

FastTimer 的设计方案

FastTimer 具有以下特点：

1. 不提供 repeat 操作。

通常计时器的设计分为 Timeout timer 和 heart beat timer。Timeout timer 在超时的时候只触发一次；heart beat timer 则是使用 repeat 持续触发。Repeat 持续触发的方法将会不停地向队列抛出 timer 触发消息，对性能有影响。处于需求和性能原因，FastTimer 的设计不支持 repeat，如果需要重新触发 Timer，需要重新启动 Timer。

2. 在正常情况下，FastTimer 对性能影响不大。

一个标准定时器在使用过程中，包括以下几个步骤：

- (1) new Timer;
- (2) start Timer;
- (3) stop Timer;
- (4) delete Timer。

对此，FastTimer 采用在栈上分配的方法，减少 New 和 delete 操作对性能的影响。

此外，对于 start 和 stop，通常计时器使用树的方法来设计，性能不高。FastTimer 设计了一个 FastTimerQueue 来提高性能，具体设计见 FastTimerQueue::start。

3. FastTimer 适用于对精度要求不高的情况。

一个典型应用场景：在 Connection 的设计中，如果 recv() 没有 timeout，server 长时间没有回应，recv() 将一直等待下去。为处理这种情况，需要实现带有超时功能的 timed_recv()，并为此设计一个计时器，在 recv 超时的时候发送一个 wakeup 消息。

FastTimer 设计方案

通过 Timer 队列（FastTimerQueue）和 Timer 服务线程（TimerService）来实现一个快速 Timer（FastTimer）：

FastTimerQueue 是一组 slot 构成的队列（见 TimerQueue 结构图）。其中，每一个 slot 包含了一个 Timer 队列和一个 mutex。

FastTimerQueue 在初始化的时候定义 slot 的个数和定时器的触发时间间隔 precision（即 Timer 的精度[1]）。

注：[1] FastTimerQueue 的 precision 设置不能小于 10ms，否则将占用过多的 CPU 资源，默认设置为 0.5 秒。

TimerService 每过一个 precision 的时间就会使 FastTimerQueue 中的当前 slot(current slot) Timer 队列上的 Timer 有机会触发[2]，同时 current slot 将会赋值为下一个 slot。

注：[2] 如果 Timer 从加入到 FastTimerQueue 的时刻开始算起，在队列中的时间已经达到了 Timeout 时间，则触发 Timer，Timer 在触发后将从 Timer 队列中清除；如果还没有达到 Timeout 的时间，Timer 将不会被触发。

FastTimerQueue::start(FastTimer* timer, duration_type timeout)

将一个 timer 加入到 TimerQueue 的方法:

1. 获取当前 FastTimerQueue 的 current slot
2. 计算 FastTimerQueue 运行几轮 (cycle) 后, 此 Timer 才会达到 Timeout 时间。
(TimerQueue 运行一轮的时间是: precision*slot 的个数)
3. 计算该 Timer 在 FastTimerQueue 中的 slot 位置:(current slot + timeout / m_precision)% 的个数
4. 锁住 slot, 将 Timer 加入该 slot 的 Timer 队列。

FastTimerQueue::tick()

TimerService 每过一个 precision 的时间会运行一次 tick():

1. 获取当前 FastTimerQueue 的 current slot, 将 current slot+1
2. 遍历当前 Timer 队列, 触发已经达到 Timeout 时间的 Timer (即 cycle 等于 0 的 Timer); 还没有到达 Timeout 时间的 Timer, 将其 cycle - 1

FastTimerQueue::restart(FastTimer* timer, duration_type timeout)

重新启动一个已经加入到 FastTimerQueue 中的 Timer。(FastTimer 将不提供 repeat 操作)

FastTimerQueue 结构图

