

## 附录B 并发库简要对比

虽然，C++11才开始正式支持并发，不过，高级编程语言都支持并发和多线程已经不是什么新鲜事了。例如，Java在第一个发布版本中就支持多线程编程，在某些平台上也提供符合POSIX C标准的多线程接口，还有Erlang支持消息的同步传递(有点类似于MPI)。当然还有使用C++类的库，比如Boost，其将底层多线程接口进行包装，适用于任何给定的平台(不论是使用POSIX C的接口，或其他接口)，其对支持的平台会提供可移植接口。

这些库或者编程语言，已经写了很多多线程应用，并且在使用这些库写多线程代码的经验，可以借鉴到C++中，本附录就对Java，POSIX C，使用Boost线程库的C++，以及C++11中的多线程工具进行简单的比较，当然也会交叉引用本书的相关章节。

特性	启动线程	互斥量	监视/等待
章节引用	第2章	第3章	第4章
C++11	std::thread和其成员函数	std::mutex类和其成员函数	std::condition_variable类和其成员函数
std::lock_guard<>模板	std::condition_variable_any类和其成员函数	std::atomic<>类模板	std::shared_mutex类模板
std::unique_lock<>模板	std::atomic_thread_fence()函数	std::atomic_future<>类模板	
Boost线程库	boost::thread类和成员函数	boost::mutex类和其成员函数	boost::condition_variable类和其成员函数
boost::lock_guard<>类模板	boost::condition_variable_any类和其成员函数	boost::shared_future<>类模板	
boost::unique_lock<>类模板			

POSIX C	pthread_t类型相关的API函数	pthread_mutex_t类型相关的API函数	pthread_cond_t类型相关的API函数
pthread_create()	pthread_mutex_lock()	pthread_cond_wait()	
pthread_detach()	pthread_mutex_unlock()	pthread_cond_timed_wait()	
pthread_join()	等等	等等	
Java	java.lang.Thread类	synchronized块	java.lang.Object.wait()和notify()方法在内部synchronized块中调用