NAME

mt-streams – STREAMS multithreading

SYNOPSIS

#include <sys/conf.h>

INTERFACE LEVEL

Solaris DDI specific (Solaris DDI).

DESCRIPTION

STREAMS drivers configures the degree of concurrency using the **cb_flag** field in the **cb_ops** structure (see **cb_ops**(9S)). The corresponding field for STREAMS modules is the **f_flag** in the **fmodsw** structure.

For the purpose of restricting and controlling the concurrency in drivers/modules, we define the concepts of **inner** and **outer perimeters**. A driver/module can be configured either to have no perimeters, to have only an inner or an outer perimeter, or to have both an inner and an outer perimeter. Each perimeter acts as a readers-writers lock, that is, there can be multiple concurrent readers or a single writer. Thus, each perimeter can be entered in two modes: shared (reader) or exclusive (writer). The mode depends on the perimeter configuration and can be different for the different STREAMS entry points (**open**(9E), **close**(9E), **put**(9E), or **srv**(9E)).

The concurrency for the different entry points is (unless specified otherwise) to enter with exclusive access at the inner perimeter (if present) and shared access at the outer perimeter (if present).

The perimeter configuration consists of flags that define the presence and scope of the inner perimeter, the presence of the outer perimeter (which can only have one scope), and flags that modify the default concurrency for the different entry points.

All MT safe modules/drivers specify the **D_MP** flag.

Inner Perimeter Flags

The inner perimeter presence and scope are controlled by the mutually exclusive flags:

D_MTPERQ The module/driver has an inner perimeter around each queue.

D_MTQPAIR The module/driver has an inner perimeter around each read/write pair of queues.

D_MTPERMOD The module/driver has an inner perimeter that encloses all the module's/driver's queues.

None of the above The module/driver has no inner perimeter.

Outer Perimeter Flags

The outer perimeter presence is configured using:

D_MTOUTPERIM In addition to any inner perimeter, the module/driver has an outer perimeter that encloses all the module's/driver's queues. This can be combined with all the inner perimeter options except **D_MTPERMOD**.

The default concurrency can be modified using:

D_MTPUTSHARED This flag modifies the default behavior when **put**(9E) procedure are invoked so that the inner perimeter is entered shared instead of exclusively.

D_MTOCEXCL This flag modifies the default behavior when **open**(9E) and **close**(9E) procedures are invoked so the the outer perimeter is entered exclusively instead of shared.

The module/driver can use **qwait**(9F) or **qwait_sig( )** in the **open**(9E) and **close**(9E) procedures if it needs to wait "outside" the perimeters.

The module/driver can use **qwriter**(9F) to upgrade the access at the inner or outer perimeter from shared to exclusive.

The use and semantics of **qprocson( )** and **qprocsoff**(9F) is independent of the inner and outer perimeters.

SEE ALSO

qprocson(9F), **qwait**(9F), **qwriter**(9F), **cb_ops**(9S)

STREAMS Programming Guide

Writing Device Drivers