

# 第7章 无锁并发数据结构设计

---

## 本章主要内容

- 设计无锁并发数据结构
- 无锁结构中内存管理技术
- 对无锁数据结构设计的简单指导

上一章中，我们了解了在设计并发数据结构时会遇到的问题，根据指导意见指引，确定设计的安全性。对一些通用数据结构进行检查，并查看使用互斥锁对共享数据进行保护的实现例子。第一组例子就是使用单个互斥量来保护整个数据结构，但之后的例子就会使用多个锁来保护数据结构的的不同部分，并且允许对数据结构进行更高级别的并发访问。

互斥量是一个强大的工具，其可以保证在多线程情况下可以安全的访问数据结构，并且不会有条件竞争或破坏不变量的情况存在。对于使用互斥量的代码，其原因也是很简单的：就是让互斥量来保护数据。不过，这并不会如你所想的那样；你可以回看一下第3章，回顾一下死锁形成的原因，再回顾一下基于锁的队列和查询表的例子，看一下细粒度锁是如何影响并发的。如果你能写出一个无锁并发安全的数据结构，那么就能避免这些问题。

在本章中，我们还会使用原子操作(第5章介绍)的“内存序”特性，并使用这个特性来构建无锁数据结构。设计这样的数据结构时，要格外的小心，因为这样的数据机构不是那么容易正确实现的，并且让其失败的条件很难复现。我们将从无锁数据的定义开始；而后，将继续通过几个例子来了解使用无锁数据结构的意义，最后给出一些通用的指导意见。