IBM **Network Computer**

Flash Memory Card Implementation for the IBM Network Station Release 3.0

Version 3.1.1 - June 10, 1999

IBM Network Computer Division

Flash memory card support was released with Network Station Manager beginning with Release 3.0.1. This is the enabling documentation for that support. **The features described in this document pertain only to IBM Network Station Manager Release 3.0.3 (aka 1.3.0.3) and later.**

This version of this document now includes information on using DHCP with flash. This has necessitated several changes, including a different method for dealing with the configuration files, **flash.nsm** and **peer.nsm**. If you currently have flash boot set up and working, there is no need to change the way you are doing things.

---

1. Introduction

Overview

The flash memory solution for the Network Station can be employed to provide boot and application loading capabilities where no local server is available. This is typical of wide-area networks where a few Network Stations are located in a remote site and it is not cost-effective to provide a boot server in each location.

Flash boot should **not** be implemented merely to reduce boot times by a few seconds in a local area network. Creating, installing, and maintaining a flash memory configuration greatly reduces the benefits of server-based management and increases the cost of maintaining a Network Station environment.

References

- *Flash Memory Card Support for the IBM Network Station Release 3.0* (http://www.ibm.com/nc)
  This is the prerequisite document for flash memory support. It provides background information on flash memory card support in the Network Station and discusses sizing and procuring flash memory cards. This document can be found at the URL listed above by selecting `Library` on the left side and the following the `White Papers` link.

- *AS/400 IBM Network Station: Techniques for Deployment in a WAN*.
  This Redbook may be ordered in the US as publication SG24-5187-00 or accessed on the web at http://www.redbooks.ibm.com. Several areas in Chapter 3 are out of date so this white paper and the document listed above should be used as the latest sources of information.

- *IBM Network Station Manager Installation and Use*.
  This document may be ordered in the US as publication SC41-0664 or accessed on the web at http://publib.boulder.ibm.com/pubs/html/netcom/r3.htm.

- *Configuration Files*.
  This document may be accessed on the web at http://publib.boulder.ibm.com/pubs/html/netcom/config.htm.

- *Full-Screen Solutions*.
  This document may be accessed on the web at http://publib.boulder.ibm.com/pubs/html/netcom/template.htm.

Document Terminology and Conventions

The terms "flash", "flash memory", "flash card", and "flash memory card" are used throughout the document to refer to the PCMCIA flash memory card. "flash boot" and "flash memory boot" are used interchangeably to refer to the process of loading a Network Station from a PCMCIA flash memory card.

The terms "kernel" and "operating system" are used interchangeably throughout the document. The Network Station operating system consists of a small, multi-tasking, UNIX-derivative kernel. It provides X Window support and also recognizes a limited set of commands to run built-in applications. Numerous extensions and libraries provide applications for emulators, browser, console, ICA client, etc.

Notes

1. The current document addresses US English only. All locale-specific information needs to be modified for other geographies. This information will be added in a later version of the document.

2. Portions of this document were taken from copyrighted NCD, Inc. material.

2. Creating Flash Memory Cards

Introduction

This section details the general process for creating a flash card to be used for booting the Network Station. This is a generic overview of the process. Appendix A provides several cookbook examples for common applications.

The steps in creating a card are as follows:

- Stage the files necessary for booting.
- Create the configuration directories/files.
- Set up the Network Station with a flash card.
- Use the Local File Manager to format the flash card.
- Use the Network File System (NFS) to access a local file system.
- Copy files to the flash card

Each step will be detailed in the sections that follow.

Throughout the remainder of this document it will be necessary to refer to path names for the locations of files. The path names will be referred to as *PRODBASE* and *PRODDIR*. *PRODBASE* is the directory name on the server while *PRODDIR* is the exported name of the directory as seen by the Network Station. These directories and the files within are created by the installation procedure on each server. Data in many of these files is maintained by the Network Station Manager (NSM).

For the supported IBM servers, the values for *PRODBASE* and *PRODDIR* are shown in the following table. Note that these values only differ on NT 4.0.

Boot Server	*PRODBASE* Path	*PRODDIR* Path
AS/400	/QIBM/ProdData/NetworkStation	/QIBM/ProdData/NetworkStation
AIX	/usr/netstation	/usr/netstation
Windows NT 4.0	/nstation/Prodbase	/netstation/prodbase
OS/390	/usr/lpp/nstation/standard	/usr/lpp/nstation/standard
VM	/QIBM/ProdData/NetworkStation	/QIBM/ProdData/NetworkStation
PCMCIA Flash Card (Local File System)	/local	/local

Please consult Chapter 10 of the _IBM Network Station Manager Installation and Use_ for more details.

Prerequisites

In addition to an IBM Network Station with a PCMCIA adapter, the following software is required in order to format and load a flash memory card for use with the Network Station:

- IBM Network Station Manager (NSM) Release 3.0.3 or later.

- Network Station boot monitor version V3.0.7 or later. This boot monitor must be loaded into the Network Station's NVRAM before it will recognize and format cards. It is supplied as a part of NSM Release 3.0.3 and later.

- If a Network Station Series 1000 is to be used, it must contain a PCMCIA adapter. This adapter can be ordered from IBM.

- A Network File System (NFS) client. This client is included with AIX. For Windows NT/95/98, a client package can be purchased from Hummingbird Communications (http://www.hummingbird.com) through their NFS Maestro Client product, or from FTP Software (http://www.ftp.com) with their Interdrive product. See Appendix D for more information on these products.

Boot Monitor Update

In order to use the flash boot capability, the Network Station boot monitor version must be at V3.0.7 or later. The boot monitor version is displayed when the system is powered on.

A boot monitor update can be accomplished simply by booting the Network Station from the server with NSM 3.0.3 or later installed. The boot monitor code will be automatically updated if not already at the V3.0.7 level.

Staging the Files

A flash memory card is built from a subset of the files in the standard distribution of the Network Station software. There are several ways to move the necessary files from the server to the flash card:

1. Create a temporary directory on the server to stage and test the files. Then copy the files to the flash card.
2. Build scripts to copy/modify the files from the server installation to the Network Station flash card.

Option 1 is recommended as it provides the ability to easily test everything prior to creating the flash card image. It also provides a way to have a backup of everything that was done. The downside is that the files in this directory could become out of date as fixes and new releases of the Network Station code are installed.

Option 2 is typically accomplished with BAT files or shell scripts and is a bit more complicated to set up. It has the advantage of keeping up with the latest code each time a card is created. It works best for simple scenarios where the number of files is small and little editing is required.

The following sections assume that files are first staged on the server. It will assume that a directory called **nsflash** has been created on the server and this directory will be populated with a subset of the files and directories (e.g. **mods**, **SysDef**, etc.) from the *PRODBASE* directories on the server. This directory will be a mirror-image of what is actually put on the flash card. The files chosen will depend on what application(s) are to be run.

Creating Configuration Files

Overview

Enabling flash boot is largely a matter of making the appropriate configuration file changes to redirect the loading of specific files from the NSM server disk to the flash memory card. Rather than directly edit the supplied configuration files, this document creates two new configuration files, *PRODBASE*/**flash/standard.nsm** and *PRODBASE*/**peer/standard.nsm** for local and peer booting respectively. *PRODBASE*/**peer/standard.nsm** is only required if multiple Network Stations are to be booted from the same flash card.

Note: Older versions of this document created the files *PRODBASE*/**configs/flash.nsm** and *PRODBASE*/**configs/peer.nsm**. However, since DHCP does not provide a method for specifying the Configuration File name (it must use the default, **standard.nsm**) those files would not work in DHCP environments. The current method supports both NVRAM and DHCP setup.

The *PRODBASE*/**flash/standard.nsm** and *PRODBASE*/**peer/standard.nsm** files first read the standard configuration files from the server (e.g. *PRODBASE***control.nsm**) and then override values as necessary. Since they are stored on the server, the configuration information can be easily changed without having to modify the flash card.

Create New Directories

On the NSM server, create two new directories, *PRODBASE*/**flash** and *PRODBASE*/**peer** using the appropriate operating system commnds or GUI. Make sure that the permissions on the directories allow them to be read/searched by everyone.

The *PRODBASE*/flash/standard.nsm File

Here is a prototype for the **standard.nsm** configuration file for the system containing the flash card. You may need to change `nfs` to `tftp` depending on your environment. You can increase the read and write block sizes from 1024 up to a maximum of 8192 if you do not have many hops between the Network Station and the server.

Copy the entire contents *PRODBASE*/**configs/standard.nsm** between the lines

```
# Begin PRODBASE/configs/standard.nsm.
```

and

```
# End PRODBASE/configs/standard.nsm.
```

Change all of the lines from that file that begin with "read" to "read /netstation/prodbase/configs/". For example, the line

```
read control.nsm
```

would be changed to

```
read /netstation/prodbase/configs/control.nsm
```

Do **not** edit *PRODBASE*/**configs/standard.nsm** directly!

There are a number of other edits that need to be made so read through the file carefully.

```
#
# standard.nsm - used to flash boot a Network Station
# Place in the PRODBASE/flash directory on the server
#
# Uncomment (remove the # from the {...} lines) for the server type being
# used for configuration and
# replace 0.0.0.0 with the IP address of that system.
set file-service-table = {
#
# The following three lines are for WindowsNT.
# Some lines are split for readability but must be in the file
# as a single line between the {...}.
#    {"/netstation/prodbase" nil 0.0.0.0 nfs
"/netstation/prodbase" unix 3 10 1024 1024}
#    {"/netstation/prodbase/configs" nil 0.0.0.0 nfs
"/netstation/prodbase/configs" unix 3 10 1024 1024}
#    {"/netstation" nil 0.0.0.0 nfs "/netstation" unix 3 10 1024 1024}
#
# The following three lines are for AIX.
# Some lines are split for readability but must be in the file
# as a single line between the {...}.
```

```
#     {"/netstation/prodbase" nil 0.0.0.0 nfs
"/usr/netstation" unix 3 10 1024 1024}
#     {"/netstation/prodbase/configs" nil 0.0.0.0 nfs
"/usr/netstation/configs" unix 3 10 1024 1024}
#     {"/usr/netstation" nil 0.0.0.0 nfs "/usr/netstation" unix 3 10 1024 1024}
#
# The following three lines are for AS/400.
# Some lines are split for readability but must be in the file
# as a single line between the {...}.
#     {"/netstation/prodbase" nil 0.0.0.0 tftp
"/QIBM/ProdData/NetworkStation" unix 3 10 1024 1024}
#     {"/netstation/prodbase/configs" nil 0.0.0.0 tftp
"/QIBM/ProdData/NetworkStation/configs" unix 3 10 1024 1024}
#     {"/QIBM/ProdData" nil 0.0.0.0 tftp
"/QIBM/ProdData" unix 3 10 1024 1024}
}
#
# Begin PRODBASE/configs/standard.nsm.
# End PRODBASE/configs/standard.nsm.
#
# Augment/override server values
set boot-desired-source = local
set boot-second-source = none
set boot-third-source = none
set config-default-file = /netstation/prodbase/flash/standard.nsm
set diag-buffer-size = 65535
set exec-startup-commands = {
    { mcuis }
#
# Modify the following line to replace 0.0.0.0 with the IP
# address of your authentication server
    { "actlogin -authserv 0.0.0.0" }
}
#
# The following lines enable peer boot
set file-enable-nfs-server = true
set file-export-directory-list = { { "/peerboot" "/local" } }
set file-nfs-access-control-default = read-only
#
# The following line enables you to access the flash card from
# each server whoes IP address is listed in the table.
# The following line is not necessary if you do not need to do remote
# maintenance. Replace the 0.0.0.0 with the IP address of the
# server.
set file-nfs-access-control-list = {
    { "/local" 0.0.0.0 read-write }
}
#
set file-try-all-matches-on-open = true
#
# Set up to get Java classes from flash
set java-directory = /local/java
#
# Set up to get executable files from flash
set modules-directory = /local/mods
#
# Get the splash screen and screensaver from flash
set pref-screen-background-bitmap-file = "/local/SysDef/ibmwall.xbm"
set pref-screensaver-bitmap-file = "/local/SysDef/ibmwall.xbm"
#
# Modify the following line to replace nil with the IP
# address of your authentication or time server. If you do not need to keep
```

```
# time locally on the Network Station or if your authentication server is
# AS/400, just leave the value as nil
set time-server = nil
set xserver-keysym-file = /local/XKeysymDB
set xserver-rgb-file = /local/rgb.txt
#
# The following statements override the standard settings
# to allow the IBM Network Station console to be used for
# debug.
#set config-enforce-passwords-locally = false
#set config-pref-enforce-passwords-locally = false
#set config-console-enforce-password-locally = false
#set exec-disabled-commands = { }
#set file-extended-diagnostics = true
#set xserver-initial-x-resources = "ncdconsole.disableReboot: false\n\
#    ncdconsole.disableLoginMenu: false\n\
#    ncdconsole.disableTerminalMenu: false\n\
#    ncdconsole.disableSetupMenu: false"
```

The *PRODBASE*/peer/standard.nsm File

Here is the Configuration File for the Network Stations which are going to peer boot from the system containing the flash card. There are a number of edits that need to be made so read through the file carefully.

```
#
# standard.nsm - used to boot peer Network Stations
# Place in the PRODBASE/peer directory on the server
#
# Uncomment (remove the # from the {...} lines) for the server type being
# used for configuration and
# replace 0.0.0.0 with the IP address of that system.
#
set file-service-table = {
#
# The following line is for WindowsNT.
# It is split for readability but must be in the file
# as a single line between the {...}.
#    {"/netstation/prodbase/flash" nil 0.0.0.0 nfs
"/netstation/prodbase/flash" unix 3 10 1024 1024}
#
# The following line is for AIX.
# It is split for readability but must be in the file
# as a single line between the {...}.
#    {"/netstation/prodbase/flash" nil 0.0.0.0 nfs
"/usr/netstation/flash" unix 3 10 1024 1024}
#
# The following line is for AS/400.
# It is split for readability but must be in the file
# as a single line between the {...}.
#    {"/netstation/prodbase/flash" nil 0.0.0.0 tftp
"/QIBM/ProdData/NetworkStation/flash" unix 3 10 1024 1024}
}

read /netstation/prodbase/flash/standard.nsm
#
# Augment/override server values
set boot-desired-source = nfs
set boot-nfs-directory = /peerboot/
set config-default-file = /netstation/prodbase/peer/standard.nsm
```

```
set file-enable-nfs-server = false
set file-try-all-matches-on-open = true
#
# Set up to get Java classes from flash
set java-directory = /peerboot/java
#
# Set up to get executable files from flash
set modules-directory = /peerboot/mods
#
# Get the splash screen and screensaver from flash
set pref-screen-background-bitmap-file = "/peerboot/SysDef/ibmwall.xbm"
set pref-screensaver-bitmap-file = "/peerboot/SysDef/ibmwall.xbm"
set xserver-keysym-file = /peerboot/XKeysymDB
set xserver-rgb-file = /peerboot/rgb.txt
```

The boot.nsl File

The **boot.nsl** file indicates to ACTLogin how to configure the File Service Table for the correct protocol to use to get to the boot server. For flash booting, this file must be in **/local/boot.nsl**. The line beginning

```
Login.bootConfigType:
```

must be changed from the default setting. Change the line to:

```
Login.bootConfigType: MOUNT_NONE
```

If this is not done, the Network Station may be unable to find files on the flash card after login.

If your **boot.nsl** file came from a Windows NT installation, you must also comment out the

```
Login.NTbootEntry: /netstation/prodbase/
```

Setting Up the Flash Card on the Network Station

In order to be recognized by the Network Station, the flash card must be in the PCMCIA slot at boot time. The card size information is read from the card during boot. If the card has been previously formatted, its file system is automatically mounted and made available. To make the flash card ready for use:

1. Power off the Network Station.
2. Ensure that the "write-protect" switch on the card is in the proper position ("Off" to format/copy files; "On" to prevent erasure/over-write).
3. Insert the flash card in the PCMCIA slot.
4. Power on the Network Station.
5. Enter the Setup Utility by interrupting the boot process during the `Search for Host System...` phase (use the `Esc` key).
6. Press `F2` to go to the View Hardware Configuration screen. The last lines on this screen should show that a valid PCMCIA flash card was detected.
7. Hit `Enter` twice to continue booting.

The Network Station should boot normally from the server. If you have diagnostic messages turned

on via the

```
F10 = Set Verbose Diagnostic Messages Enabled
```

option on the IBM Network Station Setup Utility screen, you should see the flash card be recognized and its size output during booting. The flash card is now ready for formatting through the Local File Manager, as explained below.

Formatting a Flash Card

The interface to a PCMCIA flash card is a hierarchical local file system. Like raw diskettes, new flash cards must first be formatted to be usable. Formatting creates a hierarchical file system recognized only by the Network Station. On a flash card, the root of the local file system is specified as **/local**.

Local File Manager

The Local File Manager is a utility for managing a local (flash) file system. The Local File Manager includes commands to:

- Format the local file system and verify its structure.
- Make directories, change the current working directory, and list directory content recursively.
- Copy, delete, list, and compare files.
- Display information about the local file system and the current directory.
- List Local File Manager commands.
- Quit from the Local File Manager utility.

A complete list of the Local File Manager commands is provided in Appendix B.

The Local File Manager can be started from the Console window on the Network Station or via telnet from the server.

Starting the Local File Manager Using the Network Station Console

The `Terminals` option on the Network Station Console is usually disabled. However, by using **debug.nsm** file as the `Configuration File`, this option will be enabled. To set up the Network Station boot monitor to boot from this configuration file, do the following:

1. Enter the Setup Utility by interrupting the boot process during the `Search for Host System...` phase (use the `Esc` key).
2. Choose `F5 = Set Configuration Parameters`.
3. Set the `Configuration File` to `debug.nsm`.
4. Press `Enter` to return to the Setup Utility. Then press `Enter` again to start the boot process.

The system will boot to the Login screen. Login as usual.

To start the Local File Manager on the Network Station:

1. Use `<Shift>-<Alt>-<Home>` to bring up the Console window.

2. From the Console, select *Terminals => New Terminal* to start a Terminal Host Chooser.
3. Select *File* from the Terminal Host Chooser.
4. Click on *OK* or press Return. The following output appears:

```
Connecting to the host "filed_telnet".......success.
*** NCD X Terminal Local File Manager ***
>
```

Starting the Local File Manager Using Telnet

To start the Local File Manager via telnet from the server, remote file manager access permissions and a password must be enabled. The following lines can be added to one of the config files (e.g. *PRODBASE*/**configs/defaults.dft**) to enable this. Replace *password* and *0.0.0.0* below with an access password and the IP address of the server.

```
set file-manager-password = password
set file-manager-access-control-enabled = true
set file-manager-access-control-list[-1] = 0.0.0.0
```

Telnet to the Network Station using
telnet *0.0.0.0* 5996
or the corresponding telnet command on your server where *0.0.0.0* is replaced by the IP address of the Network Station.

The following output appears:

```
*** NCD X Terminal Local File Manager ***
Password:
```

Enter your password and the > prompt appears.

Formatting

To format a card from the Local File Manager:
> format

The following messages appear (answer y to the prompt to proceed):

```
WARNING: format command destroys all data on pcmcia memory card!
Estimate that it will take up to 47 minutes to format device.

Do you want to proceed with formatting? [yes | no]>
nn% complete
%FILED-I_FORMATSUCCESS, format completed successfully
```

The following command provides a display of the total formatted size of the card in bytes. It should match the size of the card.

```
> df
file system information for /local:
total bytes = nnnnnnnn
reclaimable bytes = 0
```

```
free bytes = nnnnnnnn
```

The card is now formatted and ready for use. It must now be mounted to the server using NFS so files can be copied to it.

Note: Although the message indicates that it may take a long time to do a **format**, in practice it takes less than one minute.

Using NFS to Access the Local File System

NFS is used to mount/access an Network Station local file system from a server. This is necessary so that files can be copied from the server to the flash memory card. This client NFS code is provided as a part of AIX. For NT, you will have to get a 3rd party package. Several are available for download on the Internet.

More specific information on accessing the flash card from the various server platforms can be found in Appendix D.

Add the lines

```
set file-enable-nfs-server = true
set file-nfs-access-control-default = read-write
```

to the appropriate config file (suggest you use **defaults.dft**) and reboot the Network Station or use the Console screen to do
Setup => File Service => Enable NFS Server

The commands will vary depending on the server platform. On UNIX systems, this would be
`mount` *Network Station_IP_address*`:/local /mnt`

or any other suitable mount point on the server.

Standard commands or graphical interfaces on the server can now be used to access the Network Station local file system.

Copying Files to the Flash Card

Once the **/local** file system has been mounted, use commands on the server to copy files and directories to the flash card just as for any other mounted file system. If the files have been staged to the **nsflash** directory, then the UNIX commands to copy everything to the flash card is:
```
cd nsflash
cp -r * /mnt
```

3. Flash Booting the Network Station

Introduction

Once the flash card has been created, it can be used to boot one or more Network Stations. This section describes how to set up an Network Station to boot from its local PCMCIA flash card or to peer boot from a card in another Network Station.

Local Boot

Power off the Network Station and insert the flash memory card. Then clear NVRAM and set up the network parameters as explained in the following sections.

Clear NVRAM

The first step is to clear the NVRAM to ensure that there is no residual information.

1. Enter the Setup Utility by interrupting the boot process during the `Search for Host System...` phase (use the `Esc` key).
2. Use `<LeftCtrl>-<LeftAlt>-<LeftShift>-F1` to enter the Boot Monitor. The NS responds with .
3. `nv` (you should get a `->>` prompt)
4. `l`
5. `s`
6. `y`
7. `q`
8. `se`

You will now be back at the Setup Utility screen.

Using NVRAM

This section deals with setting up the parameters in NVRAM to boot/configure the Network Station. If you want to use DHCP instead, please skip to the next section.

From the Setup Utility, complete the following steps. If you are unsure about a particular entry, more information can be found in Chapter 10 of the *IBM Network Station Manager Installation and Use*.

1. Choose `F2 = View Hardware Configuration` to verify that the flash card has been successfully configured.

2. Depress `Enter` to return to the Setup Utility screen.

3. Choose `F3 = Set Network Parameters`.
   a. Set the `IP Addressed from` to `NVRAM`.
   b. Set the `Network Station IP Address` to reflect the IP address of the Network Station. Leave the entries under `Boot Host IP Address` as 0.0.0.0. Set the `First Host` value under the `Configuration Host IP Address` section to the IP address of the NSM server.
   c. Set the `Gateway IP Address`, `Subnet Mask`, and `Broadcast IP Address` parameters as appropriate for your network.

d. Depress `Enter` to return to the Setup Utility screen.

4. Choose `F4 = Set Boot Parameters`.
   a. Leave the `Boot File` value blank.
   b. Leave the `TFTP Boot Directory` and `NFS Boot Directory` entries blank.
   c. Set the `TFTP Order` and `NFS Order` under `Boot Host Protocol` to `D`.
   d. Set `LOCAL Order` to `1`.
   e. Depress `Enter` to return to the Setup Utility screen.

5. Choose `F5 = Set Configuration Parameters`.
   a. Leave the `Configuration File` value blank.
   b. Set the `First` entry under `Configuration Directory` to *PRODDIR*`/flash/`.
      See the earlier <u>table</u> for the values for *PRODDIR* on each platform. For example, in WindowsNT 4, enter `/netstation/prodbase/flash/`
      The trailing / is required at the end of the Configuration Directory path.
   c. Set the `First` entry under `Configuration Host Protocol` depending on your server:
      - AIX: `NFS`
      - NT: `NFS`
      - AS/400: `RFS/400`
   d. Press `Enter` to return to the Setup Utility.

6. Choose `F7 = Set Language Parameters`.
   a. Choose `F2 = Select Keyboard Language`.
   b. Set the appropriate language for your system using the cursor keys.
   c. Press `Enter` to return to the Select Keyboard Language screen.
   d. Press `Enter` to confirm your keyboard language (if correct). You will return to the Set Language Parameters screen.
   e. Press `Enter` to return to the Setup Utility.

7. Press `Enter` again to start the boot process.

The Network Station should boot and display the message `Booting from local device` and then load the kernel. The IBM Server Login (ACTLogin) screen will appear shortly.

Using DHCP

This section discusses the set up required to use DHCP to boot the Network Station.

Using DHCP requires that you define several new DHCP options, define scopes/subnets, and add reservation(s) for the Network Stations containing the flash cards. A Reservation assigns a fixed IP address to a server system (the system(s) with flash cards) so that other machines know where to boot from. Its values override those set for all of the machines in the scope. If you are not going to peer boot, you do not need a Reservation.

DHCP Options

The following table describes the options that need to be defined/set for DHCP for the Network Stations which have flash cards installed. These are scope options if you are not using peer boot or reservation options if you are doing peer booting.

DHCP Option Number	Option Name	Value
1	Subnet Mask	Subnet mask value (e.g. 255.255.255.0)
3	Router IP	IP Address of Router/Gateway (e.g. 90.0.0.1)
66	TFTP Server (Boot Server)	`0.0.0.0`
67	Boot File Name	`kernel` or `kernel.Z` or `kernel.63a` or `kernel.63Z`
211	Base Code Server Protocol	`local`
212	Terminal Configuration Server	IP address of NSM server (e.g. 9.53.115.220)
213	Terminal Configuration Path	*PRODDIR*`/flash/`
214	Terminal Configuration Protocol	`nfs` for AIX and NT `rfs/400` for AS/400

Using DHCP

The following steps are for verification only and it is not mandatory that they be done. If you have a new Network Station or NVRAM has just been cleared, the system should already be set up to use DHCP.

1. Choose `F2 = View Hardware Configuration` to verify that the flash card has been successfully configured.

2. Depress `Enter` to return to the Setup Utility screen.

3. Choose `F3 = Set Network Parameters`.
   a. Make sure that the `IP Addressed from` is set to `Network`.

4. Choose `F7 = Set Language Parameters`.
   a. Choose `F2 = Select Keyboard Language`.
   b. Set the appropriate language for your system using the cursor keys.
   c. Press `Enter` to return to the Select Keyboard Language screen.
   d. Press `Enter` to confirm your keyboard language (if correct). You will return to the Set Language Parameters screen.

e. Press `Enter` to return to the Setup Utility.

5. Press `Enter` again to start the boot process.

The Network Station should boot and display the message `Booting from local device` and then load the kernel. The IBM Server Login (ACTLogin) screen will appear shortly.

Peer Boot

To boot a Network Station from another Network Station's flash card, the following configuration information is required:

- The IP address of the client Network Station
- The IP address of the host/server Network Station (the one with the flash card installed)
- The subnet mask for the network (if any)
- The IP address of the gateway (if any)

First configure the Network Station containing the flash card (the "server" system) as per the "Local Boot" section above. The following edits to *PRODBASE*/**flash/standard.nsm** must be made to enable the Network Station to start an NFS server and export the local file system.

```
set file-enable-nfs-server = true
set file-export-directory-list = {
    { "/peerboot" "/local" }
}
set file-nfs-access-control-default = read-only
```

The following additional edits to *PRODBASE*/**flash/standard.nsm** allow the flash card to be accessed and maintained from each server listed in the table. Replace the *0.0.0.0* below with the IP address of the server.

```
set file-nfs-access-control-list = {
    { "/local" 0.0.0.0 read-write }
}
```

Using NVRAM

On each client Network Station, complete the following steps:

1. Clear NVRAM, as explained earlier.

2. From the Setup Utility screen, choose `F3 = Set Network Parameters`.
   a. Set the `IP Addressed from` to `NVRAM`.
   b. Set the `Network Station IP Address` to reflect the IP address of the Network Station. Set the `First Host` entry under `Boot Host IP Address` to the IP address of the Network Station containing the flash card. Set the `First Host` value under the `Configuration Host IP Address` section to the IP address of the server.
   c. Set the `Gateway IP Address`, `Subnet Mask`, and `Broadcast IP Address` parameters as appropriate for your network.
   d. Depress `Enter` to return to the Setup Utility screen.

3. Choose `F4 = Set Boot Parameters`
    a. Leave the `Boot File` value blank.
    b. Set the `NFS Boot Directory` to `/peerboot/`
    c. Set the `TFTP Order` and `LOCAL Order` under `Boot Host Protocol` to `D`.
    d. Set `NFS Order` to `1`.
    e. Depress `Enter` to return to the main Setup Utility screen

4. Choose `F5 = Set Configuration Parameters`
    a. Leave the `Configuration File` blank.
    b. Set the `First` entry under `Configuration Directory` to *PRODDIR*`/peer//`.
       See the earlier <u>table</u> for the values for *PRODDIR* on each platform. For example, in WindowsNT 4, enter `/netstation/prodbase/peer/`
       The trailing `/` is required at the end of the Configuration Directory path.
    c. Set the `First` entry under `Configuration Host Protocol` depending on your server:
        ▪ AIX: `NFS`
        ▪ NT: `NFS`
        ▪ AS/400: `RFS/400`
    d. Press `Enter` to return to the Setup Utility.

5. Choose `F7 = Set Language Parameters`.
    a. Choose `F2 = Select Keyboard Language`.
    b. Set the appropriate language for your system using the cursor keys.
    c. Press `Enter` to return to the Select Keyboard Language screen.
    d. Press `Enter` to confirm your keyboard language (if correct). You will return to the Set Language Parameters screen.
    e. Press `Enter` to return to the Setup Utility.

6. Press `Enter` again to start the boot process.

The Network Station should boot and display a message that it is booting the kernel from the IP address of the flash-enabled Network Station.

Using DHCP

DHCP Scope Options

The following table describes the options that need to be defined/set for DHCP for the Network Stations which have are to be peer booted. These are scope options.

DHCP Option Number	Option Name	Value
1	Subnet Mask	Subnet mask value (e.g. 255.255.255.0)
3	Router IP	IP Address of Router/Gateway (e.g. 90.0.0.1)

66	TFTP Server (Boot Server)	IP address of Network Station w/flash card (e.g. 90.0.0.2)
67	Boot File Name	`/peerboot/kernel`
211	Base Code Server Protocol	`nfs`
212	Terminal Configuration Server	IP address of NSM server (e.g. 9.53.115.220)
213	Terminal Configuration Path	*PRODDIR*`/peer/`
214	Terminal Configuration Protocol	`nfs` for AIX and NT `rfs/400` for AS/400

Troubleshooting

The first thing to ensure is that the flash card is recognized by the Network Station. This can be done by depressing

```
F2 = View Hardware Configuration
```
from the Setup Utility to ensure that a valid card has been recognized. After that, check all of the NVRAM setup information. Trailing slashes ("/") must be on the `Configuration Directory` and `NFS Boot Directory` entries. It also helps to clear NVRAM and re-enter the settings if you have made any changes to the network or the server configuration.

By default, the Network Station uses an NFS block size of 8192 when booting. If you are seeing a message that the kernel cannot be loaded or are having other problems loading files, especially when peer booting, try the following. It may be necessary to clear NVRAM and start over if you experience a problem where the peer system gets the kernel loaded correctly but subsequent file reads are unsuccessful. Try clearing NVRAM on both systems, reset the block size to 1024 as per the directions below, and re-enter the NVRAM values.

1. Edit the file **resources.nsl** on the server and change the following entries from 8192 to 1024:

    ```
    Login.RFS_buffer_size: 8192
    Login.NFS_buffer_size: 8192
    Login.OTHER_buffer_size: 8192
    ```

    If you are on Windows 95/98/NT, use WordPad **not** NotePad to edit this file. NotePad adds extra control characters that will cause the file to be ignored by the Network Station.

2. Power on or reboot the Network Station.

3. Enter the Setup Utility by interrupting the boot process during the `Search for Host`

`System...` phase (use the `Esc` key).

4. Use `<LeftCtrl>-<LeftAlt>-<LeftShift>-F1` to enter the Boot Monitor. The Boot Monitor responds with
   ```
   IBM Network Station model xxxx-xxx 8-bit Color Boot Monitor
   >
   ```

5. `> nf 1024`
   The Boot Monitor responds with
   ```
   Current block size is 1024 bytes
   ```

6. `> rs`

The system will now reboot.

The best tools for troubleshooting are the Console diagnostics. The Console is a Network Station application that provides access to other local clients and to diagnostic messages. The default key combination to display/hide the Console window is `<LeftShift>-<LeftAlt>-<Home>`.

The basic diagnostics messages can be enabled by pressing
`F10 = Set Verbose Diagnostic Messages Enabled` on the Setup Utility screen. This will cause the system to output file access information on the screen that is also captured in the Console log. Much more verbose diagnostics can be enabled by specifying

```
set file-extended-diagnostics = true
```

in *PRODBASE*dlash/standard.nsm.

The Console log can be viewed at the Network Station by bringing up the Console and then clicking on the `Messages` button in the Console window. The Console log can also be viewed and saved remotely using **telnet**. The diagnostics port number is 5998 and a telnet session to this port will yield a log of the Console messages. For example, under AIX the command

```
telnet  5998 | tee /tmp/console.out
```

will cause the Console messages to be displayed on the screen and also captured in file **/tmp/console.out**. A complete discussion of access to the Diagnostics and Configuration features of the Network Station can be found in Appendix E.

Performance Tips

If you are experiencing slow boot times, especially after logging in, ensure that all of the Network Station IP addresses are in the **hosts.nsm** file. This will greatly speed up the boot process. The best way to refresh **hosts.nsm** is to put all of the Network Station names and IP addresses in the server's DNS file (e.g. **/etc/hosts** on AIX) if they are not already there. Then start NSM and select `Hardware => Workstations => System defaults => Next`. In the "Domain Name Server" section of the panel, check the box "Update Network Station Manager DNS File". Then click on the "Finish" button.

Appendix A: Example Scenarios

Overview

This appendix will examine several scenarios for flash boot. The files required as well as the configuration information that must be supplied will be presented. Each deployment will probably vary somewhat, but this section will serve to illustrate how to go about some of the common setups.

The setup described here is recommended for the majority of flash boot situations. Its advantages are that only the large files (kernel, loadable modules, and Java class libraries) are stored on the flash card. All of the other information, especially the volatile configuration files, is kept on the server. This is separation of servers in action.

The procedure for using flash with ACTLogin is very straightforward. First the system and user preferences are configured on the server system using the Network Station Manager tool. Then the kernel and executable modules are loaded on the flash card. The configuration information is read from the server and augmented by flash-specific overrides.

By default, ACTLogin attempts to authenticate to the boot server. This obviously will not work correctly when booting from flash since there is no default boot server. A specific authentication server must therefore be provided to ACTLogin, as shown earlier in the description of the *PRODBASE*/**flash/standard.nsm** file.

In some cases, it is desirable to not have the ACTLogin screen displayed. This would be the case if, for example, the user just wanted to have an emulator session and didn't want/need to login explicitly to the Network Station to establish personal preferences. Information on configuring ACTLogin in kiosk (read-only) mode can be found in *Full-Screen Solutions*.

Series-Specific Files

The Network Station Series 1000 systems contain a different CPU chip (PowerPC 603) than the Series 100/300 systems (PowerPC 403). There are also different electronics and optimizations done for the two different plaftorms.

The Network Station Manager is shipped with two kernels, **kernel** and **kernel.63a** optimized for the 100/300 and 1000 respectively. There are also modules in the **mods** directory that are specifically designed for use on the Series 1000. These files can be indentified by a **.63** suffix. They are required whenever a Series 1000 system is to be booted whether directly from a local flash card or as a peer boot from another Network Station.

To save space on the flash card, do not include any **.63** suffixed files unless you are using a Series 1000 Network Station.

Compressed Kernels

In order to save time when booting across a network, the Network Station kernel (base operating

system) is supplied in both a compressed and uncompressed format. The compressed kernel is named **kernel.Z** for the Series 100/300 and **kernel.63Z** for the Series 1000. The uncompressed kernels are about 4 Mbytes in size while the compressed kernels are roughly half as large.

Either can be used when booting from flash. The uncompressed kernels may load faster from the flash card (since they don't have to be uncompressed after loading) but take up more space. If you are doing peer booting, the compressed kernel(s) should be used to reduce network traffic. Be sure to place the appropriate kernel(s) on the card for the systems you will be booting.

Scenarios

All of the sample scenarios shown in this section rely on using ACTLogin to authenticate to a server. Thus the flash card contains only the files necessary to boot the system and run applications. Configuration files and system/group/user preference information are created and managed through the NSM tool on the authentication server and remain on that system. ACTLogin processes these files, many of which are encoded, to create the environment for applications. The applications themselves are loaded from the flash card to save time and network bandwidth.

While using ACTLogin is the only supported method for booting Network Stations, there are occasions where customer preference or infrastructure may dictate otherwise. Flash boot without ACTLogin is much more difficult to configure. Appendix F provides several examples for common applications.

The files/setup required for these scenarios were taken from NSM Release 3.0.1. Later releases may require additional files, fonts, or configuration information. Testing is key to ensuring a successful flash implementation.

3270/5250/VTxxx Emulators

In this scenario, the Network Station is to be used as a 3270, 5250, or VTxxx terminal. Note that this is a setup using US English as the language. For other languages/locales, different files would be required under **X11/locales** and **keyboards** directories.

All of this code will fit on a 10MB flash card, including both compressed kernels.

Files Required

The following files and directories should be on the flash card for this scenario.

```
XKeysymDB
boot.nsl
kernel.63Z
kernel.Z
rgb.txt
SysDef
  ibmwall.xbm
X11
  app-defaults
    Mcuis        Mwm           XApplication system.mwmrc
  locale
    locale.alias locale.dir
```

```
    UTF-8_C
      XLC_LOCALE
    UTF-8_iso8859-1
      XLC_LOCALE
keyboards
  AB83useng AB83usiso BFBFuseng
mods
  actlogin.nws   keymap52.nws   lpd.nws       ns3270.nws     setup.nws
  colormap.nws   keypad.nws     lprd.nws      ns5250.nws     term.nws
  export.63a     libconf.nws    mcuis.nws     ns5250xx.nws   miscpref.nws
  export.nws     libmlc.nws     miscpr32.nws  nsterm.nws
  filed.nws      libprapi.nws   mwm.nws       sbcs_im.nws
  helpview.nws   libprxapi.nws  nfsd.nws      seriald.nws
nls
  C¹
    msg
      X.cat           Xm.cat          Xt.cat
  EN_US
    MRI
      HelpLogin       N3_5KeyH        NetHelpLogin
      Login           N3_5KeyM        NetWarningLogin
      N3_5ClrH        N3_5KyPH
      N3_5ColM        N3_5LocP
      N3_5HlpV        N3_5Resc
    msg
      X.cat           Xt.cat          libprxapi.cat mwm.cat
      Xm.cat          common.cat      mcuis.cat     term.cat
```

Notes:

[1] This directory does not exist on Windows systems. Copy the corresponding **msg** directory and files from the **EN_US** directory into the **C** directory when staging the data. If you do not have the **C** directory, boot times may be longer.

3270/5250/VTxxx Emulators w/Fonts

In the emulator scenario above, all font information is read from the server. This scenario can be enhanced by adding font files to the flash memory card. This reduces the network bandwidth and Network Station boot time since fewer files need to be transferred across the network. Note that it somewhat trail and error to determine which fonts actually get used by each application. There are over 25 Mbytes of font files, and these also vary by locale, so it is generally best to read these from the server.

The following lines would need to be added to the config files to instruct the system to read fonts from the card.

The PRODBASE/flash/standard.nsm File

Add these lines to the end of *PRODBASE*/**flash/standard.nsm**:

```
#
# Get the fonts from flash
set xserver-default-font-path = {
    {"/local/X11/fonts/pcf/i18n"}
    {"built-ins"}
}
```

The PRODBASE/peer/standard.nsm File

Add these lines to the end of *PRODBASE*/**peer/standard.nsm**:

```
#
# Get the fonts from flash
set xserver-default-font-path = {
    {"/peerboot/X11/fonts/pcf/i18n"}
    {"built-ins"}
}
```

Files Required

A number of additional files are required for this scenario. Here is a list of what was used added in addition to the files in the emulator scenario above.

```
X11
  fonts
    pcf
      i18n
        Block11.iso1_UCS.pcf.Z Rom14.base_UCS.pcf.Z   Rom8.iso1_UCS.pcf.Z
        Block17.iso1_UCS.pcf.Z Rom14.iso1_UCS.pcf.Z   fonts.alias
        Ergo15.iso1_UCS.pcf.Z  Rom22.base_UCS.pcf.Z   fonts.dir
        Ergo17.iso1_UCS.pcf.Z  Rom22.iso1_UCS.pcf.Z
```

NC Navigator with JVM

In this scenario, the Network Station is booted and the NC Navigator browser is available along with the Java Virtual Machine to run applets. There is no change to the configuration files since all of the configuration for NC Navigator would be done on the server using NSM. The only difference is in what files need to be included on the flash card. This setup requires 20 Mbytes on the card if the compressed kernels for all systems are to be used. If the compressed kernel and mods files for only the Series 100/300 or Series 1000 is required, the size drops to less than 16 Mbytes.

More information on configuring NC Navigator for the Network Station can be found in the NC Navigator 3 for Release 3 Workbook (http://w3.rchland.ibm.com/~jepe/navio.html). **Disk caching should not be used when running NC Navigator with flash. The system is shipped with disk caching disabled, and this setting should not be changed.**

Note that this is a setup using US English as the language. For other languages/locales, different files would be required under **X11/locales** and **keyboards** directories.

Files Required

```
XKeysymDB
boot.nsl
kernel.63Z
kernel.Z
rgb.txt
SysDef
  ibmwall.xbm
X11
```

```
EN_US
  app-defaults
    Navio
  app-defaults
    Mcuis          Mwm            XApplication system.mwmrc
  locale
    locale.alias locale.dir
    UTF-8_C
      XLC_LOCALE
    UTF-8_iso8859-1
      XLC_LOCALE
java
  classes.zip  javacpa0.gif javacpat.gif nwshacl.zip  nwspackg.zip
  lib
    appletviewer.properties  font.properties          rmic.properties
    awt.properties           font.properties.en        serialver.properties
    content-types.properties javac.properties
    security
      java.security
keyboards
  AB83useng
mods
  actlogin.nws        jcomm.nws          jsysresource.nws mwm.nws
  desktop.nws         jjpeg.63a          jzip.63a          navio.nws
  export.63a          jjpeg.nws          jzip.nws          nfsd.nws
  export.nws          jmath.63a          libconf.nws       sbcs_im.nws
  filed.nws           jmath.nws          libmlc.nws        seriald.nws
  java.63a            jmmedia.63a        libprapi.nws      setup.nws
  java.nws            jmmedia.nws        libprxapi.nws
  jawt.63a            jnet.63a           loadb.nws
  jawt.nws            jnet.nws           mcuis.nws
  NAV
    navio.zip
nls
  C$^1$
    msg
      X.cat          Xm.cat          Xt.cat
  EN_US
    MRI
      CAddress.pt.i.xpm    MMsgNew.pt.xpm        THome.pt.xpm
      CAddress.pt.xpm      MMsgPst.pt.i.xpm      TLoadImages.pt.i.xpm
      CAttachM.pt.i.xpm    MMsgPst.pt.xpm        TLoadImages.pt.xpm
      CAttachM.pt.xpm      MMsgPstF.pt.i.xpm     TNext.pt.i.xpm
      CQuote.pt.i.xpm      MMsgPstF.pt.xpm       TNext.pt.xpm
      CQuote.pt.xpm        MMsgPstR.pt.i.xpm     TOpenUrl.pt.xpm
      CSendLater.pt.i.xpm  MMsgPstR.pt.xpm       TPrint.pt.i.xpm
      CSendLater.pt.xpm    MMsgRep.pt.xpm        TPrint.pt.xpm
      CSendM.pt.i.xpm      MMsgRep.pt.xpm        TReload.pt.xpm
      CSendM.pt.xpm        MMsgRepA.pt.i.xpm     TStop.pt.i.xpm
      HelpLogin            MMsgRepA.pt.xpm       TStop.pt.xpm
      Login                MNextU.pt.i.xpm       about.htm
      MDel.pt.i.xpm        MNextU.pt.xpm         fontedit.hlp
      MDel.pt.xpm          MPrevU.pt.i.xpm       header.gif
      MGetM.pt.xpm         MPrevU.pt.xpm         help.htm
      MMarkA.pt.i.xpm      MPrint.xpm            ibmlogo.gif
      MMarkA.pt.xpm        MStop.xpm             index.htm
      MMarkT.pt.i.xpm      NetHelpLogin          links.htm
      MMarkT.pt.xpm        NetWarningLogin       splash.htm
      MMsgFwd.pt.i.xpm     TBack.pt.i.xpm        userdict.hlp
      MMsgFwd.pt.xpm       TBack.pt.xpm
      MMsgNew.pt.i.xpm     TFind.pt.xpm
    msg
      X.cat          Xt.cat          libprxapi.cat mwm.cat
```

```
    Xm.cat          common.cat      mcuis.cat
```

Notes:

[1] This directory does not exist on Windows systems. Copy the corresponding **msg** directory and files from the **EN_US** directory into the **C** directory when staging the data. If you do not have the **C** directory, boot times may be longer.

ICA Client

In this scenario, the Network Station is booted and the ICA client is available to connect to a WinCenter or MetaFrame server. There is no change to the configuration files. This setup requires less than 8 Mbytes on the card if the compressed kernels for all systems are to be used.

Note that this is a setup using US English as the language. For other languages/locales, different files would be required under **X11/locales** and **keyboards** directories.

Files Required

```
XKeysymDB
boot.nsl
kernel.Z
kernel.63Z
rgb.txt
SysDef
  ibmwall.xbm
X11
  app-defaults
    Mcuis         Mwm             XApplication system.mwmrc
  locale
    locale.alias locale.dir
    UTF-8_C
      XLC_LOCALE
    UTF-8_iso8859-1
      XLC_LOCALE
keyboards
  AB83useng BFBFuseng
mods
  actlogin.nws   icaclnt.nws   libprapi.nws    mcuis.nws
  export.63a     icaui.nws     libprxapi.nws   mwm.nws
  export.nws     libconf.nws   lpd.nws         nfsd.nws
  filed.nws      libmlc.nws    lprd.nws        sbcs_im.nws
                                                serial.nws
nls
  C¹
    msg
      X.cat         Xm.cat        Xt.cat
  EN_US
    MRI
      HelpLogin     NetHelpLogin
      Login         NetWarningLogin
    msg
      X.cat         Xt.cat        libprxapi.cat mwm.cat
      Xm.cat        common.cat    mcuis.cat
```

Notes:

[1] This directory does not exist on Windows systems. Copy the corresponding **msg** directory and files

from the **EN_US** directory into the **C** directory when staging the data. If you do not have the **C** directory, boot times may be longer.

Java Application

In this scenario, the Network Station is booted and a Java application is started. Using the compressed kernels requires about 16 Mbytes plus the size of the Java application.

Note that this is a setup using US English as the language. For other languages/locales, different files would be required under **X11/locales** and **keyboards** directories.

Files Required

```
XKeysymDB
boot.nsl
kernel.Z
kernel.63Z
rgb.txt
SysDef
  ibmwall.xbm
X11
  app-defaults
    Mcuis        Mwm            XApplication system.mwmrc
  locale
    locale.alias locale.dir
    UTF-8_C
      XLC_LOCALE
    UTF-8_iso8859-1
      XLC_LOCALE
java
  classes.zip  javacpa0.gif javacpat.gif nwshacl.zip  nwspackg.zip
  lib
    appletviewer.properties  font.properties        rmic.properties
    awt.properties           font.properties.en     serialver.properties
    content-types.properties javac.properties
    security
      java.security
keyboards
  AB83useng BFBFuseng
mods
  actlogin.nws    jcomm.nws       jnet.63a        libprdbcs.nws
  export.63a      jjitc.63a       jnet.nws        libprxapi.nws
  export.nws      jjpeg.63a       jsysresource.nws mcuis.nws
  filed.nws       jjpeg.nws       jzip.63a        mwm.nws
  java.63a        jmath.63a       jzip.nws        nfsd.nws
  java.nws        jmath.nws       libconf.nws     sbcs_im.nws
  jawt.63a        jmmedia.63a     libmlc.nws      seriald.nws
  jawt.nws        jmmedia.nws     libprapi.nws    setup.nws
nls
  C¹
    msg
      X.cat        Xm.cat         Xt.cat
  EN_US
    MRI
      HelpLogin      NetHelpLogin
      Login          NetWarningLogin
    msg
      X.cat        Xt.cat         libprxapi.cat mwm.cat
      Xm.cat       common.cat     mcuis.cat
```

Notes:
[1] This directory does not exist on Windows systems. Copy the corresponding **msg** directory and files from the **EN_US** directory into the **C** directory when staging the data. If you do not have the **C** directory, boot times may be longer.

Tips and Tricks

Backup Servers

A Network Station can have up to three defined boot servers and two configuration/authentication servers. This capability can be used to provide backup capabilities to peer-booted Network Stations.

Redundant Peer Boot

For example, if there are two Network Stations with flash cards in a location (say NS1 and NS2), half of the peer Network Stations could be set with the First Boot Host set to NS1 and their Second Boot Host set to NS2. The other half could be set to have NS2 as their First Boot Host and NS1 as their Second Boot Host. In normal operation, each flash card would boot its own Network Station and half of the peer Network Stations. If either NS1 or NS2 were broken or powered off, all of the Network Stations would automatically revert to booting from the remaining flash-enabled Network Station.

Redundancy Using A Server

Flash/peer boot is typically used in sites that do not have a local boot server. Most of these have low-speed WAN connections back to the configuration/authentication servers at a remote location. Since these servers typically contain all of the NSM code, they can be used as (slow) boot servers in the event that the flash-enabled Newtork Station is unavailable. By defining an NFS alias on the server that equates the *PRODBASE* directory to */peerboot/*. In AIX, this can be done by creating a symbolic link from /peerboot to /usr/netstation. In NT 4.0, an NFS alias can be defined between /nstation/Prodbase and /peerboot/.

Printing Solution

The customer wanted to be able to use NSM to configure printers on a per terminal basis so that a long and confusing list of printers would not be given to all users and default printers could be chosen based on the actual location of the station.

The problem was, the terminal (workstation) level configuration data is saved in the IP_ADDRESS (e.g. 192.68.7.3) file in the *PRODBASE*/configs directory and this file is only read when the NVRAM configuration file section on the netstation is blank. We were able to work around this limitation by editing the IP_ADDRESS file that is created by NSM and replacing the first line
```
read standard.nsm
```
with
```
read /netstation/prodbase/flash/standard.nsm
read /netstation/prodbase/peer/standard.nsm
```

When all files were read it returned to this file and then read the workstation level configuration file

listed on lines 2 and 3 of the IP_ADDRESS file.

Note: This particular customer is using flash, peer, and normal booting in their environment and with this solution they could get all three to use terminal (workstation) level configuration by blanking out the configuration file line in NVRAM and then for flash or peer making the change above, and for normal boot no other change is required.

Appendix B: Local File Manager Commands

Local File Manager Command Summary

Command	Description
cd *directory*	Changes the current working directory to *directory*
compare *file1 file2* or **cmp** *file1 file2*	Compare the contents of the two files and displays a message stating whether the files are equivalent.
copy *sourcefile destfile* or **cp** *sourcefile destfile*	Copies the specified file (*sourcefile*) to the specified destination (*destfile*). Copying files may take a long time and affect the kernel's response time while it is taking place. Before copying files to the local file system from a remote file system: • On a PCMCIA card, if you have deleted files from the local file system recently, use the **reclaim** command to ensure that all available space is accessible. • Make sure that the terminal's file service table includes an entry for the remote location. Before copying files from the local file system to a remote file system: • If you are copying files from the local file system to a remote location, an empty file with the desired name must exist already on the remote file system when using TFTP. • Write access must be enabled for the file on the remote file system. • The Network Station's file service table must include an entry for the remote location.
cwd or **pwd**	Displays the current working directory
delete *file(s)* or **del** *file(s)* or **remove** *file(s)* or **rm** *file(s)*	Delete the specified *file(s)* from the local file system. On a PCMCIA card, after using the **delete** command, use the **reclaim** command to re-pack previously used file space for subsequent use.

format [/local]	Formats the local file system **/local**. **Warning:** Formatting the file system destroys any data that is already on it. Formatting a flash card requires about two minutes per megabyte.
help	Displays a list of Local File Manager commands.
info or **df [/local]**	Lists the total size in bytes of the local filesystem and the total number of free bytes available.
list or **ls** or **dir [-R]** [*directory*]	Displays a list of the files stored in the local file system. This command can be used with a **-R** option to list subdirectory content recursively.
mkdir *dir_name*	Makes a directory named *dir_name* in the local file system.
quit	Disconnects from the Local File Manager
reclaim	Reclaims previously used file system space. This command may take several minutes.
verify	Confirms that the local file system structure is valid.

Configuring Local File Manager Characteristics

You can configure the Local File Manager through configuration or the Setup menu.

Local File Manager Configuration

Configurable Characteristic/ (Default Value)	**Remote Configuration Parameter**	**Change Setup Parameters =>**
Specifying the hosts permitted access to the Local File Manager; requires values for both parameters (**disabled; none**)	**file-manager-access-control-enabled**	Access Control (File Manager section) => Enable File Manager Access Control
	file-manager-access-control-list	Access Control (File Manager section) => Enable File Manager Access Control List
Specifies the ports used for Local File Manager access (TCP: **5976**; Telnet: **5996**)	**file-manager-tcp-port**	File Manager => TCP Port Number
	file-manager-telnet-port	File Manager => Telnet Port Number

Appendix C: Module Information

The following table describes the functions of the various files in the **mods** directory. However, it is not possible to completely describe when each module is required. Trial and error testing is required to ensure that a complete set of modules is placed on the flash card prior to deployment. (This section is undergoing modification to bring it up to Release 3)

Module Name	Type	Description
actlogin.nws	client	Login Authentication Manager
audio.nws	extension	Network Audio
colormap.nws	extension	3270/5250 color map
desktop.nws	extension	Used by NC Navigator
export.nws export63a.nws	library	Symbols exported by the kernel
filed.nws	daemon	Local File Manager daemon
helpview.nws	extension	Help Viewer
jagent.nws	extension	Java ???
java.nws	extension	Java Virtual Machine
jawt.nws	extension	Java Abstract Window Toolkit
jjpeg.nws	extension	Java JPEG
jmath.nws	extension	Java math
jmmedia.nws	extension	Java multi-media
jnet.nws	extension	Java networking
jsysresource.nws	extension	???
jzip.nws	extension	uncompress compressed Java zip files
keymap52.nws	extension	5250 keymap editor local client
libconf.nws	library	Configuration library

libmlc.nws	library	Used by Console => Setup
libppp.nws	library	PPP protocol library
libprapi.nws	library	AIX printer library
libprxapi.nws	library	X Windows library
loadb.nws	utility	module loader (NC Navigator)
login.nws	utility	login local client
mcuis.nws	utility	Provides a pop-up error console
miscpref.nws	utility	5250 Miscellaneous Preferences
mwm.nws	client	Motif window manager
nav128.nws	client	NC Navigator 128-bit (US only)
navio.nws	client	NC Navigator browser (40-bit)
nfsd.nws	daemon	Network File System daemon
nsterm.nws	client	VTxxx emulator
ns3270.nws	client	3270 emulator
ns5250.nws	client	5250 emulator
pref.nws	utility	Change User Preferences (from Console)
qsetup.nws	utility	Change Quick Setup (from Console)
seriald.nws	extension	Serial/Parallel port daemon
setup.nws	utility	Change Setup Parameters (from Console)
show.nws	utility	Show Memory (from Console)
sie.nws	extension	Simple Image Extension (not used by anything)
stats.nws	utility	Show Statistics (from Console)

term.nws	client	VTxxx emulator
test.nws	utility	Test Network (from Console)
touchscr.nws	client	touchscreen support
wm.nws	client	built-in window manager
xinput.nws	client	required for touchscreen calibration

Appendix D: Platform-Specific Information

This appendix contains information on creating flash cards from specific server platforms.

AIX (and other UNIX Platforms)

This document is written from a UNIX reference point. The NFS and system commands in the examples should work as shown on all UNIX systems.

AS/400

At this time, we do not have any recommendations for using an AS/400 to copy files to a formatted flash card. It is advised that a Windows NT/95/98 or UNIX system be used for this purpose. We will update this section as more information becomes available.

Windows NT/95/98

NFS Client

Windows NT does not supply an NFS client package. The use of NFS greatly simplifies getting the staged data onto the flash card. NFS client software for Windows 95/98 and NT is available from Hummingbird Communications (http://www.hummingbird.com) through their NFS Maestro Client product, FTP Software with InterDrive, XLink Technology, and others.

In general, these packages allow you to mount the flash card as a new drive and access it through Windows Explorer.

FTP InterDrive

Care must be taken to ensure that files retain their case-sensitive names when being copied to the flash card. The InterDrive 2.0 client's configuration menus can be accessed from *Network Neighborhood => Entire Network => InterDrive NT*. This allows you to change the case mapping for files and directories.

Hummingbird NFS Maestro

Hummingbird has a 30-day evaluation copy which we have found to be very stable and useful. At installation, Maestro will attempt to determine the optimum values for NFS read/write sizes to your Network Station (or other address on the network). We suggest using this value as the maximum value for these entries in the File Service Table in *PRODBASE*/**flash/standard.nsm** and *PRODBASE*/**peer/standard.nsm**.

After installation, start the Maestro client using `Start => NFS Maestro => NFS Network Access`. In NFS Network Access, set the `Network Path` to `\\<Network_Station_IP_Address>\local`. Set the `Username` to `nobody`. Always set the "Preserve Case" checkbox.
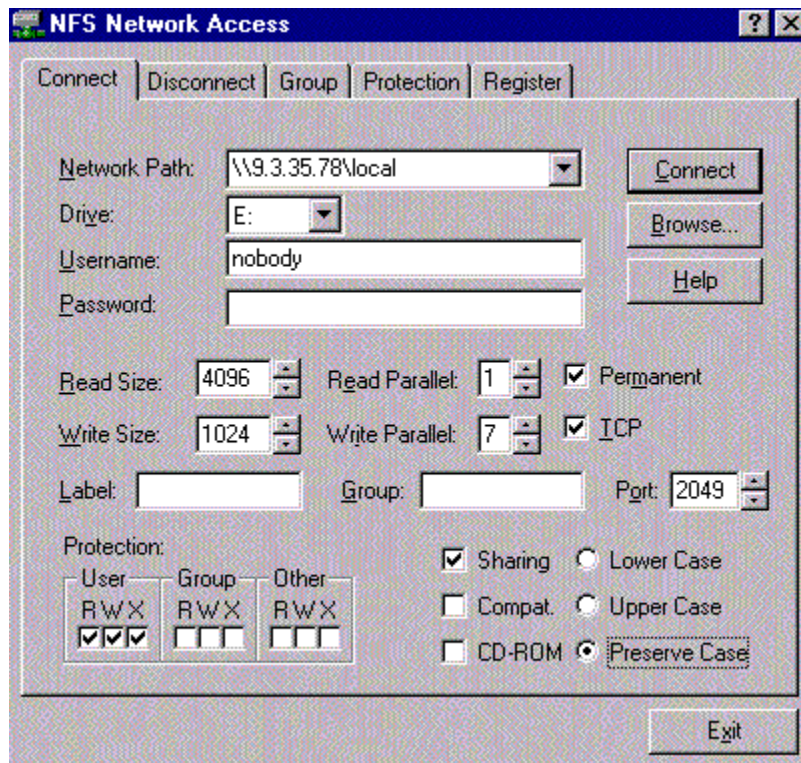
After depressing the `Connect` button, the **/local** filesystem on the flash card will be accessible from Explorer under the drive that was mapped on the `Drive` entry. You can then drag/drop files form any other Windows directory onto the flash filesystem. Be sure to disconnect when you are finished.

We have noticed instances where the Network Station panics while files are being copied. Setting the `Write Parallel` value to `1` may solve that problem. We have also found that setting
`pref-power-manage-enable = false`
in **defaults.dft** is useful.

Editing Files in Windows

The Network Station operating system and window manager are derived from UNIX. UNIX ASCII files have lines that are delimited by a new-line character. Windows editors such as Wordpad generally separate lines with a CR/LF sequence. The configuration files in *PRODBASE*/configs can be edited

using Wordpad without causing problems. However, files such as those in the **app-defaults** directories, keyboard maps, and other types of files should **not** be edited using Wordpad or other Windows editors that use a CR/LF to delineate lines.

The **Notepad** program can be used to view/edit these files. However, you must cut/paste the symbol (looks like a wide vertical bar) that delineates lines in the original file. There are some utilities that can translate the files back to UNIX format. Both NFS Maestro and InterDrive have DOS to UNIX utilities. If you must edit any of these files, it is recommended that a UNIX system be used.

Files named **pref** should never be edited on any system. These files are in unicode and will be corrupted if a non-unicode editor is used.

Mircosoft DHCP

The basic steps for using Microsoft DHCP can be found in *IBM Network Station Manager Installation and Use*. This document may be ordered in the US as publication SC41-0664 or accessed on the web at http://publib.boulder.ibm.com/pubs/html/netcom/r3.htm.

Install MS DHCP

1. From the Start => Settings => Control Panel => Network => Services => Add
2. Select "Microsoft DHCP Server"
3. Start DHCP Manager. Start => Programs => Administrative Tools => DHCP Manager

Create a Scope

Create a scope that includes the addresses that you want to use for all of the Network Stations to be booted from a single flash card.

1. Double click "Local Machine" select Scope => Create
2. Enter Start and End Address, Subnet Mask, and set `Lease Duration`.
3. Select `OK`.

Set DHCP Options for Scope

Add Reserved Clients

This will be the IP address for the Network Station with the flash card.

1. Select Scope => Add Reservations
2. IP Address: *IP address to assign to NS w/flash card*
3. Unique Identifier: *MAC address of NS*
4. Client Name: *host name of the Network Station*
5. Select `Add`.

Appendix E: How to use Diagnostic and

Interactive Configuration

Overview

This appendix will define and describe how to setup, access and use the Diagnostic and Interactive Configuration features of the IBM Network Station. For purposes of this document, only the **telnet** method of access to the Diagnostic and Interactive Configuration functions will be addressed.

This section is taken from material created by John Tesch of the Network Station Technical Support Team.

Purpose of the Diagnostic and Interactive Configuration Features

The purpose of these features is to assist in diagnosing and resolving configuration problems with the IBM Network Station. With the diagnostic feature, system administrative personnel can remotely diagnose problems by identifying error and warning messages reported by the Network Station software. Along with the diagnostic feature, the interactive configuration feature allows the resolution of these problems. All of this can occur remotely, away from the IBM Network Station. resulting in quicker resolution of Network Station configuration problems.

Definitions

Diagd

A Diagnostic session which provides network station diagnostic messages. These messages include warning, error and informational messages such as: file access, fonts, network routing, and other similar messages.

Configd

An Interactive Configuration feature which provides the ability to view and manipulate all network station parameters remotely.

The Diagnostic Feature

The Diagnostic session can be accessed without any setup or changes to the Network Station configuration. The default setup of the network station allows connection to **diagd**.

Access to the diagnostic session is provided via **telnet** into the IBM Network Station at port 5998. This can be accomplished from any host or terminal on the network.

If you are logged into a host, below is an example of this command:

```
$ telnet 138.43.215.72 5998
```

If you are on a UNIX host such as AIX, the contents of the session can be saved to a file by piping the output to the tee command:

```
$ telnet 138.43.215.72 5998 | tee /tmp/diag.log
```

This will log the session into the diagnostic session of the IBM Network Station at IP address 138.43.215.72. A name could be substituted for the IP address if the host is resolving names into IP addresses.

To exit the session, press `Ctl-t`. This will give you the **telnet** prompt, usually `tn>`. Just type `quit`.

The Interactive Configuration Feature

Accessing the interactive configuration feature is very similar to the diagnostic session. One main difference is the need for a password before entering the interactive configuration. There are two different passwords which each have different levels of capabilities. The read-only password will only allow the display and viewing of parameters. The read-write password will allow the viewing as well as modification to all passwords.

There are 4 different passwords that can be assigned to the interactive configuration feature. These passwords are **read-only and read-write for configuration parameters** and **read-only and read-write for user preferences parameters**.

Passwords can be assigned in the configuration files for the Network Station, e.g. **defaults.dft**. The configuration parameters that cover passwords are:

```
config-pref-read-only-password
config-pref-read-write-password
config-read-only-password
config-read-write-password
unit-global-password
```

Another method of assigning the passwords is to input any or all of the 4 passwords into the appropriate fields in the `Access Control` section of the `Change Setup Parameters` client which can be started from the `Setup` pulldown menu from the `User Services: Console`. After the password(s) have been entered, press the `Apply` button to activate the changes to the Network Station software.

The default read-only password is "public."

Access to the interactive configuration is accomplished via **telnet** into the terminal at port 5999. This can be accomplished from any host or terminal on the network.

If you are logged into a host, below is an example of this command:

```
$ telnet 138.43.215.72 5999
```

If you want to log the information to a file, you can run the telnet session as follows:

```
$ telnet 138.43.215.72 5999 | tee /tmp/config.log
```

The output will be stored in the file **/tmp/config.log**. This is especially useful when listing all parameters.

Either method will log this session into the interactive configuration of the IBM Network Station at IP

address 138.43.215.72. A name could be substituted for the IP address if the host is resolving names into IP addresses.

After making a telnet connection with the interactive configuration, a password will need to be input. At this point, the read-only or read-write password can be entered. A greater-than sign (>) will be the prompt once a successful password is entered.

Commands

Below is a list of most of the interactive configuration commands. For a complete listing of commands and command options, type 'help' while in the interactive configuration session.

Interactive Configuration Command Summary

Command	Description
help	print out all commands with syntax
get	display parameter or parameters with current values
get-protect	protect against get operations of specified parameters
set	set a parameter with new values
set-protect	protect against set operations of specified parameters
pending	display all modified parameters not yet applied
cancel	cancel all modified parameters not yet applied
apply	apply all modified parameters not yet applied
read nvram	read NVram values
write nvram	update nvram with all parameters with latest values
read *file*	read parameters from *file*
write *file param*	write to *file* all parameters or specific parameter *param*
lock	gain exclusive control of configuration
unlock	release exclusive control of configuration
quit	disconnect from configuration session

Viewing parameter values

Viewing all parameters

All parameters may be viewed by issuing the following command:

```
> get all
```

All parameters will be displayed with their respective values.

Viewing parameter groups

Each network station parameter belongs to a group of parameters. Listed below is a list of all parameter groups. (Note: This is a nice feature to find the syntax of a specific parameter if the group is known)

```
boot       config      diag        enet
exec       file        font        icmp
ip         lat         login       ncdnet
net        nfs         ppp         pref
serial     snmp        tcp         tcpip
term       tftp        tokring     udp
unit       xremote     xserver
```

To view a group of parameters, issue the **get** command. For example, if display of all boot parameters (group 'boot') are desired, issue the following command:

```
> get boot
```

All of the parameters in the boot group will be displayed with their respective values.

Viewing specific parameters

Viewing values for a specific network station parameter is quite simple. For example, if the value to be viewed is the primary default gateway, the following command would be issued:

```
> get ip-initial-default-gateway-1
```

The result is the following:

```
ip-initial-default-gateway-1 = 138.43.215.2
```

Setting parameter values

As stated previously, the password entered at the prompt will depend on which parameters can be changed. If the read-write configuration password was given, then changing values to any parameter is permitted. If the read-write user preferences password was entered, then only the parameters in the 'pref' group may be changed. Both read-only passwords only allow viewing of parameter values.

The examples below assume the read-write configuration password was entered when entering the interactive configuration.

Changing a parameter requires two different steps. The first step is actually changing the parameter, the second is applying the change to make the change active. For example, to turn on extended file diagnostics, the **file-extended-diagnostics** value needs to be modified.

```
> set file-extended-diagnostics = true
```

Individual table entries can be added to a table. This is done by subscripting the parameter with [-1]. For example, if an additional host needs to be added to the tcpip-name-servers parameter, the input would look as follows:

```
> set tcpip-name-servers[-1] = 138.43.214.24
```

Individual table entries can be changed in a table as well. This is done by subscripting the parameter with the table entry number. For example, the 138.43.214.28 host in the tcpip-name-servers should be 138.43.215.28 instead. Below is the syntax to change one of the TCP/IP name servers in a table:

```
> set tcpip-name-servers[1] = 138.43.215.28
```

Important: The second step in modifying a parameter is applying it. The following commands will verify that there are parameters that have been modified but not yet applied and then eventually applied.

```
> pending
file-extended-diagnostics = true
tcpip-name-servers[-1] =  { }
tcpip-name-servers[-1]server = 138.43.215.24
tcpip-name-servers[1] =  { }
tcpip-name-servers[1]server = 138.43.215.28
> apply
>
```

The listing above indicates the original 2 entries, 1 addition, and 1 change to the tcpip-name-servers parameter as we performed them in the example above.

Appendix F: Flash Boot Without ACTLogin

Overview

The combination of NSM and ACTLogin provides a powerful management tool for Network Stations. The kiosk mode of ACTLogin can be employed in situations where the user does not need to explicity authenticate prior to running applications. While the only supported method of running a Network Station is via ACTLogin, there are occasions when this is not desirable or a customer prefers not to have even a silent (kiosk) authentication. This section describes setting up a flash boot environment without ACTLogin.

ACTLogin provides the mechanism for reading the system, group, and user preferences set up by NSM. Using these files, it sets up the appropriate environment for the native Network Station applications, i.e. 5250, 3270, NC Navigator, etc. These applications do not themselves read the various preference files but rather rely on environment variable and X Windows resources that are initialized by ACTLogin. The major stumbling block to creating a flash boot scenario without ACTLogin is determining how to recreate the ACTLogin environment for the applications to be run.

Local Boot

Configuration Files

Due to some implementation issues (aka bugs), the Network Station will ignore any configuration directory other than **/local/configs**, whether specified through NVRAM or DHCP. Therefore, the flash configuration file must be put in **/local/configs/standard.nsm** and not **/local/flash/standard.nsm**.

Setting the Network Parameters

From the Setup Utility, complete the following steps. If you are unsure about a particular entry, more information can be found in Chapter 10 of the *IBM Network Station Manager Installation and Use*.

1. Clear NVRAM, as explained earlier.
2. From the Setup Utility screen, choose `F2 = View Hardware Configuration` to verify that the flash card has been successfully configured.
3. Depress `Enter` to return to the Setup Utility screen.
4. Choose `F3 = Set Network Parameters`.
5. Set the `IP Addressed from` to `NVRAM` or `Network` depending on whether you want to use DHCP to assign the Network Station's address (`Network`) or hard-code it.
6. If using NVRAM boot, set the `Network Station IP Address` to reflect the IP address of the Network Station. Leave the entries under `Boot Host IP Address` and `Configuration Host IP Address` as 0.0.0.0.
7. Set the `Gateway IP Address`, `Subnet Mask`, and `Broadcast IP Address` parameters as appropriate for your network.
8. Depress `Enter` to return to the Setup Utility screen.
9. Choose `F4 = Set Boot Parameters`.
10. Leave the `Boot File` value blank.
11. Leave the `TFTP Boot Directory` and `NFS Boot Directory` entries blank.
12. Set the `TFTP Order` and `NFS Order` under `Boot Host Protocol` to `D`.
13. Set `LOCAL Order` to `1`.
14. Depress `Enter` to return to the Setup Utility screen.
15. Choose `F5 = Set Configuration Parameters`.
16. Leave the `Configuration File` blank.
17. Set the `First` entry under `Configuration Directory` to `/local/configs/`.
18. Set the `First` entry under `Configuration Host Protocol` to `Local`.
19. Press `Enter` to return to the Setup Utility. Then press `Enter` again to start the boot process.

The Network Station should boot and display the message `Booting from local device` and then load the kernel. The IBM Server Login (ACTLogin) screen will appear shortly.

Peer Boot

To boot a Network Station from another Network Station's flash card, the following configuration information is required:

- The IP address of the client Network Station
- The IP address of the host/server Network Station (the one with the flash card installed)
- The subnet mask for the network (if any)

- The IP address of the gateway (if any)

First configure the Network Station containing the flash card (the "server" system) as per the "Local Boot" section above. The following edits to *PRODBASE*/**flash/standard.nsm** must be made to enable the Network Station to start an NFS server and export the local file system.

```
set file-enable-nfs-server = true
set file-export-directory-list = {
    { "/peerboot" "/local" }
}
set file-nfs-access-control-default = read-only
```

The following additional edits to *PRODBASE*/**flash/standard.nsm** allow the flash card to be accessed and maintained from each server listed in the table. Replace the *0.0.0.0* with the IP address of the server.

```
set file-nfs-access-control-list = {
    { "/local" 0.0.0.0 read-write }
}
```

Then, on each peer Network Station, complete the following steps:

1. Clear NVRAM, as explained earlier.
2. From the Setup Utility screen, choose `F3 = Set Network Parameters`.
3. Set the `IP Addressed from` to `NVRAM` or `Network` depending on whether you want to use DHCP to assign the Network Station's address (`Network`) or hard-code it.
4. If using NVRAM boot, set the `Network Station IP Address` to reflect the IP address of the Network Station. Set the `First Host` entry under `Boot Host IP Address` to the IP address of the Network Station containing the flash card. Leave the `Configuration Host IP Address` as 0.0.0.0.
5. Set the `Gateway IP Address`, `Subnet Mask`, and `Broadcast IP Address` parameters as appropriate for your network.
6. Depress `Enter` to return to the Setup Utility screen.
7. Choose `F4 = Set Boot Parameters`.
8. Leave the `Boot File` value blank.
9. Leave the `TFTP Boot Directory` entry blank.
10. Set the `NFS Boot Directory` to /peerboot/.
11. Set the `TFTP Order` and `LOCAL Order` under `Boot Host Protocol` to D.
12. Set `NFS Order` to 1.
13. Depress `Enter` to return to the Setup Utility screen.
14. Choose `F5 = Set Configuration Parameters`.
15. Set the `Configuration File` to peer.nsm.
16. Set the `First` entry under `Configuration Directory` to /peerboot/configs/.
17. Set the `First` entry under `Configuration Host Protocol` to NFS.
18. Press `Enter` to return to the Setup Utility. Then press `Enter` again to start the boot process.

The Network Station should boot and display a message that it is booting the kernel from the IP address of the flash-enabled Network Station.

Scenarios

Unlike the ACTLogin scenarios, each of these scenarios requires special configuration files to start and configure values for the application(s).

3270/5250/VTxxx Emulators

This scenario shows a 3270, 5250, and VT220 session being started automatically at system startup. These applications are also available via a drop-down menu accessed by doing `Alt-Shift-RightMouse`. The 3270 and 5250 applications have special keymap files. The 5250 session is started without a menubar. The VT220 session always reconnects to a host after it terminates. All of these are settings from the **pref** files that have been added to the xserver-initial-x-resources section. This configuration requires a 16 Mbyte card.

Note that the 3270 and 5250 emulators create files in the user's home directory in order to save preferences such as desktop position, colormap, background, etc. These files will be created on the flash card. While the files are relatively small, linear flash does not reuse space and over time the card can fill up. The `Options` and `Print` memus should be disabled in *PRODBASE*/**flash/standard.nsm** or a larger card should be procured if you do not want to run the risk of running out of space. The `-NO_DESKTOP_FILE` option on the command line will disable writing some of these files.

The /local/configs/standard.nsm File

The **standard.nsm** file reads all of the other configuration files and then overrides values. The easiest way to set up these non-ACTLogin scenarios is to configure the system using ACTLogin and then copy all of the files except **standard.nsm** onto the card. Create a new **standard.nsm** as below and have it read all of the configuration files previously read by **standard.nsm** on the NSM server..

Replace the *0.0.0.0* entries with the IP addresses of the various hosts. Replace other entries as indicated.

```
#
# standard.nsm for emulator scenario
#
# Add/Remove the following "read" statments to match exactly the same files
# that are read by PRODBASE/configs/standard.nsm
read required.nsm
read control.nsm
read hosts.nsm
read defaults.nsm
read defaults.dft
#
# Overrides for flash boot
set boot-desired-source = local
set boot-prom-update-file="/local/proms/bootm"
set boot-test-ram = false
set boot-token-ring-update-file="/local/proms/trflash"
set boot-second-source = none
set boot-third-source = none
set config-unix-directory = /local/configs/

#
# The following lines provide full screen emulator session for a 1024x768
# display. For other options, see the Network Station documentation.
```

```
set exec-command-menu = {
    { "5250" "ns5250 0.0.0.0 -wide disable -geometry 1024x768+0+0" }
    { "3270" "ns3270 0.0.0.0 -wide disable -geometry 1024x768+0+0" }
    { "vt220" "nsterm -tn vt220 -ctype telnet -fn 10x20 -geometry 1024x768+0+0 -host
}
set exec-startup-commands = {
    { "mcuis" }
    { "mwm" }
    { "ns5250 0.0.0.0 -wide disable -geometry 1024x768+0+0" }
    { "ns3270 0.0.0.0 -wide disable -geometry 1024x768+0+0" }
    { "nsterm -tn vt220 -ctype telnet -fn 10x20 -geometry 1024x768+0+0 -host 0.0.0.0
}
set file-enable-nfs-server = true
set file-export-directory-list = { { "/peerboot" "/local" } }
set file-nfs-access-control-default = read-write
set file-try-all-matches-on-open = true
set modules-directory = /local/mods
#
# The following 'set' line defines a local parallel printer.
# Uncomment/update this line if you have a parallel printer
# attached to the Network Station.
# Replace  with ps, pcl, or asc depending on whether you
# have a Postscript, PCL, or ASCII printer.
#set print-lpr-servers[-1]={"localhost" "PARALLEL1" " " "Local Parallel Printer" nil
#
# The following 'set' line defines a local serial printer.
# Uncomment/update this line if you have a serial printer
# attached to the Network Station.
# Replace  with ps, pcl, or asc depending on whether you
# have a Postscript, PCL, or ASCII printer.
#set print-lpr-servers[-1]={"localhost" "SERIAL1" " " "Local Serial Printer" nil nil
#
# The following 'set' line defines a remote printer.
# Replace 0.0.0.0 with the IP address of a remote printer.
# Replace  with the queue name on the remote printer.
# Replace  with ps, pcl, or asc depending on whether you
# have a Postscript, PCL, or ASCII printer.
# You may add additional lines if you want to define more that one
# remote printer.
#set print-lpr-servers[-1]={"0.0.0.0" " " "" "Remote Printer" nil nil nil nil false

set pref-environment = {
# Environment variables
    { "BOOTPATH" "/local" }
    { "HOME" "/local/nsm/users/nobody" }
    { "IMKEYMAPPATH" "/local/nls" }
    { "LANG" "EN_US" }
    { "LC_CTYPE" "" }
    { "LC_MESSAGES" "" }
    { "LC_MONETARY" "" }
    { "LC_NUMERIC" "" }
    { "LC_TIME" "" }
    { "LOCALENAME" "EN_US" }
    { "LOCPATH" "/local" }
    { "NLSPATH" "/local/nls/%L/msg/%N" }
    { "NSM_ADMIN_SYSDEFAULTS" "/local/nsm/SysDef" }
    { "NSM_GROUP_PREFS" "/local/SysDef" }
    { "NSM_NS3270_PREF_VERSION" "R3M0" }
    { "NSM_NS5250_PREF_VERSION" "R3M0" }
    { "NSM_PROD_SYSDEFAULTS" "/local/SysDef" }
    { "NSM_USERCONFIG_SERVER" "/local/nsm" }
    { "NSM_USER_PREFS" "/local/nsm/users/nobody" }
    { "PATH" "/local/mods" }
```

```
        { "USER" "nobody" }
        { "GROUP" "staff" }
        { "XLOCALEDIR" "/local/X11/locale" }
        { "XMODIFIERS" "" }
}
set pref-screen-background-bitmap-file = "/local/SysDef/ibmwall.xbm"
set pref-screensaver-bitmap-file = "/local/SysDef/ibmwall.xbm"
set pref-screensaver-enable=true
set pref-screensaver-time=1800

set xserver-default-font-path = {
        { "/local/fonts/pcf/75dpi" }
        { "/local/X11/fonts/pcf/i18n" }
        { "built-ins" }
}
#
# You can modify these X resources to change colors, remove drop down
# menus, etc.
set xserver-initial-x-resources = "ncdconsole.disableReboot: true\n\
        ncdconsole.disableLoginMenu: true\n\
        ncdconsole.disableTerminalMenu: true\n
        ncdconsole.disableSetupMenu: true\n\
        Mwm*activeBackground: #80FF80\n\
        Mwm*useIconBox: False\n\
        Mwm*background: #C0C0C0\n\
        Mwm*clientDecoration: None\n\
        Mwm*iconPlacement: left bottom\n\
        Mwm*foreground: #000000\n\
        Mwm*keyboardFocusPolicy: explicit\n\
        Mwm*fontList: *dt*interface*system*-medium-r*-s*:\n\
        Mwm*resizeBorderWidth: 5\n\
        *marginHeight: 0\n\
        NS3270*KeyRemap: disable\n\
        NS3270*KeymapPath: 0\n\
        NS3270*ColorMap: basic\n\
        NS3270*ColorMapPath: 0\n\
        NS3270*rows: 32\n\
        NS3270*cols: 80\n\
        NS3270*buttonBox: true\n\
        NS3270*Speckey: false\n\
        NS3270*Port: 23\n\
        NS3270*Graphics: false\n\
        NS3270*DefaultColorMapPath: 0\n\
        NS3270*PlayBack: disable_and_hide\n\
        NS3270*PlayBackPath: 0\n\
        NS3270*MiscPref: enable\n\
        NS3270*LocalPrint: enable\n\
        NS3270*Edit: enable\n\
        NS3270*Command: enable\n\
        NS3270*Option: enable\n\
        NS3270*Help: enable\n\
        NS3270*FontMenu: enable\n\
        NS3270*ChangeIPAddress: enable\n\
        NS5250*27x132: enable\n\
        NS5250*ImageView: disable\n\
        NS5250*ColumnSeparator: disable\n\
        NS5250*KeyRemap: disable\n\
        NS5250*ColorMap: basic\n\
        NS5250*KeyPad: enable\n\
        NS5250*PlayBack: disable_and_hide\n\
        NS5250*PlayBackPath: 0\n\
        NS5250*Record: disable\n\
        NS5250*Control: enable\n\
```

```
    NS5250*Edit: enable\n\
    NS5250*LocalPrint: enable\n\
    NS5250*ChangeIPAddress: enable\n\
    NS5250*Command: enable\n\
    NS5250*KeymapPath: 0\n\
    NS5250*ColorMapPath: 0\n\
    NS5250*DefaultColorMapPath: 0\n\
    NS5250*MiscPref: enable\n\
    NS5250*Option: enable\n\
    NS5250*Help: enable\n\
    NS5250*FontMenu: enable\n"
set xserver-keysym-file = /local/XKeysymDB
set xserver-rgb-file = /local/rgb.txt
#
# Uncomment these lines for debug only
#set diag-buffer-size = 65535
#set file-extended-diagnostics = true
```

The /local/peer/standard.nsm File

The **/local/peer/standard.nsm** file reads all of the other configuration files by
reading **/local/configs/standard.nsm** and then overrides values.

```
#
# standard.nsm for peer emulator scenario
#
read /peerboot/configs/standard.nsm
set boot-desired-source = nfs
set boot-nfs-directory = /peerboot/
set boot-prom-update-file="/peerboot/proms/bootm"
set boot-token-ring-update-file="/peerboot/proms/trflash"
set file-enable-nfs-server = false
set modules-directory = /peerboot/mods
set config-unix-directory = /peerboot/peer/

set pref-environment = {
#    Environment variables
    { "BOOTPATH" "/peerboot" }
    { "HOME" "/peerboot/nsm/users/nobody" }
    { "IMKEYMAPPATH" "/peerboot/nls" }
    { "LANG" "EN_US" }
    { "LC_CTYPE" "" }
    { "LC_MESSAGES" "" }
    { "LC_MONETARY" "" }
    { "LC_NUMERIC" "" }
    { "LC_TIME" "" }
    { "LOCALENAME" "EN_US" }
    { "LOCPATH" "/peerboot" }
    { "NLSPATH" "/peerboot/nls/%L/msg/%N" }
    { "NSM_ADMIN_SYSDEFAULTS" "/peerboot/SysDef" }
    { "NSM_GROUP_PREFS" "/peerboot/SysDef" }
    { "NSM_NS5250_PREF_VERSION" "R3M0" }
    { "NSM_PROD_SYSDEFAULTS" "/peerboot/SysDef" }
    { "NSM_USERCONFIG_SERVER" "/peerboot/nsm" }
    { "NSM_USER_PREFS" "/peerboot/nsm/users/nobody" }
    { "PATH" "/peerboot/mods" }
    { "USER" "nobody" }
    { "GROUP" "staff" }
    { "XLOCALEDIR" "/peerboot/X11/locale" }
    { "XMODIFIERS" "" }
}
```

```
set pref-screen-background-bitmap-file = "/peerboot/SysDef/ibmwall.xbm"
set pref-screensaver-bitmap-file = "/peerboot/SysDef/ibmwall.xbm"

set xserver-default-font-path = {
    { "/peerboot/fonts/pcf/75dpi" }
    { "/peerboot/X11/fonts/pcf/i18n" }
    { "built-ins" }
}
set xserver-keysym-file = /peerboot/XKeysymDB
set xserver-rgb-file = /peerboot/rgb.txt
```

Files Required

```
XKeysymDB
kernel.63Z
kernel.Z
rgb.txt
SysDef
  ibmwall.xbm
  NS3270
  NS5250
X11
  app-defaults
    Mcuis          Mwm              XApplication system.mwmrc
  fonts
    pcf
      i18n
        Block11.iso1_UCS.pcf.Z    ErgoB15.ucs.pcf.Z        helvB14.iso1_UCS.pcf.Z
        Block11E.iso1_UCS.pcf.Z   ErgoI15.ucs.pcf.Z        helvBO10.iso1_UCS.pcf.Z
        Block11WE.iso1_UCS.pcf.Z  Rom14.base_UCS.pcf.Z     helvBO12.iso1_UCS.pcf.Z
        Block17.iso1_UCS.pcf.Z    Rom14.iso1_UCS.pcf.Z     helvBO14.iso1_UCS.pcf.Z
        Ergo11.ucs.pcf.Z          Rom22.base_UCS.pcf.Z     helvO10.iso1_UCS.pcf.Z
        Ergo15.iso1_UCS.pcf.Z     Rom22.iso1_UCS.pcf.Z     helvO12.iso1_UCS.pcf.Z
        Ergo15.ucs.pcf.Z          Rom8.iso1_UCS.pcf.Z      helvO14.iso1_UCS.pcf.Z
        Ergo17.iso1_UCS.pcf.Z     fonts.alias              helvR10.iso1_UCS.pcf.Z
        Ergo17.ucs.pcf.Z          fonts.dir                helvR12.iso1_UCS.pcf.Z
        Ergo17E.iso1_UCS.pcf.Z    helvB08.iso1_UCS.pcf.Z   helvR14.iso1_UCS.pcf.Z
        Ergo17E.ucs.pcf.Z         helvB10.iso1_UCS.pcf.Z
        Ergo22.ucs.pcf.Z          helvB12.iso1_UCS.pcf.Z
  locale
    locale.alias locale.dir
    UTF-8_BASE-0
      XLC_LOCALE
    UTF-8_C
      XLC_LOCALE
    UTF-8_iso8859-1
      XLC_LOCALE
configs
  control.nsm  defaults.dft standard.nsm hosts.nsm
  defaults.nsm required.nsm
fonts
  pcf
    75dpi
      cour1.pcf.Z     courBO08.pcf.Z cour05.pcf.Z   helvB10.pcf.Z  helvBO6.pcf.Z
      cour2.pcf.Z     courBO1.pcf.Z  cour056.pcf.Z  helvB12.pcf.Z  helvBO64.pcf.Z
      cour3.pcf.Z     courBO10.pcf.Z cour06.pcf.Z   helvB14.pcf.Z  helvBO7.pcf.Z
      cour4.pcf.Z     courBO12.pcf.Z cour064.pcf.Z  helvB18.pcf.Z  helvBO72.pcf.Z
      cour5.pcf.Z     courBO14.pcf.Z cour07.pcf.Z   helvB2.pcf.Z   helvO08.pcf.Z
      cour56.pcf.Z    courBO18.pcf.Z cour072.pcf.Z  helvB24.pcf.Z  helvO1.pcf.Z
      cour6.pcf.Z     courBO2.pcf.Z  courR08.pcf.Z  helvB3.pcf.Z   helvO10.pcf.Z
      cour64.pcf.Z    courBO24.pcf.Z courR10.pcf.Z  helvB4.pcf.Z   helvO12.pcf.Z
      cour7.pcf.Z     courBO3.pcf.Z  courR12.pcf.Z  helvB5.pcf.Z   helvO14.pcf.Z
```

```
            cour72.pcf.Z    courBO4.pcf.Z   courR14.pcf.Z  helvB56.pcf.Z  helvO18.pcf.Z
            courB08.pcf.Z   courBO5.pcf.Z   courR18.pcf.Z  helvB6.pcf.Z   helvO2.pcf.Z
            courB1.pcf.Z    courBO56.pcf.Z  courR24.pcf.Z  helvB64.pcf.Z  helvO24.pcf.Z
            courB10.pcf.Z   courBO6.pcf.Z   fonts.alias    helvB7.pcf.Z   helvO3.pcf.Z
            courB12.pcf.Z   courBO64.pcf.Z  fonts.dir      helvB72.pcf.Z  helvO4.pcf.Z
            courB14.pcf.Z   courBO7.pcf.Z   helv1.pcf.Z    helvBO08.pcf.Z helvO5.pcf.Z
            courB18.pcf.Z   courBO72.pcf.Z  helv2.pcf.Z    helvBO1.pcf.Z  helvO56.pcf.Z
            courB2.pcf.Z    courO08.pcf.Z   helv3.pcf.Z    helvBO10.pcf.Z helvO6.pcf.Z
            courB24.pcf.Z   courO1.pcf.Z    helv4.pcf.Z    helvBO12.pcf.Z helvO64.pcf.Z
            courB3.pcf.Z    courO10.pcf.Z   helv5.pcf.Z    helvBO14.pcf.Z helvO7.pcf.Z
            courB4.pcf.Z    courO12.pcf.Z   helv56.pcf.Z   helvBO18.pcf.Z helvO72.pcf.Z
            courB5.pcf.Z    courO14.pcf.Z   helv6.pcf.Z    helvBO2.pcf.Z  helvR08.pcf.Z
            courB56.pcf.Z   courO18.pcf.Z   helv64.pcf.Z   helvBO24.pcf.Z helvR10.pcf.Z
            courB6.pcf.Z    courO2.pcf.Z    helv7.pcf.Z    helvBO3.pcf.Z  helvR12.pcf.Z
            courB64.pcf.Z   courO24.pcf.Z   helv72.pcf.Z   helvBO4.pcf.Z  helvR14.pcf.Z
            courB7.pcf.Z    courO3.pcf.Z    helvB08.pcf.Z  helvBO5.pcf.Z  helvR18.pcf.Z
            courB72.pcf.Z   courO4.pcf.Z    helvB1.pcf.Z   helvBO56.pcf.Z helvR24.pcf.Z
keyboards
  AB83useng AB83usiso BFBFuseng
mods
  colormap.nws   keypad.nws     lpd.nws         mwm.nws         nsterm.nws
  export.63a     libconf.nws    lprd.nws        nfsd.nws        sbcs_im.nws
  export.nws     libmlc.nws     mcuis.nws       ns3270.nws      seriald.nws
  helpview.nws   libprapi.nws   miscpr32.nws    ns5250.nws      setup.nws
  keymap52.nws   libprxapi.nws  miscpref.nws    ns5250xx.nws    term.nws
nls
  EN_US.im         EN_US.imkeymap sbcs.imcompose
  C¹
    msg
      X.cat   Xm.cat Xt.cat
  EN_US
    MRI
      N3_5ClrH N3_5Help N3_5KeyH N3_5KyPH N3_5MscP
      N3_5ColM N3_5HlpV N3_5KeyM N3_5LocP N3_5Resc
    msg
      X.cat           Xt.cat          libprxapi.cat mwm.cat
      Xm.cat          common.cat      mcuis.cat     term.cat
  codepage
    IBM1208 IBM37   IBM819
nsm
  users
    nobody
      NS3270
        K²
      NS5250
        K³
proms
  bootm         bootm.63a    trflash      trflash.43b trflash.63a
```

Notes:

[1] This directory does not exist on Windows systems. Copy the corresponding **msg** directory and files from the **EN_US** directory into the **C** directory when staging the data. If you do not have the **C** directory, boot times may be longer.

[2] If you are using a 3270 keyboard remap file, it must be placed in ***NSM_USER_PREFS*/NS3270/K**

[3] If you are using a 5250 keyboard remap file, it must be placed in ***NSM_USER_PREFS*/NS5250/K**

NC Navigator with JVM

In this scenario, the Network Station is booted and the NC Navigator browser is available along with the Java Virtual Machine to run applets. This setup requires 24 Mbytes on the card if the compressed kernels for all systems are to be used. Space can be reduced by eliminating unused fonts, but this has to be done by trial and error.

Configuring NC Navigator to run from a flash card without relying on ACTLogin to set up the environment requires a great deal of skill and fortitude. The easiest way to accomplish this is to use NSM to set up all of the network and browser characteristics for a single user. Then copy that user's preferences and other browser files to the flash card. In this section, the userid will be indicated as *nobody*.

When booting from a flash card and running NC Navigator without an NSM server, you are forced to run Navigator in a *read-only* mode. If you did not run in read-only, bookmarks, history, and other Navigator preferences would be saved to the flash card. Eventually the flash card would run out of space. Therefore, when running this configuration, bookmarks and other preferences are not allowed to be saved. In addition, you are not allowed to use disk caching.

Note: The **preferences** file is overwritten when NC Navigator starts. You should turn on the write-protect on the flash card so that the card does not eventually fill up. Putting a dummy **lock** file on the card and setting the write-protect switch will cause NC Navigator to output several error/warning messages when starting. Just respond to the messages and NC Navigator will continue to load. There is no workaround for this behavior in the current release.

More information on configuring NC Navigator for the Network Station can be found in the NC Navigator 3 for Release 3 Workbook (http://w3.rchland.ibm.com/~jepe/navio.html). Disk caching should not be used when running NC Navigator with flash. The system is shipped with disk caching disabled, and this setting should not be changed.

Note that this is a setup using US English as the language. For other languages/locales, different files would be required under X11/locales and keyboards directories.

Files Required

```
XKeysymDB
kernel.63Z
kernel.Z
rgb.txt
SysDef
  ibmwall.xbm
  NAV
    pref
X11
  EN_US
    app-defaults
      Navio
  app-defaults
    Mcuis        Mwm           XApplication system.mwmrc
  fonts
    pcf
      i18n
```

```
                Block11.iso1_UCS.pcf.Z   courO12.iso1_UCS.pcf.Z   helvBO12.iso1_UCS.pcf.Z
                Block17.iso1_UCS.pcf.Z   courR12.iso1_UCS.pcf.Z   helvBO14.iso1_UCS.pcf.Z
                Ergo15.iso1_UCS.pcf.Z    fonts.alias              helvO10.iso1_UCS.pcf.Z
                Ergo17.iso1_UCS.pcf.Z    fonts.dir                helvO12.iso1_UCS.pcf.Z
                Rom14.base_UCS.pcf.Z     helvB08.iso1_UCS.pcf.Z   helvO14.iso1_UCS.pcf.Z
                Rom14.iso1_UCS.pcf.Z     helvB10.iso1_UCS.pcf.Z   helvR10.iso1_UCS.pcf.Z
                Rom22.base_UCS.pcf.Z     helvB12.iso1_UCS.pcf.Z   helvR12.iso1_UCS.pcf.Z
                Rom22.iso1_UCS.pcf.Z     helvB14.iso1_UCS.pcf.Z   helvR14.iso1_UCS.pcf.Z
                Rom8.iso1_UCS.pcf.Z      helvBO10.iso1_UCS.pcf.Z
        locale
          locale.alias locale.dir
          UTF-8_BASE-0
            XLC_LOCALE
          UTF-8_C
            XLC_LOCALE
          UTF-8_iso8859-1
            XLC_LOCALE
configs
  control.nsm  defaults.dft defaults.nsm hosts.nsm     required.nsm standard.nsm
fonts
  pcf
    75dpi
      ARIABD32.pcf.Z courBO14.pcf.Z helvO08.pcf.Z  luIS12.pcf.Z   sy1.pcf.Z
      ARIABD40.pcf.Z courBO18.pcf.Z helvO1.pcf.Z   luIS14.pcf.Z   sy2.pcf.Z
      ARIABD48.pcf.Z courBO2.pcf.Z  helvO10.pcf.Z  luIS18.pcf.Z   sy3.pcf.Z
      ARIABD56.pcf.Z courBO24.pcf.Z helvO12.pcf.Z  luIS19.pcf.Z   sy31.pcf.Z
      ARIABD64.pcf.Z courBO3.pcf.Z  helvO14.pcf.Z  luIS24.pcf.Z   sy36.pcf.Z
      ARIABD72.pcf.Z courBO4.pcf.Z  helvO18.pcf.Z  luRS08.pcf.Z   sy38.pcf.Z
      ARIABI32.pcf.Z courBO5.pcf.Z  helvO2.pcf.Z   luRS10.pcf.Z   sy4.pcf.Z
      ARIABI40.pcf.Z courBO56.pcf.Z helvO24.pcf.Z  luRS12.pcf.Z   sy48.pcf.Z
      ARIABI48.pcf.Z courBO6.pcf.Z  helvO3.pcf.Z   luRS14.pcf.Z   sy5.pcf.Z
      ARIABI56.pcf.Z courBO64.pcf.Z helvO4.pcf.Z   luRS18.pcf.Z   sy56.pcf.Z
      ARIABI64.pcf.Z courBO7.pcf.Z  helvO5.pcf.Z   luRS19.pcf.Z   sy6.pcf.Z
      ARIABI72.pcf.Z courBO72.pcf.Z helvO56.pcf.Z  luRS24.pcf.Z   sy64.pcf.Z
      ARIAIT32.pcf.Z courO08.pcf.Z  helvO6.pcf.Z   lubB08.pcf.Z   sy7.pcf.Z
      ARIAIT40.pcf.Z courO1.pcf.Z   helvO64.pcf.Z  lubB10.pcf.Z   sy72.pcf.Z
      ARIAIT48.pcf.Z courO10.pcf.Z  helvO7.pcf.Z   lubB12.pcf.Z   symb08.pcf.Z
      ARIAIT56.pcf.Z courO12.pcf.Z  helvO72.pcf.Z  lubB14.pcf.Z   symb10.pcf.Z
      ARIAIT64.pcf.Z courO14.pcf.Z  helvR08.pcf.Z  lubB18.pcf.Z   symb12.pcf.Z
      ARIAIT72.pcf.Z courO18.pcf.Z  helvR10.pcf.Z  lubB19.pcf.Z   symb14.pcf.Z
      ARIAL32.pcf.Z  courO2.pcf.Z   helvR12.pcf.Z  lubB24.pcf.Z   symb18.pcf.Z
      ARIAL40.pcf.Z  courO24.pcf.Z  helvR14.pcf.Z  lubBI08.pcf.Z  symb24.pcf.Z
      ARIAL48.pcf.Z  courO3.pcf.Z   helvR18.pcf.Z  lubBI10.pcf.Z  tech14.pcf.Z
      ARIAL56.pcf.Z  courO4.pcf.Z   helvR24.pcf.Z  lubBI12.pcf.Z  techB14.pcf.Z
      ARIAL64.pcf.Z  courO5.pcf.Z   lBIS1.pcf.Z    lubBI14.pcf.Z  term14.pcf.Z
      ARIAL72.pcf.Z  courO56.pcf.Z  lBIS2.pcf.Z    lubBI18.pcf.Z  termB14.pcf.Z
      charB08.pcf.Z  courO6.pcf.Z   lBIS3.pcf.Z    lubBI19.pcf.Z  timB08.pcf.Z
      charB10.pcf.Z  courO64.pcf.Z  lBIS4.pcf.Z    lubBI24.pcf.Z  timB1.pcf.Z
      charB12.pcf.Z  courO7.pcf.Z   lBIS5.pcf.Z    lubI08.pcf.Z   timB10.pcf.Z
      charB14.pcf.Z  courO72.pcf.Z  lBIS56.pcf.Z   lubI10.pcf.Z   timB12.pcf.Z
      charB18.pcf.Z  courR08.pcf.Z  lBIS6.pcf.Z    lubI12.pcf.Z   timB14.pcf.Z
      charB24.pcf.Z  courR10.pcf.Z  lBIS64.pcf.Z   lubI14.pcf.Z   timB18.pcf.Z
      charBI08.pcf.Z courR12.pcf.Z  lBIS7.pcf.Z    lubI18.pcf.Z   timB2.pcf.Z
      charBI10.pcf.Z courR14.pcf.Z  lBIS72.pcf.Z   lubI19.pcf.Z   timB24.pcf.Z
      charBI12.pcf.Z courR18.pcf.Z  lBS1.pcf.Z     lubI24.pcf.Z   timB3.pcf.Z
      charBI14.pcf.Z courR24.pcf.Z  lBS2.pcf.Z     lubR08.pcf.Z   timB4.pcf.Z
      charBI18.pcf.Z fonts.alias    lBS3.pcf.Z     lubR10.pcf.Z   timB5.pcf.Z
      charBI24.pcf.Z fonts.dir      lBS4.pcf.Z     lubR12.pcf.Z   timB6.pcf.Z
      charI08.pcf.Z  helv1.pcf.Z    lBS5.pcf.Z     lubR14.pcf.Z   timB7.pcf.Z
      charI10.pcf.Z  helv2.pcf.Z    lBS56.pcf.Z    lubR18.pcf.Z   timBI08.pcf.Z
      charI12.pcf.Z  helv3.pcf.Z    lBS6.pcf.Z     lubR19.pcf.Z   timBI1.pcf.Z
      charI14.pcf.Z  helv4.pcf.Z    lBS64.pcf.Z    lubR24.pcf.Z   timBI10.pcf.Z
      charI18.pcf.Z  helv5.pcf.Z    lBS7.pcf.Z     lutBS08.pcf.Z  timBI12.pcf.Z
```

```
charI24.pcf.Z    helv56.pcf.Z    lBS72.pcf.Z     lutBS10.pcf.Z     timBI14.pcf.Z
charR08.pcf.Z    helv6.pcf.Z     lIS1.pcf.Z      lutBS12.pcf.Z     timBI18.pcf.Z
charR10.pcf.Z    helv64.pcf.Z    lIS2.pcf.Z      lutBS14.pcf.Z     timBI2.pcf.Z
charR12.pcf.Z    helv7.pcf.Z     lIS3.pcf.Z      lutBS18.pcf.Z     timBI24.pcf.Z
charR14.pcf.Z    helv72.pcf.Z    lIS4.pcf.Z      lutBS19.pcf.Z     timBI3.pcf.Z
charR18.pcf.Z    helvB08.pcf.Z   lIS5.pcf.Z      lutBS24.pcf.Z     timBI4.pcf.Z
charR24.pcf.Z    helvB1.pcf.Z    lIS56.pcf.Z     lutRS08.pcf.Z     timBI5.pcf.Z
cour1.pcf.Z      helvB10.pcf.Z   lIS6.pcf.Z      lutRS10.pcf.Z     timBI6.pcf.Z
cour2.pcf.Z      helvB12.pcf.Z   lIS64.pcf.Z     lutRS12.pcf.Z     timBI7.pcf.Z
cour3.pcf.Z      helvB14.pcf.Z   lIS7.pcf.Z      lutRS14.pcf.Z     timI08.pcf.Z
cour4.pcf.Z      helvB18.pcf.Z   lIS72.pcf.Z     lutRS18.pcf.Z     timI1.pcf.Z
cour5.pcf.Z      helvB2.pcf.Z    lRS1.pcf.Z      lutRS19.pcf.Z     timI10.pcf.Z
cour56.pcf.Z     helvB24.pcf.Z   lRS2.pcf.Z      lutRS24.pcf.Z     timI12.pcf.Z
cour6.pcf.Z      helvB3.pcf.Z    lRS3.pcf.Z      ncenB08.pcf.Z     timI14.pcf.Z
cour64.pcf.Z     helvB4.pcf.Z    lRS4.pcf.Z      ncenB10.pcf.Z     timI18.pcf.Z
cour7.pcf.Z      helvB5.pcf.Z    lRS5.pcf.Z      ncenB12.pcf.Z     timI2.pcf.Z
cour72.pcf.Z     helvB56.pcf.Z   lRS56.pcf.Z     ncenB14.pcf.Z     timI24.pcf.Z
courB08.pcf.Z    helvB6.pcf.Z    lRS6.pcf.Z      ncenB18.pcf.Z     timI3.pcf.Z
courB1.pcf.Z     helvB64.pcf.Z   lRS64.pcf.Z     ncenB24.pcf.Z     timI4.pcf.Z
courB10.pcf.Z    helvB7.pcf.Z    lRS7.pcf.Z      ncenBI08.pcf.Z timI5.pcf.Z
courB12.pcf.Z    helvB72.pcf.Z   lRS72.pcf.Z     ncenBI10.pcf.Z timI6.pcf.Z
courB14.pcf.Z    helvBO08.pcf.Z luBIS08.pcf.Z   ncenBI12.pcf.Z timI7.pcf.Z
courB18.pcf.Z    helvBO1.pcf.Z   luBIS10.pcf.Z   ncenBI14.pcf.Z timR08.pcf.Z
courB2.pcf.Z     helvBO10.pcf.Z luBIS12.pcf.Z   ncenBI18.pcf.Z timR1.pcf.Z
courB24.pcf.Z    helvBO12.pcf.Z luBIS14.pcf.Z   ncenBI24.pcf.Z timR10.pcf.Z
courB3.pcf.Z     helvBO14.pcf.Z luBIS18.pcf.Z   ncenI08.pcf.Z     timR12.pcf.Z
courB4.pcf.Z     helvBO18.pcf.Z luBIS19.pcf.Z   ncenI10.pcf.Z     timR14.pcf.Z
courB5.pcf.Z     helvBO2.pcf.Z   luBIS24.pcf.Z   ncenI12.pcf.Z     timR18.pcf.Z
courB56.pcf.Z    helvBO24.pcf.Z luBS08.pcf.Z    ncenI14.pcf.Z     timR2.pcf.Z
courB6.pcf.Z     helvBO3.pcf.Z   luBS10.pcf.Z    ncenI18.pcf.Z     timR24.pcf.Z
courB64.pcf.Z    helvBO4.pcf.Z   luBS12.pcf.Z    ncenI24.pcf.Z     timR3.pcf.Z
courB7.pcf.Z     helvBO5.pcf.Z   luBS14.pcf.Z    ncenR08.pcf.Z     timR4.pcf.Z
courB72.pcf.Z    helvBO56.pcf.Z luBS18.pcf.Z    ncenR10.pcf.Z     timR5.pcf.Z
courBO08.pcf.Z helvBO6.pcf.Z   luBS19.pcf.Z    ncenR12.pcf.Z     timR6.pcf.Z
courBO1.pcf.Z   helvBO64.pcf.Z luBS24.pcf.Z    ncenR14.pcf.Z     timR7.pcf.Z
courBO10.pcf.Z helvBO7.pcf.Z   luIS08.pcf.Z    ncenR18.pcf.Z
courBO12.pcf.Z helvBO72.pcf.Z luIS10.pcf.Z    ncenR24.pcf.Z
java
  courB16.pcf.Z    fixedR16.pcf.Z  luBIS36.pcf.Z   lubI40.pcf.Z      ncenI40.pcf.Z
  courB20.pcf.Z    fixedR18.pcf.Z  luBIS40.pcf.Z   lubI48.pcf.Z      ncenI48.pcf.Z
  courB28.pcf.Z    fixedR28.pcf.Z  luBIS48.pcf.Z   lubR16.pcf.Z      ncenR16.pcf.Z
  courB32.pcf.Z    fixedR32.pcf.Z  luBS16.pcf.Z    lubR20.pcf.Z      ncenR20.pcf.Z
  courB36.pcf.Z    fixedR36.pcf.Z  luBS28.pcf.Z    lubR28.pcf.Z      ncenR28.pcf.Z
  courB40.pcf.Z    fixedR40.pcf.Z  luBS32.pcf.Z    lubR32.pcf.Z      ncenR32.pcf.Z
  courB48.pcf.Z    fixedR48.pcf.Z  luBS36.pcf.Z    lubR36.pcf.Z      ncenR36.pcf.Z
  courBO16.pcf.Z fonts.alias     luBS40.pcf.Z    lubR40.pcf.Z      ncenR40.pcf.Z
  courBO20.pcf.Z fonts.dir       luBS48.pcf.Z    lubR48.pcf.Z      ncenR48.pcf.Z
  courBO28.pcf.Z helvB16.pcf.Z   luIS16.pcf.Z    lutBS16.pcf.Z     timB16.pcf.Z
  courBO32.pcf.Z helvB20.pcf.Z   luIS28.pcf.Z    lutBS28.pcf.Z     timB20.pcf.Z
  courBO36.pcf.Z helvB28.pcf.Z   luIS32.pcf.Z    lutBS32.pcf.Z     timB28.pcf.Z
  courBO40.pcf.Z helvB32.pcf.Z   luIS36.pcf.Z    lutBS36.pcf.Z     timB32.pcf.Z
  courBO48.pcf.Z helvB36.pcf.Z   luIS40.pcf.Z    lutBS40.pcf.Z     timB36.pcf.Z
  courO16.pcf.Z    helvB40.pcf.Z   luIS48.pcf.Z    lutBS48.pcf.Z     timB40.pcf.Z
  courO20.pcf.Z    helvB48.pcf.Z   luRS16.pcf.Z    lutRS16.pcf.Z     timB48.pcf.Z
  courO28.pcf.Z    helvBO16.pcf.Z luRS28.pcf.Z    lutRS28.pcf.Z     timBI16.pcf.Z
  courO32.pcf.Z    helvBO20.pcf.Z luRS32.pcf.Z    lutRS32.pcf.Z     timBI20.pcf.Z
  courO36.pcf.Z    helvBO28.pcf.Z luRS36.pcf.Z    lutRS36.pcf.Z     timBI28.pcf.Z
  courO40.pcf.Z    helvBO32.pcf.Z luRS40.pcf.Z    lutRS40.pcf.Z     timBI32.pcf.Z
  courO48.pcf.Z    helvBO36.pcf.Z luRS48.pcf.Z    lutRS48.pcf.Z     timBI36.pcf.Z
  courR16.pcf.Z    helvBO40.pcf.Z lubB16.pcf.Z    ncenB16.pcf.Z     timBI40.pcf.Z
  courR20.pcf.Z    helvBO48.pcf.Z lubB20.pcf.Z    ncenB20.pcf.Z     timBI48.pcf.Z
  courR28.pcf.Z    helvO16.pcf.Z   lubB28.pcf.Z    ncenB28.pcf.Z     timI16.pcf.Z
  courR32.pcf.Z    helvO20.pcf.Z   lubB32.pcf.Z    ncenB32.pcf.Z     timI20.pcf.Z
```

```
        courR36.pcf.Z  helvO28.pcf.Z  lubB36.pcf.Z   ncenB36.pcf.Z  timI28.pcf.Z
        courR40.pcf.Z  helvO32.pcf.Z  lubB40.pcf.Z   ncenB40.pcf.Z  timI32.pcf.Z
        courR48.pcf.Z  helvO36.pcf.Z  lubB48.pcf.Z   ncenB48.pcf.Z  timI36.pcf.Z
        fixedB08.pcf.Z helvO40.pcf.Z  lubBI16.pcf.Z  ncenBI16.pcf.Z timI40.pcf.Z
        fixedB10.pcf.Z helvO48.pcf.Z  lubBI20.pcf.Z  ncenBI20.pcf.Z timI48.pcf.Z
        fixedB12.pcf.Z helvR16.pcf.Z  lubBI28.pcf.Z  ncenBI28.pcf.Z timR16.pcf.Z
        fixedB16.pcf.Z helvR20.pcf.Z  lubBI32.pcf.Z  ncenBI32.pcf.Z timR20.pcf.Z
        fixedB18.pcf.Z helvR28.pcf.Z  lubBI36.pcf.Z  ncenBI36.pcf.Z timR28.pcf.Z
        fixedB20.pcf.Z helvR32.pcf.Z  lubBI40.pcf.Z  ncenBI40.pcf.Z timR32.pcf.Z
        fixedB24.pcf.Z helvR36.pcf.Z  lubBI48.pcf.Z  ncenBI48.pcf.Z timR36.pcf.Z
        fixedB28.pcf.Z helvR40.pcf.Z  lubI16.pcf.Z   ncenI16.pcf.Z  timR40.pcf.Z
        fixedB32.pcf.Z helvR48.pcf.Z  lubI20.pcf.Z   ncenI20.pcf.Z  timR48.pcf.Z
        fixedB36.pcf.Z luBIS16.pcf.Z  lubI28.pcf.Z   ncenI28.pcf.Z
        fixedB40.pcf.Z luBIS28.pcf.Z  lubI32.pcf.Z   ncenI32.pcf.Z
        fixedB48.pcf.Z luBIS32.pcf.Z  lubI36.pcf.Z   ncenI36.pcf.Z
java
  classes.zip  javacpa0.gif javacpat.gif nwshacl.zip  nwspackg.zip
  lib
    appletviewer.properties  font.properties          rmic.properties
    awt.properties           font.properties.en       serialver.properties
    content-types.properties javac.properties
    security
      java.security
keyboards
  AB83useng
mods
  desktop.nws        jjpeg.63a         jzip.nws          navio.nws
  export.63a         jjpeg.nws         libconf.nws       nfsd.nws
  export.nws         jmath.63a         libmlc.nws        sbcs_im.nws
  filed.nws          jmath.nws         libprapi.nws      seriald.nws
  helpview.nws       jmmedia.63a       libprxapi.nws     setup.nws
  java.63a           jmmedia.nws       loadb.nws         term.nws
  java.nws           jnet.63a          lpd.nws           test.nws
  jawt.63a           jnet.nws          lprd.nws
  jawt.nws           jsysresource.nws  mcuis.nws
  jcomm.nws          jzip.63a          mwm.nws
  NAV
    navio.zip pref
nls
  EN_US.im        EN_US.imkeymap sbcs.imcompose
  C¹
    msg
      X.cat   Xm.cat Xt.cat
  EN_US
    MRI
      TBack.pt.i.xpm       TNext.pt.xpm          about.htm
      TBack.pt.xpm         TOpenUrl.pt.xpm       header.gif
      TFind.pt.xpm         TPrint.pt.i.xpm       help.htm
      THome.pt.xpm         TPrint.pt.xpm         ibmlogo.gif
      TLoadImages.pt.i.xpm TReload.pt.xpm        index.htm
      TLoadImages.pt.xpm   TStop.pt.i.xpm        links.htm
      TNext.pt.i.xpm       TStop.pt.xpm          splash.htm
    msg
      X.cat          Xt.cat          java.cat       mcuis.cat     rprint.cat
      Xm.cat         common.cat      libprxapi.cat mwm.cat
  codepage
    IBM1208 IBM37    IBM819
nsm
  SysDef
    NAV
      pref
  users
    nobody
```

```
    NAV
       bookmarks.html cookies         key.db           pref
       cert5.db        history.db      lock             preferences
       tmp
```

Notes:
[1]This directory does not exist on Windows systems. Copy the corresponding msg
directory and files from the EN_US directory into the C directory when staging
the data. If you do not have the C directory, boot times may be longer.

The /local/configs/standard.nsm File

```
#
# standard.nsm for flash Navigator scenario
#
# Add/Remove the following "read" statments to match exactly the same files
# that are read by PRODBASE/configs/standard.nsm
# Also change any IP addresses, host names, and URLs to match your
# environment.
read required.nsm
read control.nsm
read hosts.nsm
read defaults.nsm
read defaults.dft
#
# Overrides for flash boot
set boot-desired-source = local
set boot-prom-update-file = nil
set boot-second-source = none
set boot-third-source = none
set config-unix-directory = /local/configs/
set diag-buffer-size = 65535

set exec-command-menu = {
    { "NC Navigator" "loadb navio" }
}
set exec-startup-commands = {
    { "mcuis" }
    { "mwm" }
    { "loadb navio" }
}
set file-enable-nfs-server = true
set file-export-directory-list = { { "/peerboot" "/local" } }
set file-nfs-access-control-default = read-write
set file-try-all-matches-on-open = true
set modules-directory = /local/mods

set pref-environment = {
# Environment variables
    { "BOOTPATH" "/local" }
    { "HOME" "/local/nsm/users/nobody" }
    { "IMKEYMAPPATH" "/local/nls" }
    { "LANG" "EN_US" }
    { "LC_CTYPE" "" }
    { "LC_MESSAGES" "" }
    { "LC_MONETARY" "" }
    { "LC_NUMERIC" "" }
    { "LC_TIME" "" }
    { "LOCALENAME" "EN_US" }
    { "LOCPATH" "/local" }
    { "NLSPATH" "/local/nls/%L/msg/%N" }
```

```
        { "NSM_ADMIN_SYSDEFAULTS" "/local/nsm/SysDef" }
        { "NSM_GROUP_PREFS" "/local/nsm/groups/staff" }
        { "NSM_NAV_PREF_VERSION" "R3M0" }
        { "NSM_PROD_SYSDEFAULTS" "/local/SysDef" }
        { "NSM_USER_PREFS" "/local/nsm/users/nobody" }
        { "NSM_USERCONFIG_SERVER" "/local/nsm/" }
        { "PATH" "/local/mods" }
        { "USER" "nobody" }
        { "GROUP" "staff" }
        { "XLOCALEDIR" "/local/X11/locale" }
        { "XMODIFIERS" "" }
#
# The following variables are used by NC Navigator.
# They should be set based on which features of the browser are required.
# See the NC Navigator Workbook for more details.
        { "FTP_PROXY_HOST" "" }
        { "FTP_PROXY_PORT" "" }
        { "FTP_PROXY_OVERRIDES" "" }
        { "HTTP_PROXY_HOST" "" }
        { "HTTP_PROXY_PORT" "" }
        { "HTTP_PROXY_OVERRIDES" "" }
        { "HTTPS_PROXY_HOST" "" }
        { "HTTPS_PROXY_PORT" "" }
        { "GOPHER_PROXY_HOST" "" }
        { "GOPHER_PROXY_PORT" "" }
        { "GOPHER_PROXY_OVERRIDES" "" }
        { "SOCKS_HOST" "" }
        { "SOCKS_PORT" "1080" }
        { "SMTP_SERVER" "" }
        { "NSM_HTTP_PORT" "80" }
        { "EMAIL_USERID" "" }
        { "FULL_NAME" "" }
        { "HOME_PAGE" "http://xxx" }
        { "REPLY_TO" "" }
        { "POP3_SERVER" "" }
        { "NNTP_SERVER" "" }
}
set pref-screen-background-bitmap-file = "/local/SysDef/ibmwall.xbm"
set pref-screensaver-bitmap-file = "/local/SysDef/ibmwall.xbm"

set xserver-default-font-path = {
        {"/local/fonts/pcf/75dpi"}
        {"/local/X11/fonts/pcf/i18n"}
        {"built-ins"}
        {"/local/fonts/pcf/java"}
}
set xserver-keysym-file = /local/XKeysymDB
set xserver-rgb-file = /local/rgb.txt
#
# debug only
#set exec-disabled-commands = { }
#set file-extended-diagnostics = true
#set xserver-initial-x-resources = "ncdconsole.disableReboot: false\n\
#    ncdconsole.disableLoginMenu: false\n\
#    ncdconsole.disableTerminalMenu: false\n\
#    ncdconsole.disableSetupMenu: false\n"
```

The /local/nsm/users/nobody/NAV/lock File

This lock file needs to have a MAC Address that is not one of the Network
Stations that will be used. You should be able to use the following as a lock
file.

```
00:00:a7:22:ff:a0-9.3.35.1-25
```

The /local/nsm/NAV/preferences File

Copy the files in *PRODBASE***/nsm/user/***nobody***/NAV** directory to the staging directory and modify any path names in the **preferences** file to reflect **/local** as the root directory instead of *PRODBASE*.

                        *** End of Document ***