Q: What is the size of the swapfile for NEXTSTEP?

A: The size of the swapfile depends on the parameters set up in the **/etc/swaptab** configuration file and the demand for paging on the system.   The swapfile dynamically grows on demand; the demand for swapping depends on the memory needs of the running applications, and the size of the ªrealº memory (the more real memory the less the demand for paging space).

The **swaptab** file looks like this:

        #

```
#      /etc/swaptab
#
/private/vm/swapfile    lowat=20971520  # 20 Meg low water mark
```

This tells the system to use the file **/private/vm/swapfile** for swapping.   It also specifies
the parameter **lowat=20971520**, the ªlow water mark.º   This low water mark is the
smallest size to which the file shrinks.   The paging file then grows as needed to meet
demand.

A **hiwat=*n*** parameter may be used in the **swaptab** to set a limit on how large this file
grows.

Q: What size is the **swapfile** when a fresh copy of NEXTSTEP is installed ?

A: It starts out at 16MB out of the box.   It should never get smaller, since this the default size of the lowat mark in /etc/swaptab.

Q: Will the **swapfile** get smaller (if above the low water mark) while the system is running, or is it reset only upon reboot?

A: The swapfile works in a stack-oriented mannerÐit shrinks only if the system detects that all pages above a certain address are unallocated.   This rarely occurs.   Consider

launching a 4MB application which is completely paged out followed by one more page owned by some other application in the system.   Even if the 4MB is freed the swapfile cannot be truncated until that extra page is freed.   What generally happens is that long-running applications allocate a few pages at the high addresses, and **swapfile** apparently never shrinks.

During a reboot when paging is turned on (by the invocation of **mach_swapon**), the **swapfile** is immediately truncated to the minimum value.

Note: Despite the file being called the **swapfile**, NEXTSTEP uses pure demand-paged memory management.   mach_swapon(8) tells the kernel to initiate paging.

QA569

Valid for 1.0, 2.0, 3.0, 3.1