

# LABVIEW™

## Version 5.0

These release notes introduce you to LabVIEW, describe the system requirements for the LabVIEW software, and contain installation instructions and updated documentation information.



**Note**

*LabVIEW is Year-2000 compliant. Because LabVIEW has never stored two-digit years, the change to 2000 does not affect any internal storage of dates.*

## Contents

---

How to Proceed .....	2
Required System Configuration .....	3
Windows .....	3
Macintosh.....	4
UNIX .....	5
Installing LabVIEW .....	6
Windows .....	6
Macintosh.....	8
UNIX .....	9
Data Acquisition and GPIB Installation Notes .....	11
Windows .....	11
Macintosh.....	12
GPIB and VXI Installation Notes for Sun .....	12
Where to Go from Here .....	13
Examples and Solutions for Your LabVIEW Programs .....	13
Additional Resources .....	13
Differences between LabVIEW for Windows 3.1, Windows 95, and Windows NT .....	14
File Names .....	14
Code Interface Nodes.....	14
Dynamic Link Libraries.....	15
Low-Level Register I/O .....	15
ActiveX and Drag and Drop .....	15
Common LabVIEW Launch Errors on UNIX .....	16
Configuring LabVIEW Windows on UNIX .....	16

Configuring LabVIEW with the Tab Window Manager.....	16
Configuring LabVIEW with the HP VUE Window Manager.....	16
Configuring LabVIEW with the Motif Window Manager.....	17
Notice to Sun SPARCstation 5 Users.....	17
Notice to HP-UX Users.....	19
Notice to Concurrent PowerMAX Users.....	19
Manual Clarifications and Additions.....	19
Multithreading.....	19
ActiveX.....	19
Instrumentation.....	20
General Interface Features.....	20
Support for Template VIs and Controls.....	24
Adding VIs to the Project and Help Menus.....	25
Applibs and Toolkits.....	25
Allocation of Threads on Concurrent PowerMAX and Solaris 2.....	26
Clarifications to the LabVIEW User Manual.....	27
VISA Error Codes.....	28

## How to Proceed

---

If you are upgrading from a previous version of LabVIEW, carefully read the *LabVIEW Upgrade Notes* included with your upgrade package before continuing with this installation. You need to consider several issues before converting your VIs to this version of LabVIEW.

Read the *Required System Configuration* section, then follow the instructions in the *Installing LabVIEW* section of these release notes. After you have installed LabVIEW, see the *Where to Go from Here* section for information about getting started with LabVIEW. In addition, you should read the *Manual Clarifications and Additions* section before using LabVIEW 5.0.

# Required System Configuration

---

This section describes the minimum system requirements needed to run LabVIEW 5.0.

## Windows

LabVIEW 5.0 for Windows is distributed primarily on CD. The LabVIEW CD contains versions for Windows 3.1, Windows 95, and Windows NT and includes the complete instrument driver library available when LabVIEW 5.0 released.

If you do not have a CD-ROM drive, the Windows versions are available on floppy disks as a Base Development System or a Full Development System. The Base Development System consists of a set of disks for installing LabVIEW and the associated VIs. The Full Development System includes the Base Development System disks as well as a second set of disks for installing the advanced analysis VIs.

The *LabVIEW Online Tutorial* default configuration requires the LabVIEW 5.0 distribution CD to be in your CD-ROM drive. You also can install the *LabVIEW Online Tutorial* files on your hard drive. This installation requires approximately 40 MB of hard disk space.

The *LabVIEW Online Tutorial* and LabVIEW Help files contain 256-color graphics. Your video driver, configured through **Control Panels»Display**, must be configured for 256 colors. Minimum requirements to view the tutorial are 800 × 600 pixel resolution and the Microsoft Video for Windows driver. For viewing Help files, National Instruments recommends that you configure your video driver for 256 colors with 800 × 600 pixel resolution.



### Note

**(Windows NT)** *If your display is in 16-bit mode, you might want to change it to 256 or 24-bit color for better performance when viewing the LabVIEW Online Tutorial.*

**(Windows 3.1)** LabVIEW 5.0 runs on Windows 3.1 and Windows for Workgroups 3.1 in 386 enhanced mode. National Instruments recommends that you have a minimum of 8 MB of RAM and at least 60 MB of hard disk space. LabVIEW can run on an 80386-based PC, but a computer with an 80486 CPU is strongly recommended. In addition, LabVIEW for Windows requires a math coprocessor if you are using an 80386- or 80486-based PC.

**(Windows 95)** LabVIEW 5.0 runs on any system that supports Windows 95. You should have a minimum of 16 MB of RAM, but National Instruments recommends 32 MB for this version to run effectively.

**(Windows NT)** LabVIEW 5.0 only runs on Windows NT 4.0 Service Pack 3 or later. You should have a minimum of 16 MB of RAM and a 486/DX processor, but National Instruments recommends 32 MB of RAM and a Pentium processor for this version to run effectively. LabVIEW for Windows NT only runs on Windows NT 80x86 computers. It does not run on other processors, such as the DEC Alpha, MIPS, or PowerPC, because LabVIEW uses 80386 instructions. DEC Alpha, MIPS, and PowerPC 80x86 emulators must emulate 80386 instructions to run LabVIEW.



**Note**

*On Windows systems, you need approximately 60 MB of disk storage space to install the Base package and at least 85 MB for the Full Development System.*

## Macintosh

LabVIEW for Macintosh is distributed either on a CD containing both 680x0 and Power Macintosh software, or on a set of floppies for the 680x0 Macintosh. In addition to the LabVIEW application, the CD also contains an instrument driver directory that includes all the LabVIEW instrument drivers available when LabVIEW 5.0 released.

The floppy disk version of LabVIEW for Macintosh is available as a Full Development System, which contains a set of disks for installing LabVIEW and the associated VIs. In addition, you also can install the VXI VI library.

LabVIEW for the 680x0-based Macintosh requires a math coprocessor. You cannot use a 68000-based computer, such as the Macintosh SE. If you are using a Macintosh IIsi or a Macintosh LC, you might not have a math coprocessor in your computer. Also, computers using the 680LC40 processor (Centris 610s and some low-end Centris 650s) do not have the math coprocessor functions, and therefore cannot run LabVIEW.

LabVIEW requires a minimum of 6 MB of available RAM, in addition to the RAM requirements for your system software and any other applications that you want to run simultaneously. National Instruments recommends that you have at least 8 MB of RAM. You might need more memory, depending on the size of the application you design in LabVIEW and the amount of data that your application manipulates.

**(Power Macintosh)** LabVIEW requires 24 MB of RAM and at least 100 MB of hard disk space.

**(680x0-Based Macintosh)** LabVIEW requires at least 100 MB of hard disk space.

LabVIEW requires System 7 or 8 and cannot run on System 6 or any previous versions of the Macintosh operating system.

For more accurate timing, install the Apple QuickTime extension. When you use QuickTime, timing accuracy should increase from 16.6 ms resolution to approximately 1 ms resolution. System response varies depending on background applications, other extensions, networking activity, and disk caching.



**Note**

*On Macintosh systems, you need approximately 70 MB of disk storage space for a minimal installation of LabVIEW or 120 MB for a full installation.*

## UNIX

LabVIEW for Sun and HP-UX is distributed on CD only. LabVIEW for Concurrent PowerMAX is distributed on 4 mm DAT tape. These versions are not available on floppy disks. LabVIEW requires an X Window System server, such as OpenWindows 3, HP-VUE, or X11R6. LabVIEW does not require a specific graphical user interface (GUI) such as Motif or OpenLook, because LabVIEW uses `xlib` to create its own GUI.

LabVIEW for Sun runs on SPARCstations with SunOS 4.1.3 (Solaris 1.1) or later and Solaris 2.4 or later. LabVIEW for HP-UX runs on Hewlett-Packard Model 9000 Series 700 computers with HP-UX 9.0.5 or later. LabVIEW for Concurrent PowerMAX runs on PowerMAX version 4.1 or later.

The workstation should have 32 MB of RAM with 32 MB or more of swap space storage for LabVIEW to run effectively. In addition, the workstation must have a minimum of 65 MB of disk storage space if you want to install the entire LabVIEW package. To save space, install only the VIs you plan to use.

LabVIEW uses a directory for storing temporary files. Some of the temporary files are large, so keep several megabytes of disk space available for this temporary directory. The default for the temporary directory is `/tmp`. You can change the temporary directory by selecting **Edit>Preferences...**

If LabVIEW aborts unexpectedly, it might leave files behind in the temporary directory. Remove old files occasionally to avoid depleting your disk space.

**(Sun)** You can use a TMPFS file system for this directory to improve performance. For Solaris 1.x, refer to the *Sun System and Network Administration* manual, part number 800-3805-10, for more information about the TMPFS file system. Solaris 2 uses TMPFS by default.

## Operating System Patches on Sun

If you plan to run LabVIEW on SunOS 4.1.3, you need to obtain the latest revision of the following patch from Sun:

- 100458-xx: Setitimer sometimes fails to deliver SIGALRM

You do not need any other patches to run LabVIEW on Solaris 2.4 or later.

## Operating System Patches on Concurrent PowerMAX

You need at least the following patches to run LabVIEW on Concurrent PowerMAX 4.1. The base patches are odd numbers only.

patch	base-001	Base System Patch 001
patch	base-003	Base System Patch 003
patch	base-005	Base System Patch 005
patch	base-007	Base System Patch 007
patch	base-009	Base System Patch 009

## Installing LabVIEW

---

If you are upgrading from an earlier version of LabVIEW, read the *LabVIEW Upgrade Notes* included with your upgrade package before continuing with this installation.

### Windows

Complete the following steps to install LabVIEW for Windows.

1. **(Windows NT)** Log on to Windows NT as an administrator or as a user with administrator privileges.
2. **(CD Installation)** If you are installing LabVIEW for Windows 95/NT, run `x:\WIN95-NT\INSTALL\DISK1\SETUP`, where `x` is the drive letter for your CD-ROM drive. If you are installing LabVIEW for Windows 3.1, run `x:\WIN31\INSTALL\DISK1\SETUP`.  
**(Floppy Installation)** Insert Disk 1 and run `setup.exe`.
3. The installer gives you the option of performing a full installation or a minimal installation. If you do not have sufficient disk space (approximately 85 MB), choose the minimal installation and use your LabVIEW CD to access the remaining components.
4. After you choose an installation, follow the instructions that appear on your screen.

5. **(Windows 95)** To use ActiveX Automation, you need to install Microsoft DCOM 95.  
**(CD Installation)** The LabVIEW installer automatically launches the installer for DCOM 95.  
**(Floppy Installation)** From Disk 1 of the DCOM for Windows 95 floppy set, install DCOM 95 then launch `dcom95.exe`.
6. If you use data acquisition (DAQ) devices with LabVIEW, complete the following step. For more information about installation and configuration of DAQ drivers, refer to the [Data Acquisition and GPIB Installation Notes](#) section of this document.  
**(Windows 95/NT)** The LabVIEW installer calls the NI-DAQ installer, which installs the correct DAQ driver software.  
**(Windows 3.1)** You need to run the NI-DAQ installer yourself. Run `DRIVERS\DAQ\Setup.exe`.
7. The LabVIEW 5.0 CD contains all the latest versions of our GPIB drivers, located in the `drivers` directory. If you use GPIB devices with LabVIEW, complete the following step. For more information about installation and configuration of GPIB drivers, refer to the [Data Acquisition and GPIB Installation Notes](#) section of this document.  
**(Windows 95/NT)** The LabVIEW installer calls the National Instruments GPIB installer, which installs the GPIB drivers and modifies the registry.  
**(Windows 3.1)** Use the GPIB drivers that came with your board or device.
8. **(Full Development System)** If you have the Full Development System, complete the following steps.
  - a. **(CD Installation)** If you are installing LabVIEW for Windows 95/NT, the LabVIEW installer installs the Advanced Analysis VIs automatically. If you are installing LabVIEW for Windows 3.1, you must run the analysis setup program, located in `x:\Analysis\WIN31`, where `x` is the drive letter for your CD-ROM drive.  
**(Floppy Installation)** The Full Development System provides a second set of disks that contain the Advanced Analysis VIs. The first disk of this set contains an installer for these VIs. The installer, named `SETUP.EXE`, is similar to the LabVIEW installer. Insert Disk 1 of the analysis VI disks and run `SETUP.EXE` from the Program Manager as described in Step 2. Follow the instructions on your screen.
  - b. **(CD Installation)** If you are installing LabVIEW for Windows 95/NT, the LabVIEW installer installs NI-VISA automatically. If you are installing LabVIEW for Windows 3.1,

run `x:\Drivers\VISA\WIN31\Disk 1\SETUP.EXE`, where `x` is the drive letter for your CD-ROM drive, to install NI-VISA.



**Note**

*If you are using the CVI runtime engine, run `setup.exe` from Disk 5 of the VISA driver kit.*

**(Floppy Installation)** Insert Disk 1 of the VISA driver kit and run `SETUP.EXE`. Follow the instructions on your screen.

After you have completed the installation, LabVIEW is ready to run. If you plan to use DAQ or GPIB devices with LabVIEW, you must restart your computer to load the new drivers.

By default, the *LabVIEW Online Tutorial* requires the LabVIEW 5.0 distribution CD to be in the CD drive. If you want to install all the *LabVIEW Online Tutorial* files on you hard drive, copy the files located in `x:\tutorial` to the LabVIEW directory on you hard drive, where `x` is the drive letter for your CD-ROM drive. Overwrite any files that already exist in that directory.



**Note**

*If you have installed LabVIEW on a server, new users might want to copy the `Activity` directory from the server to their local machine. You use the `Activity` directory to complete activities that illustrate basic LabVIEW concepts. You can find these activities in the LabVIEW User Manual and the LabVIEW Online Reference, which you can access by selecting **Help»Online Reference....***

## Macintosh

Complete the following steps to install LabVIEW for Macintosh.

1. **(CD Installation)** If you are installing LabVIEW on a Power Macintosh, double-click the LabVIEW Installer icon in the Power Macintosh directory. If you are installing from a CD on a 680x0-based Macintosh, double-click the LabVIEW Installer icon in the 680x0 Macintosh directory.

**(Floppy Installation)** Insert Disk 1 and double-click the LabVIEW Installer icon.

2. The installer gives you the option of performing a full installation or a minimal installation. If you do not have sufficient disk space (approximately 120 MB), choose the minimal installation and use your LabVIEW CD to access the remaining components.
3. After you click the **Install** button, you are prompted to select a destination directory. Click the **New Folder** button to create a LabVIEW directory, then click the **Install In Current Folder** button to install the LabVIEW files in that directory.
4. **(Floppy Installation)** If you are installing the Full Development System from floppy disks, you have additional sets of disks that contain the

VXI library VIs. If you do not need VXI support, you can ignore this set of disks. The first disk of this set contains an installer for these VIs. The installer, VXI Library, is similar to the LabVIEW installer. Insert Disk 1 of the VXI VI disks and double-click the installer icon.

After you have completed the installation, LabVIEW is ready to run. If you plan to use DAQ or GPIB devices with LabVIEW, you must restart your computer to load the new drivers.

## UNIX

Complete the following steps to install LabVIEW for UNIX.

### Solaris 1

1. To enable superuser privileges, type `su root` and enter the root password.
2. If the directory `/cdrom` does not exist, type the following command:  
`mkdir /cdrom`
3. Mount the LabVIEW CD and type the following command:  
`mount -rt hsfs /dev/sr0 /cdrom`
4. To change to the installation directory, type  
`cd /cdrom/solaris1`
5. To run the installation script, type the following command:  
`./INSTALL`
6. Follow the instructions on your screen.

### Solaris 2

1. To enable superuser privileges, type `su root` and enter the root password.
2. Insert the LabVIEW CD. On Solaris 2.x, the CD automatically mounts as soon as the CD is inserted into the drive. If this feature is disabled on your workstation, you must mount the CD by typing the following command:  
`mount -o ro -F hsfs /dev/dsk/c0t6d0s2 /cdrom`
3. If your CD was mounted automatically, type the following command:  
`pkgadd -d /cdrom/cdrom0/solaris2`  
If you used the mount command in step 2, type the following command:  
`pkgadd -d /cdrom/solaris2`

4. See the README file in `/cdrom/cdrom0/solaris2` or `/cdrom/solaris2` for instructions on custom installation or other additional information.
5. Follow the instructions on your screen.

## HP-UX

1. To enable superuser privileges, type `su root` and enter the root password.
2. Mount the LabVIEW CD on the `/cdrom` directory with the SAM system administration utility.
3. To change to the installation directory, type the following command:  

```
cd /cdrom/HP-UX
```
4. To run the installation script, type the following command:  

```
./INSTALL
```
5. Follow the instructions on your screen.

## PowerMAX

1. Insert the 4 mm DAT tape into the tape drive.
2. To create the directory in which you will install LabVIEW, type the following command:  

```
mkdir /opt/lv50
```
3. To change to the new directory, type the following command:  

```
cd /opt/lv50
```
4. Extract the files from the tape by typing the following command:  

```
tar xv
```
5. To run the installation script, type the following command:  

```
./INSTALL
```
6. Follow the instructions on your screen.

After you have installed LabVIEW completely, it is ready to run.



### Note

***The LabVIEW User Manual and the LabVIEW Online Reference, which you can access by selecting Help»Online Reference..., provide activities that illustrate basic LabVIEW concepts. If you want to complete these activities, copy the Activity directory from the LabVIEW directory to your home directory.***



### Note

***The LabVIEW documentation set, including the Code Interface Reference Manual and the VXI VI Reference Manual, is available in Portable Document Format (PDF) on the LabVIEW CD in the manuals directory. You can copy this directory***

or selected PDF files to the `LabVIEW\manuals` directory on your hard drive. You must have Adobe Acrobat Reader 3.0 or later installed to view these files.



#### Note

*If you are upgrading from a previous version of LabVIEW, read the LabVIEW Upgrade Notes. If you have one of the add-on packages, such as the LabVIEW Test Executive or the Picture Control Toolkit, consider installing those files at this time.*

## Data Acquisition and GPIB Installation Notes

---

All National Instruments GPIB interfaces and DAQ devices come with the drivers and other software you need to use them. LabVIEW also comes with the drivers and other software you need to use National Instruments hardware. While the drivers included with LabVIEW are the same NI-488.2 and NI-DAQ drivers National Instruments includes with its GPIB and DAQ hardware, the version numbers might differ. Always use the driver with the higher version number. You can determine which version of NI-DAQ you are using with LabVIEW by running the Get Device Information VI. If you already own a device with a version of NI-DAQ earlier than 5.0, do not install that driver.

You must configure your DAQ hardware before you can use the LabVIEW DAQ VIs. To configure your DAQ hardware, you need to run the NI-DAQ Configuration Utility. For more information about how to configure your DAQ hardware with the NI-DAQ Configuration Utility, refer to the Help files installed with LabVIEW.

## Windows

When you install LabVIEW, the installer places the application and most of the related files in a directory you specify. The default name of this directory is `LABVIEW`. If you install DAQ or GPIB VIs, the installer places additional files, described in the following sections.

**(Windows 95/NT)** Use the NI-DAQ Configuration Utility, which you launch by clicking the **Start** button and choosing **Programs»LabVIEW»NI-DAQ Configuration Utility**.

For information about how to configure your particular DAQ device, refer to the NI-DAQ Configuration Help file. To view this Help file, click the **Start** button and select **Programs»LabVIEW»NI-DAQ Configuration Help**. Follow the links for configuring the devices you have.

You can find further information about the NI-DAQ driver in the NI-DAQ Read Me File. To view this file, click the **Start** button and select **Programs»LabVIEW»NI-DAQ Read Me File**.

**(Windows 3.1)** Run `NICFG16.EXE`, found in the Windows `SYSTEM` directory. For instructions about how to configure your particular DAQ device, refer to the NI-DAQ Configuration Help file, `NIDAQCFG.HLP`. To view this Help file, go to `x:\Programs\LabVIEW\NI-DAQ\NIDAQCFG.HLP`, where `x` is the directory in which you installed LabVIEW. Follow the links for configuring the devices you have.

For more information about the NI-DAQ driver, see the `README.WRI` text file in the `LABVIEW` directory.

**(Windows 3.1)** You can use the GPIB driver that came with your device. The LabVIEW CD comes with the latest version of our Windows 3.1 drivers, which is found in the `drivers` directory.

## Macintosh

The LabVIEW installation program installs two control panels and an extension in your system folder. `NI-GPIB` contains the driver code that communicates with your GPIB devices. `NI-DAQ` contains driver code that communicates with your DAQ devices. The `NI-DMA/DSP` extension contains DSP and DMA drivers used by DAQ, GPIB, and DSP drivers.

## GPIB and VXI Installation Notes for Sun

---

The LabVIEW installer prompts you to choose the NI-488.2M drivers for the GPIB hardware you are using (SB-GPIB-TNT, GPIB-ENET, or GPIB-SCSI-A). The installer then installs that driver for you. The only exception is the Solaris 1 NI-488.2M driver for the GPIB-SCSI-A, which you must install separately.

If you have a GPIB-SCSI-A, follow the installation instructions in the documentation that came with your original GPIB-SCSI-A hardware and software kit, including the *Getting Started with Your GPIB-SCSI-A and the NI-488.2M Software for the Sun SPARCstation* manual.



### Note

***LabVIEW does not work with the GPIB-1014 series (VME) devices or the original GPIB-SCSI box. It does work with the newer GPIB-SCSI-A box.***

The VXI device drivers for Solaris 1.x and 2.x are included on floppy disks with the LabVIEW for Sun VXI Upgrade and with the VXI/VME-SB2020 kit. A VXI device driver must be installed on your system to perform VXIbus operations from LabVIEW. Install the appropriate device driver (Solaris 1.x or Solaris 2.x) before beginning development. To install the VXI device driver, refer to the *Getting Started with Your VXISB2020 and the NI-VXI Software for Solaris* manual.

**Note**

*National Instruments periodically updates drivers for GPIB and VXI. If you add new GPIB or VXI hardware for use with LabVIEW, the included drivers might supersede those sent with LabVIEW. Compare the version numbers and use the driver with the higher number.*

## Where to Go from Here

---

The following resources can help get started with LabVIEW 5.0 quickly.

If you are a new LabVIEW user, see the *LabVIEW QuickStart Guide* to learn how to begin programming with LabVIEW. The exercises in this manual guide you through the basic steps of building and debugging a LabVIEW application.

**(Windows)** For an introduction to the LabVIEW environment, complete the exercises in the *LabVIEW Online Tutorial*. Launch the tutorial by clicking **LabVIEW Tutorial** in the LabVIEW dialog box.

**Note**

*To access the LabVIEW dialog box, either launch LabVIEW or close all open LabVIEW VIs if you already are running LabVIEW.*

## Examples and Solutions for Your LabVIEW Programs

**(Windows and Macintosh)** If you are using data acquisition (DAQ) or instrument I/O and want to find examples or generate solutions for your LabVIEW programs, launch the DAQ Solution Wizard by clicking **Solution Wizard** in the LabVIEW dialog box. For more information about the Solution Wizard, see the Chapter 3, *Data Acquisition*, and Chapter 4, *Instrumentation*, of the *LabVIEW QuickStart Guide*.

**(Windows)** To find any other type of example, open the Search Examples Help file by clicking **Search Examples** in the LabVIEW dialog box.

**(Macintosh and UNIX)** The `examples` directory contains a VI named `readme.vi`. With this VI, you can find the available examples. When you select a VI, you can see the documentation that was entered for that VI by choosing **Window»Show VI Info....** To open a VI, choose **File»Open....**

## Additional Resources

**Note**

*The LabVIEW documentation set, including the Code Interface Reference Manual and the VXI VI Reference Manual, is available in Portable Document Format (PDF) on the LabVIEW CD in the `manuals` directory. You can copy this directory or selected PDF files to the `LabVIEW\manuals` directory on your hard drive. You must have Adobe Acrobat Reader 3.0 or later installed to view these files.*

**(Windows and Macintosh)** If you need to perform data acquisition, read the *LabVIEW Data Acquisition Basics Manual*, which contains important information about using the DAQ VIs and examples you can find in LabVIEW. For reference information about a particular DAQ VI, refer to the *LabVIEW Function and VI Reference Manual* and the *LabVIEW Online Reference*, which you can access by selecting **Help»Online Reference....** The *New Features in LabVIEW 5.0* section in the *LabVIEW Upgrade Notes* also contains information about new features and changes in DAQ VIs.

The DAQ examples folder contains a VI library named `RUN_ME.LLB` that has a Getting Started example VI for analog input, analog output, digital I/O, and counters. The `RUN_ME.LLB` examples provide an excellent starting place for data acquisition.

## Differences between LabVIEW for Windows 3.1, Windows 95, and Windows NT

---

LabVIEW is very similar on all three Windows operating systems. Unless your application calls code interface nodes (CINs) or dynamic link libraries (DLLs), uses ActiveX, or communicates with hardware that is not supported by one of the operating systems, you can transfer VIs to other operating systems without any problems or having to make any modifications. The following sections describe the differences between the operating systems.

### File Names

LabVIEW for Windows 95/NT supports long filenames. In addition to the Virtual File Allocation Table (VFAT) file system, LabVIEW for Windows NT also supports the NT File System (NTFS). Windows 3.1 only supports filenames with eight characters plus an optional three characters for the extension. If you use Windows 3.1 or need to transfer files to Windows 3.1, you should use VI libraries (LLBs), a type of LabVIEW file that can contain multiple VIs inside of it. This capability of LLBs enables you to use long names for your VIs even on Windows 3.1.

### Code Interface Nodes

LabVIEW supports the creation of code interface nodes (CINs) on all three operating systems. However, on Windows 3.1 you are limited to creating CINs using the Watcom/C compiler. Watcom/C is the only compiler that can create the kind of 32-bit code LabVIEW supports in Windows 3.1. Windows 3.1 cannot load CINs from Windows 95/NT unless the CINs were created using Watcom.

Windows 95/NT supports CINs created using Visual C++, the Win32 SDK Microsoft C command line compiler (CL386.exe), and Symantec C. Those CINs are not limited in that they can call DLLs and system calls. In addition, Windows 95/NT has backward support for Watcom CINs designed for Windows 3.1, but Windows 95/NT cannot support Watcom CINs that make system calls, load DLLs, or perform low-level register based I/O.

## Dynamic Link Libraries

LabVIEW for Windows 3.1 can call 16-bit dynamic link libraries (DLLs). LabVIEW for Windows 95/NT is a Win32 application and can call only 32-bit DLLs. If you have existing 16-bit Windows 3.1 DLLs that you want to call on Windows 95/NT, you must recompile them to be 32-bit or obtain new versions. If you cannot find a new version of a DLL, it is possible on Windows 95 to write a 32-bit DLL (called a *thunking* DLL) that you can use to call a 16-bit DLL; however, writing this DLL is a fairly difficult process and National Instruments recommends avoiding this approach if possible.

## Low-Level Register I/O

LabVIEW for Windows 3.1/95 has a set of VIs named In Port and Out Port that you can use to read or write hardware registers. Windows NT applications cannot manipulate hardware directly. If you need to communicate with a hardware device in Windows NT, you must write a Windows NT driver.

## ActiveX and Drag and Drop

Drag and Drop and ActiveX Automation functions are supported only on Windows 95/NT. The ActiveX Automation functions enable you to call ActiveX interfaces in other applications, such as Excel. In addition, LabVIEW can act as an ActiveX server.

# Common LabVIEW Launch Errors on UNIX

The following table lists common errors that might occur when you launch LabVIEW for UNIX. See the [Required System Configuration](#) section of these release notes for more information about solving these and other installation problems.

Error Message/Description	Probable Cause/Solution
Xlib: connection to :0.0 refused by server	<b>Probable Cause</b> —Trying to run LabVIEW as a user who does not have permission to open a window on the display server. Typically seen after running the <code>su</code> command to temporarily become a different user, such as root (superuser).
client is not authorized to connect to server	
internal error during connection authorization check	<b>Solution</b> —Exit the <code>su</code> command and launch LabVIEW as the login user.
"Executable version doesn't match resource file"	<b>Probable Cause</b> —Version of LabVIEW executable does not match version of <code>labview.rsc</code> . <b>Solution</b> —Verify that the <code>appResFilePath</code> parameter in the configuration file correctly sets the path to the <code>labview.rsc</code> file.

## Configuring LabVIEW Windows on UNIX

This section describes procedures for configuring LabVIEW windows on UNIX operating systems.

### Configuring LabVIEW with the Tab Window Manager

If you use the Tab Window Manger (`twm`), you can change environment settings so that `twm` interacts better with LabVIEW. With `twm`, you cannot close the floating palette menus in LabVIEW if these windows do not have title bars. To correct this problem, add the following line to your `.twmrc` file in your home directory:

```
DecorateTransients
```

This line adds title bars to the floating windows so you can close them.

### Configuring LabVIEW with the HP VUE Window Manager

If you use the HP VUE Window Manager (`vuem`), you can change environment settings so that `vuem` interacts better with LabVIEW. By default, `vuem` does not incorporate the window position requests of an application. This behavior causes LabVIEW windows, such as the Panel,

Diagram, Help, and file dialog windows, to not appear in consistent locations on your screen. To change the `vuewm` behavior, use the `xrdb` command to set two `vuewm` settings:

```
Vuewm.clientAutoPlace: False
```

```
Vuewm.positionIsFrame: False
```

To add the two entries, you also can edit the following files manually:

```
$HOME/.vue/sessions/home/vue.resources
```

```
$HOME/.vue/sessions/current/vue.resources
```

## Configuring LabVIEW with the Motif Window Manager

If you use the Motif Window Manager (`mwm`), you can change environment settings so that `mwm` interacts better with LabVIEW. By default, `mwm` does not incorporate the window position requests of an application. This behavior causes LabVIEW windows, such as the Panel, Diagram, Help, and file dialog windows, to not appear in consistent locations on your screen. To change the behavior of `mwm`, use the `xrdb` command to set two `mwm` settings:

```
mwm.clientAutoPlace: False
```

```
mwm.positionIsFrame: False
```

To add the two entries, you also can edit the following file manually:

```
$HOME/.Xdefaults
```

## Notice to Sun SPARCstation 5 Users

---

A bug exists in some early revisions of the SPARCstation 5. This bug can cause LabVIEW and other programs to hang the system when executing certain floating-point operations. When this condition occurs, you must physically reset the computer to recover. The problem exists in the firmware of the computer and can occur when running SunOS 4.1.3\_U1, SunOS 4.1.4, and Solaris 2.x.



### Note

*This bug has been reported only on early revisions of the 70 MHz and 85 MHz SPARCstation 5.*

To determine whether your SPARCstation 5 is affected, perform the following steps.



**Caution** *Following these steps temporarily interrupts the operation of your computer, so you should warn anyone who might be using your computer remotely.*

1. From your SPARCstation 5 console, hold down the <Stop/L1> key (located near the upper left corner of your keyboard) and press the <A> key to break into the PROM monitor.
2. You see one of the following two prompts:  
Type b (boot), c (continue), or n (new command mode)>  
Type 'go' to resume ok  
  
In the first case, select n to go to new command mode, where you see an ok prompt. If you already have an ok prompt, skip to step 3.
3. At the ok prompt, type  
module-info  
  
You then see information similar to the following lines:  
CPU FMI,MB86904 Rev. 2.5 : 70.0 MHz  
SBus (Divide By 3) : 23.3 MHz
4. Type go to exit the monitor and resume operation of your system.

If your CPU Revision number (2.5 in this example) is earlier than 3.2, *and* your CPU clock speed (70.0 MHz in this example) is less than 110 MHz, then your computer has this problem. Contact Sun and ask to have your CPU firmware upgraded to swift\_pg 3.2 or later. (Swift is the code name used by Sun for the SPARCstation 5 firmware.) The Sun Bug ID number for this problem is 1151654.

If you have a SPARCstation 5 with this bug, National Instruments strongly recommends upgrading your firmware.



**Note** *This problem can affect programs other than LabVIEW. Notably, the GNU C compiler also can produce code that hangs your system in versions prior to 2.6.0.*

# Notice to HP-UX Users

---

By default, HP workstations limit the size of a process such as LabVIEW to 64 MB. You can change this setting by adjusting a kernel configuration parameter that limits the amount of data a process can use. To edit this parameter, enable superuser privileges by typing `su root` and entering the root password. Use the SAM system administration utility to view the list of kernel configuration parameters. Change the value of the maximum data segment size in bytes parameter to a larger value. If you need to rebuild the kernel and reboot for changes to take effect, the SAM utility guides you through this process.

# Notice to Concurrent PowerMAX Users

---

Attempting to use 7-bit data with the LabVIEW Serial VIs might crash your machine because the operating system only supports 8-bit characters.

Although you can select the hardware flow control option for the LabVIEW Serial VIs, serial-line hardware flow control is not supported by the operating system and therefore does not work.

# Manual Clarifications and Additions

---

This section contains information that was not included in the LabVIEW documentation as well as corrections to the LabVIEW manuals. For information about new features in this version of LabVIEW, see the *LabVIEW Upgrade Notes*.

## Multithreading

### **Color of Code Interface and Call Library Function Nodes—**

In LabVIEW 5.0, the color of a code interface node (CIN) or Call Library Function node on a block diagram changes depending on whether LabVIEW considers it reentrant. If LabVIEW considers a CIN or Call Library Function node reentrant, LabVIEW assigns it the current primitive color (the default is pale yellow). If a CIN or Call Library Function node is not considered reentrant, its color is orange. This color designation exists on all platforms, even if the platform itself is not threaded.

## ActiveX

**ole\_lv5container.dll**—The ActiveX Container uses a DLL named `ole_lv5container.dll`, which is located in the `resource` directory. If you build an application that includes ActiveX controls and move it to another machine, you must install this file in the same directory

as the built application or in the `System` directory. In the LabVIEW documentation, references to `ole_container.dll` should be `ole_lv5container.dll`.

**Data Format**—The compatibility VIs for the LabVIEW 4.x Automation functions require that you pass flattened data in the LabVIEW 4.x format. LabVIEW 5.0 loads your LabVIEW 4.x VIs and automatically selects the **Convert 4.x Data** option for the Flatten To String and Unflatten From String functions.

## Instrumentation

**Signal Generator by Duration VI**—The Signal Generator by Duration VI has been added to the **Analysis»Signal Generation** palette. This VI generates a signal with a shape given by the waveform type: sine, cosine, triangle, square, sawtooth, increasing ramp, or decreasing ramp.

**CVI Function Panel Converter Changes**—The improved CVI Function Panel Converter creates hierarchical text menus so you can find functions quickly. Two new options have been added to the CVI Function Panel Converter. These options are ON by default.

- **Map ViSession type to VISA Session RefNum**—This option specifies that instrument session numbers of type ViSession in the CVI Function Panel are converted to LabVIEW VISA RefNums in the resulting VI. Functions that contain the string `_init` in their name automatically register with the VISA refnum; functions that contain `_close` in their name automatically close the VISA refnum.
- **Create instr.lib menu mirroring CVI Class Hierarchy**—This option specifies that when converting a Function Panel file, a palette menu for the instrument is created in the **Instrument Drivers** menu. This menu is organized hierarchically according to the Function Panel Tree in the `.fp` file.

## General Interface Features

### Icon and Text Palettes

You can display palettes in three modes: Standard, All Icons, or All Text. Choose the palette display mode in the **Edit»Select Palette Set»Display Style** submenu.

In All Icons mode, the **Controls** and **Functions** palettes contain icons that represent options such as controls, indicators, functions, and VIs. These icon palettes are similar to the icon palettes in previous versions of LabVIEW.

In All Text mode, you pop up to access the **Controls** or **Functions** palette. These text palettes contain the names of options. Items in text palettes are organized in the same order as in the icon palettes when you read the icon palette row by row, left to right. Empty spaces in the icon palettes are skipped. Unlike icon palettes, you cannot tack down text palettes or subpalettes. In All Text mode, LabVIEW displays text palettes in the **Project** menu and the Find dialog box. In Standard or All Icons mode, LabVIEW displays these palettes as icon palettes.

Standard is the default mode. In Standard mode, all palettes default to icon palettes, but you can edit individual palettes to display them as text palettes.

When you edit a palette by selecting **Edit>Edit Control & Function Palettes...**, LabVIEW displays the palettes in All Icons mode. You cannot edit palettes in the other modes because they do not contain as much information (icon palettes have both icons and two-dimensional layout, while text palettes do not). In the editor, you can specify the mode for any palette by popping up and selecting either **Icons** or **Text** from the **Standard Menu View** submenu. The mode you select affects only the menu you are editing.



**Note**

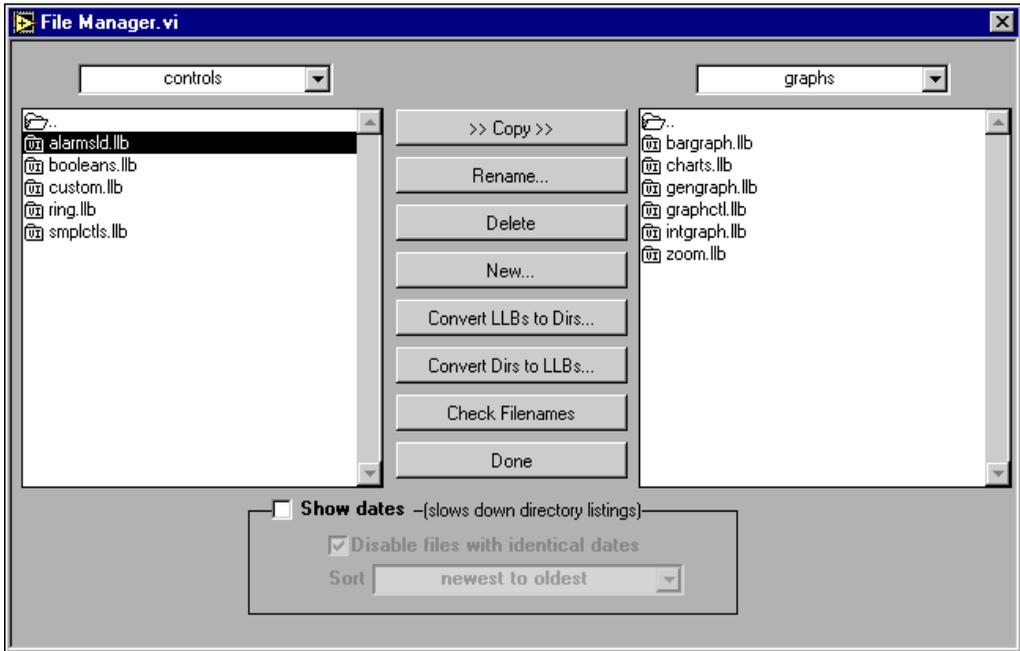
*In LabVIEW 5.0, you pop up on palettes or the icon of a submenu so you have a method for changing the settings of the top-level Controls and Functions palettes.*

## File Manager Tool

The File Manager tool, which you access by choosing **Project>File Manager...**, simplifies copying, renaming, and deleting files within VI libraries (LLBs). You also can use this tool to create new LLBs and directories and convert LLBs to and from directories.

To avoid performing a file operation on a VI already in memory, close all VIs that might be affected before using this tool.

In the File Manager dialog box, shown in the following figure, you can view two locations (directory or LLB) simultaneously. When you select a file, you can copy, rename, or delete it using the corresponding buttons between the two lists. Click **New...** to create a new directory or LLB.



**Figure 1.** File Manager Tool Dialog Box

If you select an LLB, you can click **Convert LLBs to Dirs** to convert it to a directory. If you select a directory and click this button, the tool scans for all LLBs within that directory and gives you the option to convert them to directories. The new directory is created in the same location as the original LLB.

If you assign the new directory a name that differs from that of the original LLB, LabVIEW searches for the files that were within the LLB when calling a VI (even when the name is the same minus the .llb extension). When you convert an LLB to a directory, you have the option to back up the LLB (the .llb extension changes to .llx).

To convert a directory to an LLB, select a directory and click **Convert Dirs to LLBs**.

Click **Check Filenames** to scan a directory or VI library for platform-dependent filenames. The tool scans all filenames for invalid characters (: , \ , / , ? , \* , < , > , or |) and verifies filenames to be 31 characters or less (a limitation on the Macintosh). The **Check Filenames** option also scans files within LLBs. These files are portable, even if their names contain characters that are invalid on some platforms. By scanning within LLBs, this tool helps you detect potential problems if you move your files out of VI libraries.

Use the **Show dates** option at the bottom of the dialog box to display file modification dates next to each file. You can choose to sort the files alphabetically or by date and disable files with the same name and date in both directory listings. Use this technique when comparing two directories to determine whether any files have changed.

## Other General Interface Features

**Dragging and Dropping VI Icons**—LabVIEW 5.0 simplifies the creation of VI icons. By selecting an image file and dropping it onto the VI icon in the upper-right corner of a front panel, a 32-by-32 version of the image replaces the existing icon.

You can drag a VI icon from the icon pane in the upper-right corner to a block diagram to instantly create a subVI call. By pressing <Shift> while dragging the VI icon, you automatically wire the non-default values of the controls as constants for the subVI.

If the subVI already appears in a block diagram, pressing <Shift> while dragging onto the existing call updates the attached constants. A control at its default value discards the constant attached to the subVI, and an input wired to anything other than a constant is unaffected.

When you press <Shift> while double-clicking a subVI icon to open the subVI front panel, LabVIEW loads the values of the constants wired to the subVI into the front panel controls. All unwired controls retain the default values.

You also can use the drag-and-drop technique for global variables and custom controls. Additionally, you can drag a VI icon into a VI refnum on a front panel control to load VIs into memory dynamically, which is part of the VI Server functionality.

**Print to RTF/HTML Feature**—The Print to RTF/HTML feature can export graphics in uncompressed graphics interchange format (GIF).

**Configuration File VIs**—The Configuration File VIs provide tools for reading from and writing to a platform-independent configuration file similar in format to a Windows initialization (.ini) file.

**Macintosh Open Transport Support**—LabVIEW 5.0 supports Open Transport on Power Macintosh machines. Open Transport is a PowerPC-native networking driver.

**New Preferences Options**—LabVIEW 5.0 adds the following two options in the **Miscellaneous** view of the **Edit»Preferences...** dialog box.

- **Automatically close VISA sessions**—Use this option to specify that VISA sessions, like file refnums, close automatically when the top-level VI goes idle. The default is ON, which closes VISA sessions automatically.
- **Treat read-only VI as locked**—Using this option, you can choose whether to treat read-only VIs as locked. You cannot edit locked VIs, but you can re-compile and execute them. By default the option is not selected so that read-only VIs appear normally. However, you cannot save the VI to the same location (the read-only file) unless you change the file permissions outside LabVIEW. This behavior is consistent with the behavior in previous versions of LabVIEW. When using the VI Server, the read-only status of files is ignored except when saving. This option is designed primarily to support the source-code control of the Professional G Developers Toolkit.

**Execution System Selection**—The default preferred execution system for a VI is **same as caller**. This setting allows the VI to run in the same execution system in which caller is running when the subVI call to the VI is made. The **same as caller** setting has the lowest run-time overhead. When you set a VI to **same as caller** and you run it at the top level, it runs in the **standard** execution system at its selected priority.

**(Windows 3.1) lvdevice.dll Change**—`lvdevice.dll` has changed to `resource\lvdev5.dll`.

**Icon Editor**—The **Undo** button has been removed from the Icon Editor, but you can undo an action by selecting **Edit»Undo** or `<Ctrl-Z>`.

**Seconds to Date/Time Function Change for Windows 95/NT**—

In Windows 95/NT, the value of the DST element of the **date time rec** cluster returned by the Seconds to Date/Time function is not set correctly. Instead of returning whether the **date time rec** cluster has been modified for daylight saving time, it returns whether your current system settings are set to account for daylight saving time.

**Offscreen Updates Default Value**—The default value for offscreen updates is now ON instead of OFF.

## Support for Template VIs and Controls

You can save commonly used VIs and controls as templates. To create a template VI, save a VI with a `.vit` extension (or `.ctt` extension for typedefs). When you open a template VI or control, the new file you create is named automatically using your template name and a number corresponding to the number of times it has been opened. When you finish

editing the VI and save it, LabVIEW prompts you to enter a new name for the file.

To modify a template, open it, make your changes, then save over the `.vit` (or `.ctt`) file that you originally created.

**(Macintosh)** You also can use the **Stationery Pad** checkbox of the Get Info dialog box in the Finder to change a VI to a template.

## Adding VIs to the Project and Help Menus

You can add VIs to the **Project** and **Help** menus by placing them inside the `Project` or `Help` directories in the LabVIEW directory. You can use this technique to provide quick access to VIs that act as tools in your system. National Instruments uses this feature to make the Tech Support VIs accessible from the **Help** menu. Also, if you have the Application Builder libraries installed, you can see a **Create Distribution Kit** option in the **Project** menu.

Any VI placed at the top level of the `Project` or `Help` directory is appended directly to the corresponding menu. If you create a subdirectory, a submenu is appended. For subdirectory names in Windows 3.1, which does not support mixed case or long file names, you can place a file with the same name as the directory with a `.txt` extension. If you place a library (`.llb`) inside of one of these directories, only VIs that are marked as top level are appended to the menu bar.

## Applibs and Toolkits

Most existing toolkits function with LabVIEW 5.0 without problems. However, you need to move the VIs so they appear in the menus. LabVIEW 5.0 is compatible with toolkits designed for 3.0, with the following exceptions.

You must upgrade the following toolkits for compatibility with LabVIEW 5.0:

- **LabVIEW Application Builder**—You must upgrade to LabVIEW Application Builder 5.0. This upgrade is free to existing users of the LabVIEW Application Builder. If you have the Professional Development System, the new version of the Application Builder libraries is included in the installation.
- **Professional G Developers Toolkit**—If you have the Professional G Developers Toolkit 4.1, you must upgrade to version 5.0. This upgrade is free to existing users of the Professional G Developers Toolkit. If you have the Professional Development System, the new version of the Professional G Developers Toolkit is included in the installation.

- **LabVIEW Test Executive**—If you use LabVIEW Test Executive 5.0 or earlier, you must upgrade to LabVIEW Test Executive 5.1. This upgrade is free to existing users of the LabVIEW Test Executive 5.0.

With minor exceptions, you can use the previous version of the following toolkits with LabVIEW 5.0:

- **Picture Control Toolkit for G**—You can use the Picture Control Toolkit 1.0 with LabVIEW 5.0 with the exception of the Draw 1-bit Pixmap VI. You can download an updated version of this VI from the National Instruments FTP site (<ftp.natinst.com>). The Picture Control Toolkit is being updated to include the fix mentioned above, and the upgrade is free to existing users.
- **Internet Developers Toolkit for G**—You can use the Internet Developers Toolkit 4.1 with LabVIEW 5.0, but you must delete `printvi.llb`, located in the `user.lib\internet\image` directory. The Internet Developers Toolkit is being updated to version 5.0 to include this fix, and this upgrade is free to existing users.

The following toolkits do not install VIs in a location that causes them to appear in the palettes. These toolkits are being updated to version 5.0. You can use the existing toolkits by moving VIs to `vi.lib\addons` or `user.lib`. Alternatively, you can choose **Edit** » **Edit Control and Function Palettes** and add them to the palette of your choice.

- Picture Control Toolkit 1.0
- Statistical Process Control Toolkit 1.0
- Proportional-Integral-Derivative Toolkit 1.0

## Allocation of Threads on Concurrent PowerMAX and Solaris 2

On Concurrent PowerMAX and Solaris 2, LabVIEW allocates threads in the following two ways.

- If LabVIEW has permission to increase its Light Weight Process (LWP) priorities from the default, it binds all created threads to LWPs.
  - Profiling is very accurate because each thread is bound to a LWP and the kernel monitors the execution timing of LWPs.
  - The LabVIEW priority system is reflected in the way the kernel runs LWPs. Higher-priority execution threads (LWPs) take over the system, not allowing lower-priority system tasks to execute.
  - Switching between threads might require more time because LabVIEW runs through the system scheduler.
- If LabVIEW cannot increase its LWP priorities from the default, it creates a LWP per thread, but leaves the threads and LWPs unbound so the created threads have a pool of LWPs on which to run. The typical

user does not have permission to raise LWP priorities. If LabVIEW threads are not bound to LWPs:

- Profiling strictly uses wall-clock time. Threads might switch LWPs dynamically without kernel knowledge, so LabVIEW cannot use LWP timing statistics.
- The LabVIEW priority system only has an effect internal to LabVIEW. The system treats all the LabVIEW LWPs as another process to schedule at the same priority as any other task in the system.
- Context switching between threads might be faster because it does not involve the kernel scheduler.

The About LabVIEW dialog box, which you can view by choosing **Help»About LabVIEW...**, indicates how LabVIEW currently allocates threads.

## Clarifications to the LabVIEW User Manual

The following clarifications pertain to the *LabVIEW User Manual*:

- **(Windows 3.1)** In the *Organization of the LabVIEW System (Windows)* section of Chapter 1, *Introduction*, `lvdevice.dll` in the resource directory should be `lvdev5.dll`.
- In Chapter 2, *Creating VIs*, the text and an illustration refer to an **Undo** button in the Icon Editor. The **Undo** button has been removed, but you can undo an action by selecting **Edit»Undo** or `<Ctrl-Z>`.
- In Activity 6-3, *String Subsets and Number Extraction*, the block diagram shows the From Exponential/Fract/Eng function. The block diagram should show the Scan From String function, as described in the text.
- In Activity 15-3, *Calculate Harmonic Distortion*, the pathname for library that includes the THD Example VI should be `examples\analysis\measure\measxmpl.ll`.
- **(Windows 95)** In the *Using NetDDE* section of Chapter 23, *Using DDE*, the manual refers to `REGEDIT` and `REDEGIT` executables. The correct name is `REGEDIT`.
- In Chapter 25, *Program-to-Program Communication*, the *PPC Client Example* section refers to the PPC Open Connection, PPC Open Session, PPC Close Session, and PPC Close Connection VIs. These should be the PPC Open Port, PPC Start Session, PPC End Session, and PPC Close Port VIs, respectively. The *PPC Server Example* section refers to the PPC Close Session VI, which should be the PPC End Session VI.

# VISA Error Codes

The following table lists numeric VISA error codes that were not included in the printed documentation.

Error Code	Error Name	Description
-1073807333	VI_ERROR_INV_DEGREE	Specified degree is invalid.
-1073807328	VI_ERROR_INV_LOCK_TYPE	The specified type of lock is not supported by this resource.
-1073807327	VI_ERROR_INV_ACCESS_KEY	The access key to the specified resource is invalid.
-1073807312	VI_ERROR_ABORT	User abort occurred during transfer.
-1073807301	VI_ERROR_QUEUE_ERROR	Unable to queue the asynchronous operation.
-1073807254	VI_ERROR_ASRL_PARITY	A parity error occurred during transfer.
-1073807253	VI_ERROR_ASRL_FRAMING	A framing error occurred during transfer.
-1073807252	VI_ERROR_ASRL_OVERRUN	An overrun error occurred during transfer. A character was not read from the hardware before the next character arrived.
-1073807240	VI_ERROR_INV_PARAMETER	The value of some parameter (which parameter is not known) is invalid.
-1073807229	VI_ERROR_INV_LENGTH	Invalid length specified.
-1073807204	VI_ERROR_SESN_NLOCKED	The current session did not have a lock on the resource.
1073676290	VI_SUCCESS_EVENT_EN	Specified event is already enabled for at least one of the specified mechanisms.
1073676291	VI_SUCCESS_EVENT_DIS	Specified event is already disabled for at least one of the specified mechanisms.
1073676292	VI_SUCCESS_QUEUE_EMPTY	Operation completed successfully, but queue was already empty.
1073676293	VI_SUCCESS_TERM_CHAR	The specified termination character was read.
1073676294	VI_SUCCESS_MAX_CNT	The number of bytes transferred is equal to the input count.
1073676416	VI_SUCCESS_QUEUE_NEMPTY	Wait terminated successfully on receipt of an event notification. There is at least one more event occurrence of the type specified by inEventType available for this session.
1073676420	VI_WARN_NSUP_ATTR_STATE	Although the specified state of the attribute is valid, it is not supported by this resource implementation.
1073676421	VI_WARN_UNKNOWN_STATUS	The status code passed to the operation could not be interpreted.
1073676424	VI_WARN_NSUP_BUF	The specified I/O buffer is not supported.

<b>Error Code</b>	<b>Error Name</b>	<b>Description</b>
1073676441	VI_SUCCESS_NESTED_SHARED	Operation completed successfully, and this session has nested shared locks.
1073676442	VI_SUCCESS_NESTED_EXCLUSIVE	Operation completed successfully, and this session has nested exclusive locks.
1073676443	VI_SUCCESS_SYNC	Operation completed successfully, but the operation was actually synchronous rather than asynchronous.
-1073807278	VI_ERROR_INV_WIDTH	Invalid access width specified.
-1073807275	VI_ERROR_NSUP_VAR_WIDTH	Cannot support source and destination widths that are different.
-1073807248	VI_ERROR_NSUP_ALIGN_OFFSET	The specified offset is not properly aligned for the access width of the operation.
-1073807247	VI_ERROR_USER_BUF	A specified user buffer is not valid or cannot be accessed for the required size.
-1073807246	VI_ERROR_RSRC_BUSY	The resource is valid, but VISA cannot currently access it.
1073676407	VI_WARN_CONFIG_NLOADED	The specified configuration either does not exist or could not be loaded. VISA-specified defaults will be used.
1073676413	VI_SUCCESS_DEV_NPRESENT	Session opened successfully, but the device at the specified address is not responding.
1073676418	VI_WARN_NULL_OBJECT	The specified object reference is uninitialized.
-1073807313	VI_ERROR_NENABLED	You must be enabled for events of the specified type in order to receive them.
-1073807202	VI_ERROR_LIBRARY_NFOUND	A code library required by VISA could not be located or loaded.
-1073807271	VI_ERROR_RESP_PENDING	A previous response is still pending, causing a multiple query error.
-1073807263	VI_ERROR_NSYS_CNTL	The interface associated with this session is not the system controller.
1073676440	VI_SUCCESS_NCHAIN	Event handled successfully. Do not invoke any other handlers on this session for this event.



321778A-01

Jan98