NAME

scond_t − Shore Condition Variable Class

SYNOPSIS

```
#include <sthread.h>

/*
 *  Condition Variable
 */
class scond_t : public sthread_named_base_t  {
public:
    NORET                 scond_t(const char* name = 0);
    NORET                 ~scond_t();

    w_rc_t                wait(
       smutex_t&              m,
       int4_t                 timeout = WAIT_FOREVER);
    void                  signal();
    void                  broadcast();
    bool                  is_hot() const;

};
```

DESCRIPTION

Threads usually wait on a conditional variable because they can only continue after a certain condition is met (for example, a consumer thread might wait for the condition that the input queue is not empty). Every condition variable should be protected by a **smutex_t.**

scond_t(name)

The constructor creates a condition variable. The *name* parameter is stored in the condition variable for debugging purposes.

~scond_t()

wait(mutex, timeout)

The **wait** method suspends the current thread, which must hold *mutex,* on the condition variable and releases *mutex.* Later, when the condition variable is *signaled,* the thread is awakened and it will reacquire *mutex* before returning from **wait.**

signal()

The **signal** method wakes up *at least one* thread waiting on the condition variable.

broadcast()

The **broadcast** method wakes up *all* threads waiting on the condition variable.

is_hot()

The **is_hot** method returns **true** if at least one thread is waiting on the condition.

ERRORS

TODO.

EXAMPLES

TODO.

VERSION

This manual page applies to Version 2.0 of the Shore Storage Manager.

SPONSORSHIP

The Shore project is sponsored by the Advanced Research Project Agency, ARPA order number 018 (formerly 8230), monitored by the U.S. Army Research Laboratory under contract DAAB07-91-C-Q518. Further funding for this work was provided by DARPA through Rome Research Laboratory Contract No. F30602-97-2-0247.

COPYRIGHT

SEE ALSO

errors(sthread), sthread_t(sthread), smutex_t(sthread), sevsem_t(sthread), file_handlers(sthread), intro(sthread).