一个轻量级的事件对象

作者：localvar
原文链接：https://ld246.com/article/1522204063133
来源网站：链滴

“事件”对象的一个轻量级实现，方法与“信号量”基本一样。

```
class CSlimEvent
{
private:
    SRWLOCK m_lock;
    CONDITION_VARIABLE m_cv;
    BOOL m_manual;
    BOOL m_state;

public:
    CSlimEvent( BOOL bManualReset, BOOL bInitialState )
    {
        InitializeSRWLock( &m_lock );
        InitializeConditionVariable( &m_cv );
        m_manual = bManualReset;
        m_state = bInitialState;
    }

    ~CSlimEvent() { }

    BOOL Set()
    {
        AcquireSRWLockExclusive( &m_lock );
        m_state = TRUE;
        ReleaseSRWLockExclusive( &m_lock );

        if( m_manual )
            WakeAllConditionVariable( &m_cv );
        else
            WakeConditionVariable( &m_cv );

        return TRUE;
    }

    BOOL Reset()
    {
        AcquireSRWLockExclusive( &m_lock );
        m_state = FALSE;
        ReleaseSRWLockExclusive( &m_lock );
        return TRUE;
    }

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

        AcquireSRWLockExclusive( &m_lock );

        while( true )
        {
            if( m_state )
            {
```

```
            m_state = m_manual;
            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;
  }
};
```