NAME

vm_map_lock, vm_map_unlock, vm_map_lock_read, vm_map_unlock_read, vm_map_trylock, vm_map_trylock_read, vm_map_lock_upgrade, vm_map_lock_downgrade - vm_map locking macros

SYNOPSIS

#include <sys/param.h>
#include <vm/vm.h>
#include <vm/vm_map.h>

void
vm_map_lock(vm_map_t map);

void
vm_map_unlock(vm_map_t map);

void
vm_map_lock_read(vm_map_t map);

void
vm_map_unlock_read(vm_map_t map);

int
vm_map_trylock(vm_map_t map);

int
vm_map_trylock_read(vm_map_t map);

int
vm_map_lock_upgrade(vm_map_t map);

int

vm_map_lock_downgrade(vm_map_t map);

DESCRIPTION

The **vm_map_lock**() macro obtains an exclusive lock on *map*.

The **vm_map_unlock**() macro releases an exclusive lock on *map*.

The **vm_map_lock_read**() macro obtains a read-lock on *map*.

The **vm_map_unlock_read**() macro releases a read-lock on *map*.

The **vm_map_trylock**() macro attempts to obtain an exclusive lock on *map*. It returns FALSE if the lock cannot be immediately acquired; otherwise return TRUE with the lock acquired.

The **vm_map_trylock_read**() macro attempts to obtain a read-lock on *map*. It returns FALSE if the lock cannot be immediately acquired; otherwise return TRUE with the lock acquired.

The **vm_map_lock_upgrade**() macro attempts to atomically upgrade a read-lock on *map* to an exclusive lock.

The **vm_map_lock_downgrade**() macro attempts to downgrade an exclusive lock on *map* to a read-lock.

IMPLEMENTATION NOTES

Currently, all of the locking macros implement their locks as sleep locks.

SEE ALSO

vm_map(9)

AUTHORS

This manual page was written by Bruce M Simpson < bms@spc.org>.