应该确实颇有深度，以至于译者的水平完全hold不住，只能把英文一字一句的直译成中文，读起来拗口得很。我几乎需要把中文在脑中转成英文才能理解原作的意义。译者简直就是大学本科在读生的水平。央行合肥科技处真是笑话！
  - 15、不能打零星吗？
- 信达雅至少丢了两个，机器翻译的都比这个好。翻完之后，到底有没有找过人通读一遍？即使不找别人读，译者们自己也该都一遍吧？这坨汉字你们自己读得懂吗？

# 《C++并发编程实战》

## 精彩书评

1、给看这本翻译版的人几个提醒：1、要始终相信自己的理解能力，看不懂中文不是你的错。2、别太较真，除非你有时间自己翻译一遍。3、需要较真的地方一定要看原文。每一段自己找个关键词，每一节自己写个总结。记住一定要自己写，不要以为能够从翻译文字中复制几句拼起来当总结，那样会让这个总结连自己都看不懂。

2、终于上架了呀，等了2年多了吧。之前是在java并发相关的论坛看到这本书的，自己比较喜欢。苦于没有中文版，就自己硬着头皮开始阅读。英文书啃起来比较困难，本着网络精神，索性就开始翻译了。个人英文水平也不算太高，本职工作C++算法方面的工作。看到大家在吐槽发售版本的翻译问题，我也想把自己的翻译呈现给大家，希望大家在翻译和编程技术方向给一些建议，或者吐槽吧。：

) github : [https://github.com/xiaoweiChen/Cpp\\_Concurrency\\_In\\_Action](https://github.com/xiaoweiChen/Cpp_Concurrency_In_Action) 直接阅读gitbook

: [http://chenxiaowei.gitbooks.io/cpp\\_concurrency\\_in\\_action/](http://chenxiaowei.gitbooks.io/cpp_concurrency_in_action/)



# 《C++并发编程实战》

## 版权说明

本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问:[www.tushu111.com](http://www.tushu111.com)