NAME

vm_page_free, **vm_page_free_toq**, **vm_page_free_zero**, **vm_page_try_to_free** - free a page

SYNOPSIS

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

void
vm_page_free(*vm_page_t m*);

void
vm_page_free_toq(*vm_page_t m*);

void
vm_page_free_zero(*vm_page_t m*);

int
vm_page_try_to_free(*vm_page_t m*);

DESCRIPTION

The **vm_page_free_toq**() function moves a page into the free queue, and disassociates it from its object. If the page is held, wired, already free, or its busy count is not zero, the system will panic. If the PG_ZERO flag is set on the page, it is placed at the end of the free queue; otherwise, it is placed at the front.

If the page’s object is of type OBJT_VNODE and it is the last page associated with the object, the underlying vnode may be freed.

The **vm_page_free**() and **vm_page_free_zero**() functions both call **vm_page_free_toq**() to actually free the page, but **vm_page_free_zero**() sets the PG_ZERO flag and **vm_page_free**() clears the PG_ZERO flag prior to the call to **vm_page_free_toq**().

The **vm_page_try_to_free**() function verifies that the page is not held, wired, busy or dirty, and if so, marks the page as busy, drops any protection that may be set on the page, and frees it.

RETURN VALUES

vm_page_try_to_free() returns 1 if it is able to free the page; otherwise, 0 is returned.

SEE ALSO

vm_page_busy(9), vm_page_hold(9), vm_page_wire(9)

AUTHORS

This manual page was written by Chad David <*davidc@acns.ab.ca*>.