链滴

一个轻量级的信号量

作者：localvar
原文链接：https://ld246.com/article/1522203463187
来源网站：链滴

基于SRWLOCK和条件变量（Condition Variable）实现，需要windows vista及以上操作系统，编时_WIN32_WINNT要大于0x0600。基本功能和用CreateSemaphore创建出来的差不多，不过不支跨进程使用，也模拟不了WaitForMultipleObjects的情形。和系统自带的那个相比一般会快一点，具结果取决于硬件，有些机器上能快十几倍，有些机器则只稍快一丁点。请注意我说的快十几倍也仅仅指这两个实现本身的差别，而不是说程序的总体性能提高这么多。具体是否使用，大家可以根据自己测试结果来决定。代码和使用应该都很简单，就不多解释和举例了。

```
class CSlimSemaphore
{
private:
    SRWLOCK m_lock;
    CONDITION_VARIABLE m_cv;
    LONG m_value;
    LONG m_maximum;

public:
    CSlimSemaphore( LONG lInitialCount, LONG lMaximumCount )
    {
        InitializeSRWLock( &m_lock );
        InitializeConditionVariable( &m_cv );
        m_value = lInitialCount;
        m_maximum = lMaximumCount;
    }

    ~CSlimSemaphore() { }

    BOOL Release( LONG lReleaseCount, LONG* lpPreviousCount )
    {
        BOOL succeeded = FALSE;

        AcquireSRWLockExclusive( &m_lock );

        if( m_value + lReleaseCount &lt;= m_maximum )
        {
            if( lpPreviousCount != NULL )
                *lpPreviousCount = m_value;
            m_value += lReleaseCount;
            succeeded = TRUE;
        }

        ReleaseSRWLockExclusive( &m_lock );

        if( succeeded )
            WakeAllConditionVariable( &m_cv );
        else
            SetLastError( ERROR_TOO_MANY_POSTS );

        return succeeded;
    }

    DWORD Wait( DWORD dwTimeout )
    {
        DWORD result = WAIT_TIMEOUT;
        DWORD start = GetTickCount();
```

```
        AcquireSRWLockExclusive( &m_lock );

        while( true )
        {
            if( m_value &gt; 0 )
            {
                --m_value;
                result = WAIT_OBJECT_0;
                break;
            }

            if( dwTimeout != INFINITE )
            {
                DWORD end = GetTickCount();
                if( end - start &gt;= dwTimeout )
                    break;
                dwTimeout -= end - start;
                start = end;
            }

            if( !SleepConditionVariableSRW( &m_cv, &m_lock, dwTimeout, 0 ) )
            {
                if( GetLastError() != ERROR_TIMEOUT )
                    result = WAIT_FAILED;
                break;
            }
        }

        ReleaseSRWLockExclusive( &m_lock );

        return result;
    }
};
```

原文链接：一个轻量级的信号量