

构建自定义的同步工具

- 状态依赖性的管理
 - 将前提条件的失败传递给调用者
 - 通过轮询与休眠实现简单的阻塞
 - 条件队列
- 使用条件队列
 - 条件谓词
 - 过早唤醒
 - 丢失信号
 - 线程必须等待一个已经为真的条件,但在开始等待之前没有检查条件谓词
- 通知
 - 每当在等待一个条件时,一定要确保在条件谓词变为真时,通过某种方式发出通知
 - 只有同时满足以下两个条件时,才能使用单一的notify而不是notifyAll
 - 所有等待线程的类型都相同
 - 只有一个条件谓词与条件队列相关
 - 每个线程从wait返回后执行相同的操作
 - 单进单出
 - 在条件变量上每次通知,最多只能唤醒一个线程来执行
 - 阀门类 ThreadGate
- 子类的安全问题
 - 将其等待和通知等协议向子类完全公开
 - 完全禁止子类化
- 封装条件队列
- 入口协议与出口协议
 - 对于每个依赖状态的操作,以及每个修改其他操作依赖状态的操作,都应该定义一个入口协议和出口协议
 - 入口协议就是该操作的条件谓词

- 出口协议包括,检查被该操作修改的所有状态变量,确认它们是否使某个其他条件谓词变为真,如果是,则通知相关条件队列
- 显式Condition
- Condition与Lock
- `java.util.concurrent.AbstractQueueSynchronizer`