

Novell JVM^{*} for NetWare[®]

1.4.2

March 2, 2005

INSTALLATION AND ADMINISTRATION
GUIDE

www.novell.com



Novell[®]

Legal Notices

Novell, Inc. makes no representations or warranties with respect to the contents or use of this documentation, and specifically disclaims any express or implied warranties of merchantability or fitness for any particular purpose. Further, Novell, Inc. reserves the right to revise this publication and to make changes to its content, at any time, without obligation to notify any person or entity of such revisions or changes.

Further, Novell, Inc. makes no representations or warranties with respect to any software, and specifically disclaims any express or implied warranties of merchantability or fitness for any particular purpose. Further, Novell, Inc. reserves the right to make changes to any and all parts of Novell software, at any time, without any obligation to notify any person or entity of such changes.

You may not use, export, or re-export this product in violation of any applicable laws or regulations including, without limitation, U.S. export regulations or the laws of the country in which you reside.

Copyright © 2005 Novell, Inc. All rights reserved. No part of this publication may be reproduced, photocopied, stored on a retrieval system, or transmitted without the express written consent of the publisher.

Novell, Inc. has intellectual property rights relating to technology embodied in the product that is described in this document. In particular, and without limitation, these intellectual property rights may include one or more of the U.S. patents listed at <http://www.novell.com/company/legal/patents/> and one or more additional patents or pending patent applications in the U.S. and in other countries.

Novell, Inc.
404 Wyman Street, Suite 500
Waltham, MA 02451
U.S.A.

www.novell.com

Novell JVM 1.4.2 for NetWare Installation and Administration Guide

March 2, 2005

Online Documentation: To access the online documentation for this and other Novell developer products, and to get updates, see developer.novell.com/ndk. To access online documentation for Novell products, see www.novell.com/documentation.

Novell Trademarks

NetWare is a registered trademark of Novell, Inc. in the United States and other countries.

NetWare Loadable Module and NLM are trademarks of Novell, Inc.

Novell is a registered trademark of Novell, Inc. in the United States and other countries.

Novell Client is a trademark of Novell, Inc. in the United States and other countries.

Third-Party Materials

All third-party trademarks are the property of their respective owners.

Contents

About This Guide	7
1 Novell JVM For NetWare Product Overview	9
2 System Information and Requirements	11
Important Java Files	11
Supported Video Controllers	11
Hardware Requirements	12
3 Installing NJVM 1.4.2	13
Installation Prerequisites	13
Installing NJVM 1.4.2	13
Loading and Unloading NJVM	15
Loading NJVM	15
Unloading NJVM	15
4 NJVM Administration	17
Font.properties File	17
NetWare GUI	17
Starting the NetWare GUI.	18
Viewing Multiple Server Screens	18
Toggling between the NetWare GUI and the System Console	18
Activating the Taskbar Menu	18
Configuring the Taskbar Menu	18
Configuring X Server Using the NetWare GUI Environment Utility	20
Configuring Your System for an Accelerated Video Driver	20
Selecting a Background Pattern	21
Using the NetWare GUI without a Mouse.	21
Using Shortcut Keystrokes for Java Text Areas and Text Fields	21
Closing the NetWare GUI	22
Applets and Applications	23
Running an Applet	23
Running an Application	23
Binding an Application to a Specific Processor	23
Listing the Current Java Processes	24
Shutting Down a Running Java Process	24
Properties File	24
Environment Variables	25
Viewing the Current Values of Environment Variables	25
Setting the DISPLAY Environment Variable	25
Using the Remote Display Feature	26
Setting the CWD Environment Variable.	27
Setting the CLASSPATH Environment Variable	27
Performance Tuning	27
Java Configuration Files	28
Java.cfg Configuration File	28

Autoexec.ncf	28
Internationalization	28
Russian Keyboard Activation	28
AWT Limitations	29
Adding Fonts with Font.properties.allfonts.	29
JVM Client and Server Versions.	29
Client VM	29
Server VM	29
Switching JVM Versions	30
5 Java Application Development	31
Using Java Native Interface (JNI) on NetWare	31
Required Tools.	31
File Descriptions	32
Building a JNI Example	32
Installing a JNI Example	32
Running a JNI Example	32
Unloading JNI	33
JNI Notes	33
Using WATCOM 11.0 Compiler Flags for Native Method NLM Programs.	34
Source-Level Debugging	34
Required VM Options	34
Obtaining the Fully Qualified Domain Name of the Local Host	35
6 Troubleshooting	37
Java GUI Applications Fail to Load	37
The NetWare GUI Fails to Load.	37
Mouse Does Not Respond	37
Applications Using Java.net Do Not Work Properly	38
Security Alert Dialog Box Appears When Accessing Secure Web Sites	38

About This Guide

This guide introduces Novell® JVM 1.4.2 for NetWare® and explains how to install and configure it on your system. In this document, Novell JVM for NetWare is referred to as NJVM.

- ♦ Chapter 1, “Novell JVM For NetWare Product Overview,” on page 9
- ♦ Chapter 2, “System Information and Requirements,” on page 11
- ♦ Chapter 3, “Installing NJVM 1.4.2,” on page 13
- ♦ Chapter 4, “NJVM Administration,” on page 17
- ♦ Chapter 5, “Java Application Development,” on page 31
- ♦ Chapter 6, “Troubleshooting,” on page 37

Documentation Updates

For the most recent documentation on using NJVM, see the [NJVM online documentation \(http://www.novell.com/documentation/lg/jvm142/index.html\)](http://www.novell.com/documentation/lg/jvm142/index.html).

Documentation Conventions

A trademark symbol (®, ™, etc.) denotes a Novell trademark. An asterisk (*) denotes a third-party trademark.

When a single pathname can be written with a backslash for some platforms or a forward slash for other platforms, the pathname is presented with a backslash. Users of platforms that require a forward slash, such as UNIX*, should use forward slashes as required by your software.

1

Novell JVM For NetWare Product Overview

Novell® JVM for NetWare® (NJVM) runs Java* applications, servlets, and applets in a NetWare environment. The NetWare Graphical User Interface (GUI) lets you run graphical applications. NJVM also lets you export the display of X Client programs to any machine running an X server.

NJVM contains the following features:

- ♦ Full featured JVM based on Sun's* J2SE SDK 1.4.2. For more information, see [J2SE 1.4.2 \(http://java.sun.com/j2se/1.4.2/index.jsp\)](http://java.sun.com/j2se/1.4.2/index.jsp).
- ♦ GUI functionality supported by an X-Windows environment. For more information, see [“NetWare GUI” on page 17](#).
- ♦ The following utilities:
 - ♦ File editor
 - ♦ File browser
 - ♦ NRM browser for server administration
 - ♦ Server Console utility
- ♦ Select applications have been written to the X toolkits for usability improvements
- ♦ Multiserver install capabilities. For more information, see [“Installing NJVM 1.4.2” on page 13](#).

2

System Information and Requirements

This section provides information and requirements for installing and running NJVM.

- ♦ “Important Java Files” on page 11
- ♦ “Supported Video Controllers” on page 11
- ♦ “Hardware Requirements” on page 12

Important Java Files

Directory	File	Description
sys:\java\lib	charsets.jar	Character-conversion classes.
	font.properties.*	Font alias files.
	rt.jar	Core Java classes.
	tools.jar	Support classes for SDK tools and utilities.
sys:\java\lib\ext	localedata.jar	Locale data for java.text and java.util.
sys:\java\lib\security	java.policy	Security policy.
	java.security	Security properties.
sys:\java\bin	java.nlm	Executable files for launching Java applications, tools, and utilities.
sys:\java\nwgfx\bin	startx.ncf	Starts the NetWare® GUI.
	stopx.nlm	Closes the NetWare GUI.
	vesa_rsp.ncf	Script for resetting the NetWare GUI to the default configuration.
sys:\java\nwgfx\lib\x11\fonts		Fonts for displaying characters in a GUI window.
sys:\java\nwgfx\pixmaps		Images used for the NetWare GUI background.

Supported Video Controllers

Novell® supports the VESA BIOS extensions. The default X server on NetWare, Xvesa, provides graphics support for video controllers compliant with VESA 1.2 and later. The resolution for VESA video controllers is controller dependent. NetWare also includes Xi Graphics* Accelerated-X* as an alternative X server that supports over 500 specific video cards.

NJVM currently supports the following video controllers for up to 1600x1200 and True Color:

- ♦ VESA 1.x, 2.x, 3.x.

VESA support covers the majority of recently manufactured video controllers.

- ♦ Xi Graphics video controllers.

For instructions on accessing the video card tab in the taskbar menu for viewing a complete list of supported controllers, see “[Configuring X Server Using the NetWare GUI Environment Utility](#)” on page 20.

NOTE: Using an accelerated video driver can limit functionality. For instructions on configuring your monitor for an accelerated video driver, see “[Configuring Your System for an Accelerated Video Driver](#)” on page 20.

Hardware Requirements

The hardware requirements are determined by your version of NetWare.

Component	Requirement
CPU	550 MHz minimum Intel* Pentium* or compatible.
Memory	512 MB minimum. Refer to NetWare 6.0 memory requirements (http://www.novell.com/documentation/lg/nw6p/index.html?page=/documentation/lg/nw6p/setupenu/data/hz8pck9v.html) or NetWare 6.5 memory requirements (http://www.novell.com/documentation/lg/nw6p/index.html?page=/documentation/lg/nw6p/setupenu/data/hz8pck9v.html) .
Hard Disk Space	200 MB available
Mouse	PS/2* or USB. NOTE: Some applications can function without a mouse, but this is not recommended.

3

Installing NJVM 1.4.2

You can install NJVM on one or multiple servers. This section contains the following procedures for installing NJVM:

- ♦ “Installation Prerequisites” on page 13
- ♦ “Installing NJVM 1.4.2” on page 13
- ♦ “Loading and Unloading NJVM” on page 15

Installation Prerequisites

- ☐ NetWare 6 or later running with the latest Support Pack
- ☐ NJVM self-extracting executable file
- ☐ Client workstation running Windows* 2000/XP with Novell Client™ with 200 MB of free disk space

Installing NJVM 1.4.2

IMPORTANT: If you are upgrading to NJVM 1.4.2, you need to close the NetWare GUI and exit NJVM. You can do these tasks manually or have the upgrade process do them automatically. You select the options you want in [Step 6](#) of the installation process.

We recommend that you complete the installation process during off-peak production times.

To install NJVM:

- 1** (Optional) Create a text file with a list of the servers that you want to connect to when installing NJVM.

This file must have only one server per line. For example,

```
MyServer1  
MyServer2.mydomain.com  
10.0.0.3  
MyServer4
```

We recommend using IP addresses when connecting to servers over a WAN.

- 2** Run the NJVM self-extracting executable file on the client workstation.
This expands the archive and launches the InstallShield installation script.
- 3** On the Welcome page, click Next.
- 4** Accept the License Agreement.
This opens the Installation Options page.
- 5** On the Installation Options page, select one of the following options:

- ♦ **Typical installation.** Use this installation type to do a normal install.
- ♦ **Modem or slow WAN installation.** Use this installation type if you have a slow network connection and you have an alternate method of distributing the java.zip file to all the servers you want to install to. To get a copy of java.zip, see the Unpack java.zip option below.
- ♦ **Unpack java.zip.** Use this option to unpack java.zip to a directory of your choice. After you unpack java.zip, the installation process ends. Copy java.zip to SYS:\ on each target server prior to running the Modem or Slow WAN installation. .
- ♦ **View Readme.** Use this option to open the Readme for information about JVM updates and new features.

6 On the Select Options screen, select one, both, or neither of the following options:

- ♦ **Connect to the servers listed in a text file.** If you select this option, the installation process connects to each server listed in the text file you created in **Step 1**. If you do not select this option, you must manually connect to each target server.

If you are not connected to the tree that contains the servers in your list, a login screen appears. You need to log in only once for each tree represented in your server file.

Connecting to all servers listed in the text file does not mean that NJVM will install to all the servers. In **Step 8** of this installation, you select the target servers you want to install on.

- ♦ **Unload Java from Target Servers.** If you select this option, the installation process unloads java.nlm and closes the NetWare GUI on the target servers. The installation process then unloads uimon.nlm, ps2.nlm, aiops2.nlm, xidev.nlm, and xlib.nlm. If you do not select this option, you must do all these tasks manually on each target server.

7 On the Installation Requirements page, review the installation prerequisites, then click Next.

8 On the Destination Servers for Installation page, select the servers you want to install NJVM on, then click Next.

If java.nlm is still loaded, an alert screen appears.

9 (Conditional) If an alert screen appears, close the NetWare GUI, unload the specified files from the specified servers, then click Next.

NOTE: If you attempt to install an earlier version of NJVM than is currently on your servers, a Not An Upgrade warning appears. The options Yes and No apply to the server specified in the warning. The options Yes to All and No to All apply to all servers.

The installation displays a progress bar.

The installation process copies the java.zip file to the destination servers and unzips the file on each server.

When the installation is complete, a results page displays. This page lists the installation errors, servers that need to be rebooted, and servers that installed successfully. The information can also be found in c:\temp\InstallShieldLogs\njvm_server_install.log. For more installation information, refer to the sequential log file (njvm_log.txt) in the same directory.

10 Reboot your servers if necessary.

NOTE: If any problems occur with the GUI after you upgrade to NJVM 1.4.2, enter stopx at the system console and run VESA_RSP. This resets the GUI to the default settings.

The [Readme \(http://www.novell.com/documentation/lg/jvm142/index.html\)](http://www.novell.com/documentation/lg/jvm142/index.html) document contains any last-minute information about installing NJVM that is not in this documentation.

Loading and Unloading NJVM

This section explains how to load and unload NJVM.

- ♦ [“Loading NJVM” on page 15](#)
- ♦ [“Unloading NJVM” on page 15](#)

Loading NJVM

To load NJVM (java.nlm), enter **load java** at the system console.

This loads NJVM into memory and allows the command interpreter to recognize when a Java application or applet is launched. For information about different loading options, see [“JVM Client and Server Versions” on page 29](#).

Unloading NJVM

You can unload NJVM (java.nlm) from memory by entering the following command at the system console:

```
java -exit
```

This calls any registered shutdown hooks, closes all of your Java applications, and unloads Java.

4

NJVM Administration

After you install NJVM, you can maintain the following features:

- ♦ “Font.properties File” on page 17
- ♦ “NetWare GUI” on page 17
- ♦ “Applets and Applications” on page 23
- ♦ “Properties File” on page 24
- ♦ “Environment Variables” on page 25
- ♦ “Performance Tuning” on page 27
- ♦ “Java Configuration Files” on page 28
- ♦ “Internationalization” on page 28
- ♦ “JVM Client and Server Versions” on page 29

Font.properties File

The `java\lib\font.properties` file for NJVM controls the Java font to native font mapping. The `font.properties` file is nearly identical to the Solaris* implementation. For a description and explanation of the `font.properties` files, see the `font.properties` documentation on the [Sun Web Site](http://java.sun.com/j2se/1.4.2/docs/guide/intl/fontprop.html) (<http://java.sun.com/j2se/1.4.2/docs/guide/intl/fontprop.html>).

For country-specific fonts, see “Adding Fonts with `Font.properties.allfonts`” on page 29.

NetWare GUI

This section contains the following NetWare GUI procedures:

- ♦ “Starting the NetWare GUI” on page 18
- ♦ “Viewing Multiple Server Screens” on page 18
- ♦ “Toggling between the NetWare GUI and the System Console” on page 18
- ♦ “Activating the Taskbar Menu” on page 18
- ♦ “Configuring the Taskbar Menu” on page 18
- ♦ “Configuring X Server Using the NetWare GUI Environment Utility” on page 20
- ♦ “Configuring Your System for an Accelerated Video Driver” on page 20
- ♦ “Selecting a Background Pattern” on page 21
- ♦ “Using the NetWare GUI without a Mouse” on page 21
- ♦ “Using Shortcut Keystrokes for Java Text Areas and Text Fields” on page 21

- ♦ “Closing the NetWare GUI” on page 22

Starting the NetWare GUI

To start the NetWare GUI, enter **startx** at the system console.

When the Novell button appears on the taskbar menu in the lower left corner, the startup process is complete.

Viewing Multiple Server Screens

The NetWare GUI lets you view multiple server screens simultaneously. For example, you can view the Logger screen, Edit screen, and the Console screen from three different windows.

To view multiple server screens, you use the NetWare GUI’s Server Console utility. This utility is C-based and does not require Java to be loaded. You can load multiple instances of the Server Console utility.

To start the Server Console utility, click the Server Console icon on the taskbar menu, or click Novell, click Utilities, then click Server Console.

NOTE: Some screens have limited functionality and update only when they are the active screen. You need to view these screens from within a system console that follows the active screen.

The first instance of the Server Console utility you launch follows the active screen. This means that it immediately reflects changes made on the active screen. For all other instances, following the active screen is not the default setting; you can manually select this option by clicking View, then clicking Follow Active Screen.

All instances following the active screen have an asterisk in front of the screen name.

Toggling between the NetWare GUI and the System Console

- ♦ To toggle to the next screen, press Alt+Esc.
- ♦ To toggle to the NetWare screen selection list, press Ctrl+Esc.

Activating the Taskbar Menu

You can use one of three methods to activate the taskbar menu:

1. Click Novell in the lower left corner of the screen.
2. Press the Windows meta key.
3. Right-click on the NetWare GUI background.

Configuring the Taskbar Menu

The taskbar menu consists of the following items:

- ♦ “Menus” on page 19
- ♦ “Commands” on page 20

Menus

Menus contain other menus and commands. When you open a menu, it displays other menus and command options.

When you create menus, you can select either a default name or an international name (optional). If you specify an international name, it displays in the taskbar menu; otherwise, the default name displays. International names are stored in files with the .properties extension. The following explains the name options:

- ♦ **Default name** displays when you do not specify a name
- ♦ **International name** displays when you specify an international name. The text for the international name is specified in the Properties file.

This section contains the following procedures:

- ♦ “Creating a Menu” on page 19
- ♦ “Modifying a Menu” on page 19
- ♦ “Moving a Menu” on page 19
- ♦ “Deleting a Menu” on page 20

Creating a Menu

To create a menu:

- 1** Click Novell in the NetWare GUI, select Settings, then select Menu Editor.
- 2** Right-click the location where you want to install the menu item.
- 3** Click either New Command or New Menu.
- 4** Type the information in the Edit panel.
- 5** On the taskbar menu, click File, then click Save.

Modifying a Menu

To modify a menu:

- 1** Click Novell in the NetWare GUI, select Settings, then select Menu Editor.
- 2** Click the menu you want to change.
- 3** Modify the command information in the Edit panel.
- 4** On the taskbar menu, click File, then click Save.

Moving a Menu

To move a menu:

- 1** Click Novell in the NetWare GUI, select Settings, then select Menu Editor.
- 2** Right-click the menu you want to move.
- 3** Click either Move Up or Move Down.
- 4** On the taskbar menu, click File, then click Save.

If a target menu is open, the selected menu moves into that menu; otherwise, the selected menu moves past the target menu.

Deleting a Menu

To delete a menu:

- 1 Click Novell in the NetWare GUI, select Settings, then select Menu Editor.
- 2 Right-click the menu you want to remove.
- 3 Select Remove.
- 4 On the taskbar menu, click File, then click Save.

Commands

Commands are items in a menu that launch applications. When you add commands, you can select default name, international name, and execute. The following explains each option:

- ♦ **Default name** displays when you do not specify a name
 - ♦ **International name** displays when you specify an international name. The text for the international name is specified in the Properties file.
 - ♦ **Execute** executes the commands you create.
- IMPORTANT:** We recommend that you specify a .ncf file that contains actual commands to execute.

Configuring X Server Using the NetWare GUI Environment Utility

You use the NetWare GUI environment utility to configure VESA Super VGA and Accelerated-X Servers on NetWare. This utility lets you select the video board, keyboard, mouse, and monitor from a set of predefined hardware devices supported by the NetWare GUI environment utility.

To configure X server, click Novell in the NetWare GUI, select Settings, then select GUI Environment.

For documentation on using the NetWare GUI Environment, click the Help button to open the online help.

Configuring Your System for an Accelerated Video Driver

To achieve the best video quality when you configure your accelerated video driver, configure the monitor and select the highest refresh rate possible.

NOTE: When selecting a video driver, we recommend using the standard Xvesa driver.

To configure your system for an accelerated video driver:

- 1 In the NetWare GUI, click Novell, select Settings, then select GUI Environment.
- 2 Select your video board from the Video Board List.

This activates the Monitor tab. The default settings are Super VGA with 1024 x 768 resolution and the Monitor tab deactivated.
- 3 Select your monitor from the list of monitors.

This enables all possible refresh rates for the selected monitor.
- 4 Select the highest available refresh rate, then set the video board to the desired resolution.
- 5 Select Test, then click OK.
- 6 After the test pattern displays, click OK, click Yes, then click Yes again.

Selecting a Background Pattern

To select a background pattern:

- 1 In the NetWare GUI, click Novell, select Tools, then select Backgrounds.

All files in the sys:\java\nwgfx\pixmaps directory display. The supported graphics formats are XPM, JPEG, GIF, and BMP. If you have background patterns (in the supported formats) that you want to use, place them in the sys:\java\nwgfx\pixmaps directory.

- 2 Select the desired background pattern, then click Test.
- 3 Click OK.

Using the NetWare GUI without a Mouse

If you start the NetWare GUI without a mouse attached to a PS/2 or USB port, the NetWare GUI will start in mouseless mode.

When the NetWare GUI is operating in mouseless mode, use the keypad keys for mouse movement and button clicks.

IMPORTANT: NUMLOCK must be activated to enable keypad mouse movements.

Keypad Key	Function
Arrows	Move the mouse pointer.
Shift+arrows	Accelerate pointer movement.
5	Behaves like the default pointer button.
0	Locks the default pointer button down (for easy dragging).
. (decimal)	Unlocks the default pointer button (releases a drag).
+ (plus)	Double-clicks the default pointer button.
Alt+Tab	Switches to the next program.
Alt+Shift+Tab	Switches to the previous program.

Using Shortcut Keystrokes for Java Text Areas and Text Fields

The following table shows the keystroke actions for Java text areas and text fields as implemented for NJVM:

Action	Text Area	Text Field
Navigate out forward	Ctrl+Tab	Tab
Navigate out backward	Ctrl+Shift+Tab	Shift+Tab
Move up/down one line	Up-arrow, Down-arrow	N/A
Move to previous/next character	Left-arrow, Right-arrow	Left, Right
Move to previous/next word	Ctrl+Left-arrow, Ctrl+Right-arrow	Ctrl+Left, Ctrl+Right

Action	Text Area	Text Field
Move to start/end of line	Home, End	Home/End
Move to start/end of text area	Ctrl+Home, Ctrl+End	N/A
Move up/down a page	PgUp, PgDn	N/A
Select all	Ctrl+/	Ctrl+/
Deselect all	Ctrl+\	Ctrl+\
Extend selection up/down	Shift+Up-arrow, Shift+Down-arrow	N/A
Extend selection left/right	Shift+Left-arrow, Shift+Right-arrow	Shift+Left, Shift+Right
Extend selection to start/end of line	Shift+Home, Shift+End	Shift+Home, Shift+End
Extend selection to previous/next word	Ctrl+Shift+Left-arrow, Ctrl+Shift+Right-arrow	Ctrl+Shift+Left, Ctrl+Shift+Right
Extend selection to start/end of text area	Ctrl+Shift+Home, Ctrl+Shift+End	N/A
Extend up a page	Shift+PgUp	N/A
Extend down a page	Shift+PgDn	N/A
Copy selection	Ctrl+C	Ctrl+C
Cut selection	Ctrl+X	Ctrl+X
Paste	Ctrl+V	Ctrl+V
Delete next character	Delete	Delete
Delete previous character	Backspace	Backspace
Insert tab	Tab	N/A
Toggle Insert/Replace mode	Insert	Insert
Insert line break	Enter	N/A
Submit entry	N/A	Enter

Closing the NetWare GUI

There are three ways to shut down the NetWare GUI:

1. In the NetWare GUI, click Novell, select Close GUI, then click Yes.
2. Press Ctrl+Alt+Backspace.
3. Run **stopx** from the system console.

Applets and Applications

This section contains the following processes:

- ♦ “Running an Applet” on page 23
- ♦ “Running an Application” on page 23
- ♦ “Binding an Application to a Specific Processor” on page 23
- ♦ “Listing the Current Java Processes” on page 24
- ♦ “Shutting Down a Running Java Process” on page 24

Running an Applet

- ♦ To run an applet, enter
`appletviewer html_filename or URL`
- ♦ To run an applet sending any console output to a new screen, enter
`applet -J-ns html_filename or URL`

For example, suppose an applet called MYAPP and its corresponding HTML file are found in `sys:\myapps\`. To run this applet, you would enter `applet sys:\myapps\myapp.html`.

Running an Application

- ♦ To run an application, enter
`java application_name`
- ♦ To run the application in a new screen, enter
`java -ns application_name`

NJVM assigns a screen to the application, then executes the application. If the application requires a graphical interface, NJVM loads the NetWare GUI and displays the application.

IMPORTANT: If the application requires console keyboard input, the `-ns` option must be used.

While the application is running, you can use the Server Console utility to view any output on the console screens. For instructions on using the Server Console utility, see “[Viewing Multiple Server Screens](#)” on page 18.

All GUI applications appear in the same GUI screen.

Binding an Application to a Specific Processor

To force an application to run on a specific processor, use the `-mp` flag. The processor number should follow `-mp`. For example, to assign a JVM specifically to processor 3, enter the following at the system console:

```
java -mp3 application_name
```

To assign a JVM to processor 0, enter the following at the system console:

```
java -mp0 application_name
```

NOTE: Using the `-mp0` flag is equivalent to disabling multiprocessor support. This flag should be used only with applications that are considered mp-unsafe.

Listing the Current Java Processes

To display a list of currently running processes and their process IDs, enter the following at the system console:

```
java -show
```

Shutting Down a Running Java Process

To shut down a Java process:

1 Get a list of the running Java processes by entering **java -show** at the system console.

2 At the system console, enter

```
java -shutdownprocess_ID
```

For example:

```
java -shutdown610
```

610 is the ID of the process to be shut down.

3 To shut down all running processes, enter the following at the system console:

```
java -shutdownall
```

NOTE: When you use -shutdown, Java calls the application's registered shutdown hooks. This provides a more graceful shutdown than the Java -kill command, which closes the application without calling the application's shutdown hooks. It is the application's responsibility to register its shutdown hooks.

Properties File

A Properties file is a text file that contains key=value entries. For example, PROGRAMS=programs or UTILITIES=utilities.

The key on the left should contain only ASCII characters. The value on the right can contain characters or specifiers for Unicode* encoding. For example, you can specify the value Run using any of the following:

- ♦ RUNITEM=Run
- ♦ RUNITEM=\u0052\u0075\u006e
- ♦ RUNITEM=R\u0075\u006e

The naming convention is to specify a file with a base name and an extension of .properties. For example, *MyNames.properties*

To use a properties file, you must first place it in the sys:\java\lib\taskbar directory. Then, in the Optional Internationalized Name section of the Menu Editor, click Browse to select your properties file. When you select this file, the items in the file display in a list of names. You choose the name you want to use from this list.

IMPORTANT: Do not add your own name entries to the NWMenu.properties file. This file is overwritten when you install a newer version of NJVM.

The values you retrieve from a properties file can correspond to the language of the server. If you provide the product for an international audience, you might want to ship properties files for each of the locales supported by the server. To do this, create a file whose name will be used as the base name. Then create files with filenames that contain the base name plus a language (and optionally

locale) specifier that corresponds to the language of the values in the properties file. This follows the naming convention used by Java properties files.

For example, the following are some of the file names in the `sys:\java\lib\taskbar`:

- ♦ **NWMenu.properties** contains values for English text.
- ♦ **NWMenu_de.properties** contains values for German text.
- ♦ **NWMenu_fr.properties** contains values for French text.
- ♦ **NWMenu_ru.properties** contains values for Russian text.

When you specify the Properties file, you specify the default properties file such as `NWMenu.properties`. The actual properties file that is used to retrieve the names depends on the language and locale of the server. For example, if the `NWMenu.properties` file is specified, the server language is set to use German, and there is a `NWMenu_de.properties` file, then the `NWMenu_de.properties` file is searched first for the entry. If the entry is not found in this file, then the `NWMenu.properties` file is searched. If the entry is not found in either file, then the default name is used.

Environment Variables

This section contains the following processes:

- ♦ [“Viewing the Current Values of Environment Variables” on page 25](#)
- ♦ [“Setting the DISPLAY Environment Variable” on page 25](#)
- ♦ [“Using the Remote Display Feature” on page 26](#)
- ♦ [“Setting the CWD Environment Variable” on page 27](#)
- ♦ [“Setting the CLASSPATH Environment Variable” on page 27](#)

Viewing the Current Values of Environment Variables

To view the current values of environment variables:

- 1** Start NJVM by entering **load java** at the system console.
- 2** Enter **envset**

When you unload NJVM (`java.nlm`), the process deletes the Environment variables that were set using `ENVSET`. You can add Environment variables in the `sys:\etc\java.cfg` file, which is read when Java loads.

Setting the DISPLAY Environment Variable

It is helpful to understand the UNIX X Window system before setting up the `DISPLAY` environment variable.

To set the `DISPLAY` environment variable:

- 1** Start NJVM by entering **load java** at the system console.
- 2** At the system console, enter the following:
envset DISPLAY=IP_address:display

The default value for *display* is 127.0.0.1:0.0, which is the address of the loopback interface and the default value for the display and screen you want to use.

Using the Remote Display Feature

The Remote Variable Display feature lets you export the display of any X client programs to any machine on the network running an X server. Because the NetWare GUI is X-based, all Java GUI applications and applets are considered X clients and are able to use the remote display feature.

Allowing Remote Access to a Server

You must configure your destination server to allow remote access. There are two methods to authorize access of a remote X client to a local X server:

1. Enter **-ac** at the X server command line.

This turns off access control and lets any other client access the server. Be cautious with this method. It lets anyone who can connect to your X server run programs on your display.

2. Run **xhost** on the local system.

This lets you add hosts to the list of client workstations that you allow to access your server. It also lets you delete workstations from this list. For more information on determining the best method of allowing remote access, see your X server user manual.

Displaying a Java Applet or Application on a Local X server

To display a Java applet or Application:

- 1** Allow the remote NetWare system to access the local X server.

For instructions, see [“Allowing Remote Access to a Server” on page 26](#).

- 2** Start the X server at the local system.

- 3** At the local system, start RCONJ and log in to the NetWare server.

- 4** Do the following in the RCONJ window:

4a Start NJVM by entering **load java** at the system console.

4b Enter the following:

```
envset display=ip_address_of_local_X_server:0
```

4c Start the Java applet or application.

- 5** (Optional) To make the application always open to the NetWare GUI screen, enter the following at the command line of the NetWare server:

```
envset display=ip_address_of_local_X_server:0.0
```

All Java GUI applications now display on the remote client workstation on the NetWare GUI screen. If you do not want the application to always open to the NetWare GUI screen, remove the “.0” from the end of the IP address.

NOTE: The local system console screen does not automatically switch to the NetWare GUI screen when a GUI application is opened from a remote server or from foreign IP addresses. You must manually switch to the NetWare GUI on your local server to see the display being exported from the remote server.

Setting the CWD Environment Variable

You can use the Current Working Directory (CWD) variable to run an application in a specific directory even if it is not part of a package. The CWD variable is set to the root when Java is loaded; however, you can set it to any desired directory.

To set the CWD environment variable:

- 1 Start NJVM by entering **load java** at the system console.
- 2 To set the CWD for all applications, enter the following at the system console:

```
envset CWD=pathname
```

For example:

```
envset CWD=sys:\myjava\myapp
```

The preferred method is to set the CWD for the current application by using the -env option.
For example:

```
java -envCWD=sys:\myfiles myapp
```

Setting the CLASSPATH Environment Variable

A default CLASSPATH variable is set when Java loads. If the CLASSPATH variable is set incorrectly, your Java applications might not run. A lengthy CLASSPATH variable might decrease the performance of your Java applications.

To set the CLASSPATH environment variable:

- 1 Start NJVM by entering **load java** at the system console.
- 2 Enter the following at the system console:

```
envset CLASSPATH=$CLASSPATH;path_to_append
```

For example, to append the path sys:\myclasses to the current CLASSPATH, enter

```
envset CLASSPATH=$CLASSPATH;sys:\myclasses
```

The \$ symbol in \$CLASSPATH allows the substitution of the current value of the CLASSPATH environment variable.

NOTE: The default CLASSPATH variable includes (.), which means to look in the current working directory (CWD) for the application. If the CWD variable is not set correctly, the application might not run. For more information, see ["Setting the CWD Environment Variable" on page 27](#).

Performance Tuning

The default native (C) stack size for Java threads on NetWare is 160 KB. For applications that create many threads and do not require such a large stack size, the -Xss Java command parameter can be used to specify the stack size.

NOTE: For an explanation of the Java command options, refer to the

For an explanation of the Java command options, refer to the [Java 2 SDK documentation \(http://java.sun.com/j2se/1.4.2/docs/index.html\)](http://java.sun.com/j2se/1.4.2/docs/index.html) for the Java Interpreter.

To specify the amount of virtual memory available to Java applications for non-heap allocations, use the -vm Java command option. The default is 64 MB for NetWare 6 and 100 MB for NetWare

6.5. The -vm Java command option is unique to NetWare and is not supported by JavaSoft. The following is the syntax for this option:

```
java -vmsize MyClass
```

For example,

```
java -vm128m MyClass
```

Java Configuration Files

This section explains the following components:

- ♦ “Java.cfg Configuration File” on page 28
- ♦ “Autoexec.ncf” on page 28

Java.cfg Configuration File

The java.cfg file is an optional configuration file that can be used to set Environment variables. It is located in the sys:\etc directory and is read when java.nlm is loaded.

Autoexec.ncf

The following lines are added to the sys:\system\autoexec.ncf file during the installation. If they are not added, you can add them manually right after the “mount all” command in the file.

```
SEARCH ADD SYS:\JAVA\BIN
```

```
SEARCH ADD SYS:\JAVA\NWGFX\BIN
```

```
STARTX
```

TIP: The STARTX line is optional and should go at the end of the autoexec.ncf file. It causes the NetWare GUI to begin running at server startup. You should delete any other instances of these lines in the autoexec.ncf file.

Internationalization

This section explains the NJVM internationalization issues.

- ♦ “Russian Keyboard Activation” on page 28
- ♦ “AWT Limitations” on page 29
- ♦ “Adding Fonts with Font.properties.allfonts” on page 29

Russian Keyboard Activation

If your server is not installed as a Russian server, you must enable the Russian keyboard. To enable the Russian keyboard, select it in the NetWare GUI setup utility. The Russian keyboard toggles between standard and Russian characters in the same manner as done by keyb.nlm. The keyboard mode on startup uses standard characters. To toggle between modes, press the left and right shift keys simultaneously.

AWT Limitations

Dead keys are not supported in AWT components.

AWT components can display characters only from the current locale.

AWT components cannot display Japanese, Chinese, or Korean characters.

Keyboard entry of Russian characters in AWT components is available only if the current server language is Russian (language 13).

Adding Fonts with `Font.properties.allfonts`

The `font.properties` file with no extension is the default file for Latin character languages. Files with country-specific extensions are provided for Russian, Japanese, Chinese, and Korean servers. For example, `font.properties.ru` corresponds to the Russian `font.properties` file. No configuration is necessary beyond setting the server language during installation because NJVM automatically selects the appropriate `font.properties` file for the language setting. However, some servers might require systems or other resources located on different language platforms. To enable full character support, back up the `font.properties` file, then rename the provided file `font.properties.allfonts` to `font.properties`.

NOTE: Renaming `font.properties.allfonts` to `font.properties` slows the startup of the first application because of the additional characters that need to load.

JVM Client and Server Versions

NJVM 1.4.2 includes two implementations of the HotSpot* virtual machine (VM). This section explains the applications for each version.

This section contains the following client and server version information:

- ♦ “Client VM” on page 29
- ♦ “Server VM” on page 29
- ♦ “Switching JVM Versions” on page 30

Client VM

The Client VM is the default version of the HotSpot runtime compiler. It is specifically tuned to reduce application startup time and memory footprint.

The Client VM is located in `sys:\java\bin\client\jvm.nlm`

Server VM

The Server VM is specifically tuned to maximize peak operating speed. It is intended for long running server applications. The server version requires a larger memory footprint to execute a Java application than that of the client version. You might use this version when operating speed is more important than startup time.

Also, on NetWare you cannot run an application in server mode and another in client mode simultaneously. All applications must be in the same mode.

The Server VM is located in `sys:\java\bin\server\jvm.nlm`.

Switching JVM Versions

On NetWare you can run applications with either the Client VM or the Server VM. You cannot use both simultaneously. The command-line flags used to specify which VM to load are `-client` and `-server`. You cannot switch VMs while Java applications are running. If you attempt to start an application using one of these flags and the VM version requested is different than what is already being used, then that Java application will fail to load.

The default version of the JVM is the Client VM. To change the default VM to server, add the following line to the `sys:\etc\java.cfg` file:

```
JAVA_COMPILER=server
```

Running an application with the Server VM might require more heap space than with the Client VM. To increase the heap size, use the `-Xmxheapsizem` flag. For example, you might have a Java application that requires 128 MB of heap to run with the client JVM. Changing to the server might require more heap space, so you could set the heap size flag to `-Xmx256m`.

5

Java Application Development

This section contains information for developing Java applications with NJVM.

- ♦ “Using Java Native Interface (JNI) on NetWare” on page 31
- ♦ “Source-Level Debugging” on page 34
- ♦ “Obtaining the Fully Qualified Domain Name of the Local Host” on page 35

Using Java Native Interface (JNI) on NetWare

The java\bin directory includes a file named jni.zip, which contains simple native method examples for Java 1.x. This section provides an example of how to write native methods for NJVM. However, this is not a comprehensive tutorial on writing native methods.

NOTE: When writing graphical Java applications to run on NetWare, we recommend using the Swing classes from Sun Microsystems.

- ♦ “Required Tools” on page 31
- ♦ “File Descriptions” on page 32
- ♦ “Building a JNI Example” on page 32
- ♦ “Installing a JNI Example” on page 32
- ♦ “Running a JNI Example” on page 32
- ♦ “Unloading JNI” on page 33

Required Tools

- ☐ NJVM
- ☐ JDK* 1.4.2 for Win32
- ☐ NetWare SDK March 2003
- ☐ WATCOM* 11.0 (10.6 is not supported.)
- ☐ Microsoft* NMAKE (WATCOM supplies one, but use it at your own risk.)
- ☐ MKS Utilities (grep, cp, rm, sed) in your PATH

These tools and build environment are for the examples in jni.zip. Other compilers (like Metroworks) and build tools can be used.

File Descriptions

File	Description
makefile	Makefile for NetWare
nwimpl.c	Native method C implementation
nwmain.c	CLIB NLM™ wrapper - main()
nwnative.java	Native method Java class
nwtest.java	Test class
readme.txt	Known issues and recent updates

Building a JNI Example

After you unpack the package, complete the following:

1 Configure the following parameters in the makefile:

- ♦ WIN32JAVABASE: The location where the Win32 JDK is installed
- ♦ NWJAVABASE: The location where NJVM is installed (typically the mounted sys: drive)
- ♦ NLMSDKBASE: The location where the NetWare NDK is installed
- ♦ WATCOMBASE: The location where Watcom 11.0 is installed

2 Enter the following at a Windows command prompt:

```
nmake
```

3 To clean the build, enter the following at a Windows command prompt:

```
nmake clean
```

Installing a JNI Example

After you build the example and map your server volume sys: (for example, to drive G:), complete the following steps to install JNI:

1 Check to see if g:\java\classes exists as a directory. If not, enter

```
mkdir g:\java\classes
```

2 Copy the NLM to g:\java\bin by entering

```
copy nwnative.nlm g:\java\bin
```

3 Copy the classes to g:\java\classes, then at the command prompt in Windows, enter

```
copy *.class g:\java\classes
```

Running a JNI Example

If your CLASSPATH variable is correct, you can run JNI by entering the following at the command prompt:

```
java NWTest
```


Unloading JNI

To unload JNI, enter `java -exit` at the system console.

JNI Notes

This section contains notes and examples that might be helpful with the JNI process:

- ♦ What you can do in your `main()` function depends on how you build your NetWare Loadable Module™ (NLM) program. If you include the following option in your makefile, your NLM program can use a synchronized startup:

Option `SYNCHRONIZE`

With the synchronize option, you can initialize any global information your NLM program might contain in the `main()` function as long as you call the following function after initialization has completed:

```
void SynchronizeStart();
```

For example:

```
main()
{
    /* Do global initialization */
    SynchronizeStart(); /*MUST BE CALLED */
    ExitThread (TSR_THREAD,0); /* MUST BE CALLED*/
}
```

If you choose to not use a synchronized startup for your NLM program, you must limit your `main()` function to the following:

```
main ()
{
    ExitThread (TSR_THREAD,0);
}
```

- ♦ If you use the Metroworks IDE, the Synchronized linker option can be turned on by pressing Alt+F7, clicking NLM Linker, clicking Flags, then clicking Synchronize.
- ♦ Do not use the standard `malloc()`, `realloc()`, or `free()` calls directly. Java provides the following macros in `sys_api.h` instead:
 - ♦ `sysMalloc`: Same parameters as `malloc()`
 - ♦ `sysFree`: Same parameters as `free()`
 - ♦ `sysRealloc`: Same parameters as `realloc()`
 - ♦ `sysCalloc`: Same parameters as `calloc()`

Using these macros gives you free resource tracking. This also lets the memory used by your NLM program use virtual memory in NetWare 6.5. In some instances, you might prefer memory returned from `malloc`, such as buffers used for callbacks or ECBs.

- ♦ When you link your NLM, you might get the following errors:

Warning! W1008: cannot open math387s.lib: No such file or directory
Warning! W1008: cannot open noemu387x.lib: No such file or directory
Warning! W1008: cannot open emu387x.lib: No such file or directory
Warning! W1008: cannot open clib3s.lib: No such file or directory

NOTE: If you attempt to resolve this issue by linking the correct library, the makefile will link the wrong prelude.obj.

To prevent these warnings, do one of the following makefile options:

- ♦ If you are using C, add the following line to the makefile:

```
Option NoDefaultLibs
```

- ♦ If you are using C++, remove the \$(PRELUDE) entry from the file directive and add the following lines to the makefile:

```
LIBPath $(WATCOM)\lib386;$(WATCOM)\lib386\netware;  
LIBFile $(WATCOM)\lib386\plbx3s.lib
```

Using WATCOM 11.0 Compiler Flags for Native Method NLM Programs

Optimized flags are

```
/zp=1 /ri /ei /5s /or /ot /w3 /s /zq /ez
```

Debug flags are

```
/zp=1 /ri /ei /d2 /od /3s /w1 /s /zq /ez
```

IMPORTANT: /ri and /ei are critical for building native method NLM programs.

Source-Level Debugging

You can debug applications running on NetWare using any JDPA-compliant debugger or debugging IDE. The following are available debuggers:

- ♦ NetBeans (www.NetBeans.org)
- ♦ Forte (forte.sun.com)
- ♦ The Novell exteNd IDE
- ♦ Eclipse (www.eclipse.org)

Remote and local debugging of Java applications are available through the socket transport (dt_socket).

Required VM Options

To debug applications, you need the following debug invocation options:

- ♦ **-Xdebug** enables debugging.
- ♦ **-Xrunjdw** *suboptions* loads the JPDA reference implementation of JDWP. This library resides in the target VM and uses JVMDI and JNI to interact with it. It uses a transport and

the JDWP protocol to communicate with a separate debugger application. The following table describes the specific suboptions:

Suboption	Required	Default Value	Description
address	Yes, if <i>server</i> = n; otherwise, No		Transport address for the connection. If <i>server</i> =n, attempt to attach to the debugger application at this address. If <i>server</i> =y, listen for a connection at this address.
help	No	N/A	Prints a brief help message and exits the VM.
server	No	n	If y, listen for a debugger application to attach; otherwise, attach to the debugger application at the specified address. If y, and no address is specified, choose a transport address to listen to a debugger application and print the address to the standard output stream.
transport	Yes	no value, must be dt_socket	Name of the transport to use when connecting to the debugger application.
suspend	No	y	If y, the JVM waits for a debugger to connect before loading the main class. If n, the JVM runs the program for the debugger to connect in the background while the program is running.

Java -Xdebug Example

```
java -Xdebug  
-Xrunjdp:transport=dt_socket,server=y,address=8000  
-Xbootclasspath/a:SYS:/myapp/foo.jar HelloWorld
```

This listens for a socket connection on port 8000. Suspend this VM before the main class loads. After the debugger application connects, it can send a JDWP command to resume the VM of the HelloWorld application.

Obtaining the Fully Qualified Domain Name of the Local Host

With NJVM, the `java.net.InetAddress.getLocalHost().getHostName()` call returns the name of the NetWare server. To obtain the fully qualified domain (DNS) name, (FQDN) of the local host, use the following in your program:

```
InetAddress localhost = InetAddress.getByName(InetAddress.getLocalHost().getHostAddress()  
);
```

The object, *localhost*, now contains the correct FQDN and can be obtained via the `localhost.getHostName` call.

6

Troubleshooting

This section provides information on the following issues:

- ♦ “Java GUI Applications Fail to Load” on page 37
- ♦ “The NetWare GUI Fails to Load” on page 37
- ♦ “Mouse Does Not Respond” on page 37
- ♦ “Applications Using Java.net Do Not Work Properly” on page 38
- ♦ “Security Alert Dialog Box Appears When Accessing Secure Web Sites” on page 38

Java GUI Applications Fail to Load

If Java applications have trouble running in the NetWare GUI, ensure that the DISPLAY variable is set to 127.0.0.1:0.0 or to the IP address of the server. For more information, see “[Setting the DISPLAY Environment Variable](#)” on page 25.

The NetWare GUI Fails to Load

If the NetWare GUI fails to load after you install NJVM, run the vesa_rsp.ncf file at the system console.

This resets the NetWare GUI to its default configuration. For more information on the default configuration, see “[Configuring Your System for an Accelerated Video Driver](#)” on page 20.

The GUI resolution can then be changed from the taskbar menu. For more information, see “[Configuring X Server Using the NetWare GUI Environment Utility](#)” on page 20.

Mouse Does Not Respond

If the mouse that is properly attached to your server does not function in the NetWare GUI, it might not be configured correctly.

To reconfigure the mouse:

- 1 In the NetWare GUI, click Novell, select Close GUI, then click Yes.
- 2 At the system console, enter
vesa_rsp
- 3 Start the NetWare GUI by entering **startx** at the system console.

This automatically reconfigures the mouse and also resets the NetWare GUI to its default configuration. For more information on the default configuration, see “[Configuring Your System for an Accelerated Video Driver](#)” on page 20.

The NetWare GUI resolution can then be changed from the taskbar menu. For more information, see “[Configuring X Server Using the NetWare GUI Environment Utility](#)” on [page 20](#).

Applications Using Java.net Do Not Work Properly

If Java applications that use the java.net package have trouble resolving host names or IP addresses, ensure that the TCP/IP configuration includes the file `\etc\resolv.cfg` and that WINSOCK is loaded on the server. The `\etc\resolv.cfg` file tells the server its fully qualified domain name (FQDN) and allows the server to locate name servers that can resolve names into IP addresses.

Another problem might be that the IP address and DNS name of the client machine need to be added to the `\etc\hosts` file. The proper entry format is:

IP_address DNS_name

Security Alert Dialog Box Appears When Accessing Secure Web Sites

This dialog box might appear when you access a secure site such as NetWare Remote Manager (NRM) in the X Server GUI on the console. This dialog box displays a security certificate, which is a file sent from the server when you receive content from that server. By accepting the certificate, you indicate that you trust the content coming from the server.

The following are the certificate options:

- ♦ **Once** means the certificate is valid for one session and you agree to receive content for only that session.
- ♦ **Always** means the certificate is valid for all NRM sessions meaning you agree to always receive content from the server.
- ♦ **No** means you do not accept the certificate. This cancels your NRM session.