Network Computer

IBM Network Station

Flash Memory Card Implementation for the IBM Network Station Release 3.0

Version 2.0.1 - January 15, 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.**

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 `Solutions` on the page header and then following the `Product Solutions` link.

- *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://www.ibm.com/nc/pubs.

- *Configuration Files*.
  This document may be accessed on the web at http://www.ibm.com/nc/pubs.

- *Full-Screen Solutions*.
  This document may be accessed on the web at http://www.ibm.com/nc/pubs.

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.
- 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*, *CONFIGBASE* and *CONFIGDIR. 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, CONFIGBASE and CONFIGDIR are shown in the following table.

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

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**, **configs**, **SysDef**, etc.) from the PRODBASE and CONFIGBASE directories from 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.*

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 `<LeftShift>-<LeftAlt>-<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. CONFIGBASE/**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` <Network Station_IP_address> `5996`
or the corresponding telnet command on your server.

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.

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. *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` *or* `Network` *depending on whether you want to use DHCP/BOOTP to assign the Network Station's address (*`Network`*) or hard-code it.*
   b. *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` *as 0.0.0.0. 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. *If using Network boot, the DHCP and/or BOOTP parameters must be set to provide the same information as the values coded in NVRAM. DHCP booting is discussed in Appendix E.*
   e. *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. *Set the* `Configuration File` *to* `flash.nsm`*.*
      (The ***flash.nsm*** *file must be created as outlined in Appendix A and placed in the CONFIGBASE directory.)*
   b. *Set the* `First` *entry under* `Configuration Directory` *to CONFIGDIR*`/`*.*
      See the earlier table for the values for CONFIGDIR on each platform. For example, in WindowsNT 4, enter `/netstation/prodbase/configs/`
      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` *or* `TFTP`
   d. *Press* `Enter` *to return to the Setup Utility.*

6. *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 ***flash.nsm*** *must be made to enable the Network Station to start an NFS server and export the local file system. (The* ***flash.nsm*** *file must be created as outlined in Appendix A and placed in the CONFIGBASE directory.)*

```
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 ***flash.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 }
}
```

Then, 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` *or* `Network` *depending on whether you want to use DHCP/BOOTP to assign the Network Station's address (*`Network`*) or hard-code it.*
   b. *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. 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. *If using Network boot, the DHCP and/or BOOTP parameters must be set to provide the same information as the values coded in NVRAM. DHCP booting is discussed in Appendix E.*
   e. *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. *Set the `Configuration File` to `peer.nsm`*
      *(The **peer.nsm** file must be created as outlined in Appendix A and placed in the CONFIGBASE directory.)*
   b. *Set the `First` entry under `Configuration Directory` to CONFIGDIR/.*
      See the earlier table for the values for CONFIGDIR on each platform. For example, in WindowsNT 4, enter `/netstation/prodbase/configs/`
      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` or `TFTP`*
   d. *Press `Enter` to return to the Setup Utility.*

5. *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.

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, try the following:

1. *Edit the file CONFIGBASE/flash.nsm on the server. Change the `4096` values in the `file-service-table` to `1024`.*

2. *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.*

3. *Power on or reboot the Network Station.*

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

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

5. 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
   >
   ```

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

7. > 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
set diag-buffer-size = 65535
```

in **flash.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 F.

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 in the ***flash.nsm*** *file below.*

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.

Creating Configuration Files

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

The ***flash.nsm*** *and* ***peer.nsm*** *files first read the standard configuration files from the server (****standard.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.*

The flash.nsm File

Here is a ***flash.nsm*** *configuration file for the system containing the flash card. PRODBASE is the root path for Network Station files on the server. You may need to change* `nfs` *to* `tftp` *depending on your environment. You may also need to adjust the read and write block sizes from the stated 4096 to 1024 if you are having trouble reading/writing files across gateways. You can increase this value to a maximum of 8192 if you do not have many hops between the Network Station and the server.*

```
#
# flash.nsm - place in the "CONFIGBASE" directory on the server
#
# Uncomment the two lines for the server type being used for authentication
# and replace 0.0.0.0 with the IP address of that system.
#
set file-service-table = {
#
# The following two lines are for WindowsNT.
#    {"/netstation/prodbase" nil 0.0.0.0 nfs "/netstation/prodbase" unix 3 10 4096 4
#    {"/netstation" nil 0.0.0.0 nfs "/netstation" unix 3 10 4096 4096}
#
# The following two lines are for AIX.
#    {"/netstation/prodbase" nil 0.0.0.0 nfs "/usr/netstation" unix 3 10 4096 4096}
#    {"/usr/netstation" nil 0.0.0.0 nfs "/usr/netstation" unix 3 10 4096 4096}
#
# The following two lines are for AS/400
#    {"/netstation/prodbase" nil 0.0.0.0 tftp "/QIBM/ProdData/NetworkStation" unix
#    {"/QIBM/ProdData" nil 0.0.0.0 tftp "/QIBM/ProdData" unix 3 10 4096 4096}
}
#
# Read the basic config information from the server
read standard.nsm
#
# Augment/override server values
set boot-desired-source = local
set boot-second-source = none
set boot-third-source = none
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"
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 diag-buffer-size = 65535
#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 peer.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 no edits/changes required.

```
#
# peer.nsm - place in the "CONFIGBASE" directory on the server
#
# Read the basic config information from the server
read flash.nsm
#
# Augment/override server values
set boot-desired-source = nfs
set boot-nfs-directory = /peerboot/
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/nc.xbm"
set pref-screensaver-bitmap-file = "/peerboot/SysDef/nc.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/
```

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 G provides several examples for common applications.

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 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
  C[1]
    msg
      X.cat         Xm.cat        Xt.cat
  EN_US
    MRI
      HelpLogin       N3_5Help        N3_5KyPH        NetHelpLogin
      Login           N3_5HlpV        N3_5LocP        NetWarningLogin
      N3_5ClrH        N3_5KeyH        N3_5MscP
      N3_5ColM        N3_5KeyM        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.

flash.nsm

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

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

peer.nsm

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

```
#
# Get the fonts from flash
set xserver-default-font-path = {
    {"/peerboot/fonts/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 ***flash.nsm*** *and* ***peer.nsm*** *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.i.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 PC. There is no change to the **flash.nsm** and **peer.nsm** 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
                                              seriald.nws
nls
  C[1]
    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
  C[1]
    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 CONFIGBASE directory and this file is only read when the NVRAM configuration file section on the netstation is blank. But, flash/peer booting requires that a value be placed in this field (i.e. `flash.nsm` or `peer.nsm`). 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 flash.nsm
read peer.nsm
```

This allowed the flash or peer files to be read, and since they call standard.nsm, the correct order was still kept. 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 **flash.nsm** and **peer.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.

Hummingbird Maestro NFS Client

*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 CONFIGBASE 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.*

Appendix E: Using Flash With DHCP

This section is being revised. Please do not use at this time!

This appendix contains information on using flash cards with DHCP. The scenario is for Microsoft DHCP on NT 4 but the concepts are applicable to other server platforms supporting 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://www.ibm.com/nc/pubs.

Steps:

Create New Directories

The DHCP parameters recognized by the Network Station do not include a way to set a Configuration File. Thus, when using DHCP, the Network Station always tries to find the default file, **standard.nsm***. Therefore, the* **flash.nsm** *and* **peer.nsm** *files must be moved to other directories and renamed* **standard.nsm***.*

Perform the following steps:

1. *Ensure that the line* `read CONFIGDIR/standard.nsm` *is in both* **flash.nsm** *and* **peer.nsm***.*
2. *Change the "read" statememts in CONFIGDIR/standard.nsm to have full pathnames, i.e. prefixed by CONFIGDIR/ as in* `read /usr/netstation/configs/control.nsm`*.*
3. *Create a directory on the server called CONFIGBASE/flash.*
4. *Copy* **flash.nsm** *to this directory and rename it* **standard.nsm***.*
5. *If doing peer boot, create a directory on the server called CONFIGBASE/peer.*
6. *Copy* **peer.nsm** *to this directory and rename it* **standard.nsm***.*

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

1. *Select* `Scope` *and then select DHCP Options => Scope*
2. *Add options 03, 66, 67, 211, 212, 213, 214*

3. *Set value for 03 = router address*
4. *Set value for 66 = IP address of NS with flash card*
5. *Set value for 67 = /peerboot/kernel.Z (kernel.63Z for S/1000)*
6. *Set value for 211 = NFS*
7. *Set value for 212 = IP Address of Configuration Server*
8. *Set value for 213 = CONFIGDIR/peer/*
9. *Set value for 214 = NFS*

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.*

Set DHCP Options for the Reserved Client

This will allow you to over ride the DHCP options that are specific for the Network Station with the flash card.

1. *Select Scope => Active Leases => Properties => Options*
2. *Add options 03, 66, 67, 211, 212, 213, 214*
3. *Set value for 03 = router address*
4. *Set value for 66 = 0.0.0.0*
5. *Set value for 67 = kernel.Z (kernel.63Z for S/1000 kernel)*
6. *Set value for 211 = LOCAL*
7. *Set value for 212 = IP Address of Configuration Server*
8. *Set value for 213 = CONFIGDIR/flash/*
9. *Set value for 214 = NFS*

Network Station with Flash Card

1. *From the Setup Utility screen select F3.*
2. *IP Address from Network*
3. *DHCP IP Addressing Order : 1*

Peer Boot Network Station

1. *From the Setup Utility screen select F3*
2. *IP Address from Network*
3. *DHCP IP Addressing Order : 1*

Appendix F: 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 G: 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

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/BOOTP 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. If using Network boot, the DHCP and/or BOOTP parameters must be set to provide the same information as the values coded in NVRAM. DHCP booting is discussed in Appendix E.
9. Depress `Enter` to return to the Setup Utility screen.
10. Choose `F4 = Set Boot Parameters.`
11. Leave the `Boot File` value blank.
12. Leave the `TFTP Boot Directory` and `NFS Boot Directory` entries blank.
13. Set the `TFTP Order` and `NFS Order` under `Boot Host Protocol` to `D`.
14. Set `LOCAL Order` to `1`.
15. Depress `Enter` to return to the Setup Utility screen.
16. Choose `F5 = Set Configuration Parameters.`
17. Set the `Configuration File` to `flash.nsm`.
18. Set the `First` entry under `Configuration Directory` to `/local/configs/`.
19. Set the `First` entry under `Configuration Host Protocol` to `Local`.
20. 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 **flash.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 **flash.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/BOOTP 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. *If using Network boot, the DHCP and/or BOOTP parameters must be set to provide the same information as the values coded in NVRAM. DHCP booting is discussed in Appendix E.*
7. *Depress `Enter` to return to the Setup Utility screen.*
8. *Choose `F4 = Set Boot Parameters`.*
9. *Leave the `Boot File` value blank.*
10. *Leave the `TFTP Boot Directory` entry blank.*
11. *Set the `NFS Boot Directory` to `/peerboot/`.*
12. *Set the `TFTP Order` and `LOCAL Order` under `Boot Host Protocol` to `D`.*
13. *Set `NFS Order` to `1`.*
14. *Depress `Enter` to return to the Setup Utility screen.*
15. *Choose `F5 = Set Configuration Parameters`.*
16. *Set the `Configuration File` to `peer.nsm`.*
17. *Set the `First` entry under `Configuration Directory` to `/peerboot/configs/`.*
18. *Set the `First` entry under `Configuration Host Protocol` to `NFS`.*
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 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 **flash.nsm** and **peer.nsm** 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 12 Mbyte card.

flash.nsm

*The **flash.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 **standard.nsm** and all of the configuration files it reads onto the flash card.*

```
#
# flash.nsm for emulator scenario
#
read standard.nsm
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 exec-command-menu = {
    { "5250" "ns5250 rocket" }
    { "3270" "ns3270 RALVM17.raleigh.ibm.com" }
    { "VT220" "nsterm -tn vt220 -host bissell"}
}
set exec-startup-commands = {
    { "mcuis" }
    { "mwm" }
    { "ns5250 rocket" }
    { "ns3270 RALVM17.raleigh.ibm.com" }
    { "nsterm -tn vt220 -host bissell" }
}
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/SysDef"}
    {"IMKEYMAPPATH" "/local/nls"}
    {"LANG" "EN_US" }
    {"LOCALENAME" "EN_US"}
    {"LOCPATH" "/local"}
    {"NLSPATH" "/local/nls/%L/msg/%N"}
    {"NSM_ADMIN_SYSDEFAULTS" "/local/SysDef"}
    {"NSM_PROD_SYSDEFAULTS" "/local/SysDef"}
    {"NSM_USER_PREFS" "/local/SysDef"}
    {"NSM_NS5250_PREF_VERSION" "R3M0"}
    {"PATH" "/local/mods"}
    {"XLOCALEDIR" "/local/X11/locale"}
    {"XMODIFIERS" ""}
}
#
```

```
# The following lines should reflect whatever initial bitmap
# was set in the configs files, but with the path changed to /local.
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/X11/fonts/pcf/i18n"}
    {"/local/fonts/pcf/misc"}
    {"built-ins"}
}
#
# The following X resources should start with the values defined in
# the original config files. Values for other items can then be added.
set xserver-initial-x-resources = "ncdconsole.disableReboot: false\n\
    ncdconsole.disableLoginMenu: false\n\
    ncdconsole.disableTerminalMenu: false\n\
    ncdconsole.disableSetupMenu: false\n\
    NCDterm.autoReconnect: true\n\
    NCDterm.defaultHost: bissell\n\
    NCDterm.menuBar: false\n\
    NS3270*KeyPad: disable_and_hide\n\
    NS3270*KeyRemap: disable_and_hide\n\
    NS5250*KeyPad: disable_and_hide\n\
    NS5250*Control: disable_and_hide\n\
    NS5250*Edit: disable_and_hide\n\
    NS5250*LocalPrint: keyboard_only_local_print\n\
    NS5250*ChangeIPAddress: disable\n\
    NS5250*Command: disable_and_hide\n\
    NS5250*MiscPref: disable_and_hide\n\
    NS5250*Option: disable_and_hide\n\
    NS5250*Help: disable_and_hide\n\
    NS5250*FontMenu: disable_no_resize_or_move\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
```

peer.nsm

*The **peer.nsm** file reads all of the other configuration files by reading **flash.nsm** and then overrides values.*

```
#
# peer.nsm for emulator scenario
#
read flash.nsm
set boot-desired-source = nfs
set boot-nfs-directory = /peerboot/
set config-unix-directory = /peerboot/configs/
set file-enable-nfs-server = false
set file-try-all-matches-on-open = true
set modules-directory = /peerboot/mods

set pref-environment = {
#   Environment variables
    {"BOOTPATH" "/peerboot"}
    {"HOME" ""}
    {"IMKEYMAPPATH" "/peerboot/nls"}
    {"LANG" "EN_US" }
    {"LOCALENAME" "EN_US"}
    {"LOCPATH" "/peerboot"}
    {"NLSPATH" "/peerboot/nls/%L/msg/%N"}
    {"NSM_ADMIN_SYSDEFAULTS" "/peerboot/SysDef"}
    {"NSM_PROD_SYSDEFAULTS" "/peerboot/SysDef"}
    {"NSM_USER_PREFS" "/peerboot/SysDef"}
    {"NSM_NS5250_PREF_VERSION" "R3M0"}
    {"PATH" "/peerboot/mods"}
    {"XLOCALEDIR" "/peerboot/X11/locale"}
    {"XMODIFIERS" ""}
}
#
# The following lines should reflect whatever initial bitmap
# was set in the configs files, but with the path changed to /peerboot.
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/X11/fonts/pcf/i18n"}
    {"/peerboot/fonts/pcf/misc"}
    {"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[2]
    K
      Default.101
  NS5250[3]
    K
      Default.101
X11
  app-defaults
    Mcuis        Mwm          system.mwmrc
  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
  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.nsm hosts.nsm    standard.nsm
  defaults.dft flash.nsm    peer.nsm     required.nsm
fonts
  pcf
    misc
      12x24.pcf.Z    8x16rk.pcf.Z   clR5x10.pcf.Z  clR8x12.pcf.Z  nil2.pcf.Z
      12x24rk.pcf.Z  clB6x10.pcf.Z  clR5x6.pcf.Z   clR8x13.pcf.Z  olcursor.pcf.Z
      5x8.pcf.Z      clB6x12.pcf.Z  clR5x8.pcf.Z   clR8x14.pcf.Z  olgl10.pcf.Z
      6x12.pcf.Z     clB8x10.pcf.Z  clR6x10.pcf.Z  clR8x16.pcf.Z  olgl12.pcf.Z
      6x13B.pcf.Z    clB8x12.pcf.Z  clR6x12.pcf.Z  clR8x8.pcf.Z   olgl14.pcf.Z
      6x9.pcf.Z      clB8x13.pcf.Z  clR6x13.pcf.Z  clR9x15.pcf.Z  olgl16.pcf.Z
      7x13.pcf.Z     clB8x14.pcf.Z  clR6x6.pcf.Z   deccurs.pcf.Z  olgl19.pcf.Z
      7x13B.pcf.Z    clB8x16.pcf.Z  clR6x8.pcf.Z   decsess.pcf.Z  olgl20.pcf.Z
      7x14.pcf.Z     clB8x8.pcf.Z   clR7x10.pcf.Z  fg-22.pcf.Z    olgl24.pcf.Z
      7x14B.pcf.Z    clB9x15.pcf.Z  clR7x12.pcf.Z  fonts.alias
      7x14rk.pcf.Z   clI6x12.pcf.Z  clR7x14.pcf.Z  fonts.dir
      8x13B.pcf.Z    clI8x8.pcf.Z   clR7x8.pcf.Z   jiskan24.pcf.Z
      8x16.pcf.Z     clR4x6.pcf.Z   clR8x10.pcf.Z  k14.pcf.Z
keyboards
  AB83useng
mods
  colormap.nws  keymap52.nws  lpd.nws       mwm.nws       nsterm.nws
                libconf.nws   lprd.nws      nfsd.nws      sbcs_im.nws
  export.63a    libmlc.nws    mcuis.nws     ns3270.nws    seriald.nws
  export.nws    libprapi.nws  miscpr32.nws  ns5250.nws    setup.nws
  helpview.nws  libprxapi.nws miscpref.nws  ns5250xx.nws  term.nws
nls
  EN_US.im       EN_US.imkeymap sbcs.imcompose
  C[1]
    msg
      X.cat         Xm.cat        Xt.cat
  EN_US
    MRI
      HelpLogin       N3_5ColM        N3_5KeyH        N3_5LocP        NetHelpLogin
      Login           N3_5Help        N3_5KeyM        N3_5MscP        NetWarningLogi
      N3_5ClrH        N3_5HlpV        N3_5KyPH        N3_5Resc
    msg
      X.cat         Xt.cat        libprxapi.cat mwm.cat
      Xm.cat        common.cat    mcuis.cat     term.cat
  codepage
    IBM1208 IBM37   IBM819
```

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***

**** End of Document ****