LabVIEW™ Application Builder

Version 6.1

The LabVIEW Application Builder is an add-on package you can use to create stand alone applications and shared libraries (DLLs) with LabVIEW. Additionally, you can distribute these applications and shared libraries without the LabVIEW development system. Refer to the *National Instruments Software License Agreement* for the licensing requirements for distributing executables.

These release notes contain installation instructions and describe the system requirements for applications created with this version of Application Builder. You must use Application Builder 6.1 with LabVIEW 6.1.

Contents

LabVIEW™, National Instruments™, NI™, and ni.com™ are trademarks of National Instruments Corporation. Product and company names mentioned herein are trademarks or trade names of their respective companies.

 © 1993, 2001 National Instruments Corp. All rights reserved.

Required System Configuration

Applications and shared libraries you create with the Application Builder have the same approximate requirements as the LabVIEW development system. Memory requirements depend on the size of your application or shared library. Typically, applications and shared libraries require about the same amount of memory it takes to run VIs in the development system.

LabVIEW applications and shared libraries use a directory for storing temporary files. Some temporary files are large, and it is best if several megabytes of disk space are available for this temporary directory. You can view or change the temporary directory by selecting **Tools»Options** and selecting **Paths** from the top pull-down menu.

If your application or shared library aborts unexpectedly, it might leave files in the temporary directory. Remove old files to free disk space.

UNIX

LabVIEW applications and shared libraries that display front panels require an X Windows System server, such as OpenWindows 3, HP-VUE, or X11R6. These applications and shared libraries do not require a specific graphical user interface (GUI) such as Motif or OpenLook because LabVIEW uses `Xlib` to create its own GUI.

(Sun) LabVIEW Application Builder for Sun runs on Solaris 2.5.1 or later.

(Linux) LabVIEW for Linux runs on Linux/*x*86 computers with RedHat Linux 5.0 or later. LabVIEW Application Builder for Linux requires Linux for Intel *x*86 processors with kernel version 2.0.*x* or later. LabVIEW runs on most major Linux distributions, such as RedHat, Calderra, SuSE, and Debian.

You need a minimum of 32 MB of RAM and at least 32 MB of swap space storage. You need between 65 MB to 150 MB of disk storage space depending on the components you install.

Macintosh

To build a shared library, you must have Macintosh Programmer's Workshop (MPW) installed. MPW must have an MrC compiler located in its tools directory. The MPW installed by Metrowerks does not have this compiler. Refer to the Apple Developer Connection at `developer.apple.com` to download the correct version.

To call a LabVIEW-built shared library from a CodeWarrior application, you must have enough memory allocated to that application. Because the application uses the LabVIEW Run-Time Engine that is about 4 MB itself, you should allocate at least 5 MB for the applications that you build. The default amount of memory is 384 KB and the error that results when you do not allocate enough memory does not indicate the source of the problem.

Installing the LabVIEW Application Builder Libraries

The default installation for the LabVIEW Professional Development System includes the Application Builder. Complete the following steps to install the Application Builder if you purchased it separately.

Windows

Complete the following steps to install the Application Builder.

Note Some virus detection programs interfere with the installer program. Check the distribution media for viruses before you begin installation. Then, disable automatic virus detection and run the installer. After installation, check your hard disk for viruses again and enable virus detection.

1. Insert the installation CD.
2. Run `setup.exe`.
3. Change the path, if necessary, to your LabVIEW directory, and click the **Install** button.

Macintosh

Complete the following steps to install the Application Builder.

Note Some virus detection programs interfere with the installer program. Check the distribution media for viruses before you begin installation. Then, disable automatic virus detection and run the installer. After installation, check your hard disk for viruses again and enable virus detection.

1. Insert the installation CD.
2. Double-click the **LabVIEW AppLibs Installer** icon.
3. After you click the **Install** button, the installer prompts you to select a destination folder. Select your LabVIEW folder.

Note To build a shared library with LabVIEW, you *must* have a full installation of MPW including ToolServer on your computer. It is available as a free download from the Apple

Developer Connection at `developer.apple.com`. It is *not* sufficient to have an installation of the Codewarrior MPW. It does not have the necessary set of tools.

UNIX

Complete the following steps to install the Application Builder on Linux or Sun. Root privileges are not necessary to install these libraries, but you must be able to write to the LabVIEW directory where you want to install these libraries.

1. Mount the installation CD.
2. Enter the following UNIX command for your operating system:
   - **(Linux)**

     ```
     cd /mnt/cdrom/linux
     ```

     where `cdrom` is the directory where you mounted the CD.
   - **(Solaris 2)**

     ```
     volcheck
     cd /cdrom/cdrom0/solaris2
     ```

     where `cdrom` is the directory where you mounted the CD.
3. Run the installation program by entering the following command:

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

Verifying Installation of Application Builder

Launch LabVIEW after you install the Application Builder and select **Tools»Build Application or Shared Library (DLL)**. Verify that your LabVIEW directory contains an `AppLibs` directory. If this directory is not present, reinstall the Application Builder Libraries according to the *Installing the LabVIEW Application Builder Libraries* section.

If the libraries are installed correctly, the `examples` directory contains an `appbuild.llb` example.

Frequently Asked Questions

The following section contains answers to frequently asked questions about the LabVIEW Application Builder.

Where can I find the LabVIEW Run-Time Engine installer?

On the LabVIEW CD, the LabVIEW Run-Time Engine installer is in the `LVRunTimeEng` directory. **(Macintosh and UNIX)** The Run-Time Engine installer is in the `redist\runTime` directory. If you purchased the Application Builder separately, the Run-Time Engine installer also is on the Application Builder CD in the same locations.

You also can download the Run-Time Engine installer from the National Instruments Web site at `ni.com`.

Can I copy lvrt.dll into the same directory as my executable, instead of installing the Run-Time Engine?

When the Run-Time Engine is installed, `lvrt.dll` (the engine DLL) and its subdirectories are copied to `Program Files\National Instruments\ shared\LabVIEW Run-Time\Version` by default. Copying the files and its subdirectories into the same directory as your executable suffices because the executable looks in its directory when searching for the engine DLL. You lose some modularity with this method because each executable needs a duplicate of `lvrt.dll` in its directory. If you use this option, include these files and its subdirectories as support files in your build.

How do I include a custom About screen with my application?

Write a VI with a front panel that describes your application. The name of the VI must start with `About`. Include this VI as a dynamic VI when you build the application. This enables an **About** option in the **Help** menu of your executable to run your About VI.

Why doesn't my application use the custom icon I specified?

LabVIEW can import black-and-white and 16-color icons in each of two resolutions, 16×16 and 32×32, for a total of four possible icons. When the icon file does not contain all four icons, LabVIEW uses its default icon for the missing icon. Avoid this by using the Icon Editor to specify all icon types.

Changes to the Application Builder Libraries

The following sections describe the changes introduced in the most recent versions of the Application Builder.

Changes Introduced between Versions 6.0 and 6.1

The following list describes the changes between versions 6.0 and 6.1:

- The **Destination** dialog box is two separate dialog boxes. The first dialog box, **Build Destination Settings**, accessible from the **Source Files** tab, describes the file-by-file build destination settings. The second dialog box, **Installer File Settings**, accessible from the **Installer Settings** tab, describes the file-by-file installer settings.
- A **Most Recently Used** build script pull-down menu is located next to the load button.
- **(Windows)** LabVIEW generates Microsoft Installer (MSI) installers. You can select from 10 installation directories.
- **(Windows)** The **Media Size** and **Extra Space on first disk (kB)** buttons were removed from the **Installer Settings** tab because the Microsoft Installer (MSI) does not support disk spanning.
- **(Windows)** The **Create Uninstaller** option was removed from the **Advanced Installer Settings** dialog box because the Application Builder always creates an uninstaller. Additionally, there are checkboxes that you can use to install only the parts of the LabVIEW Run-Time Engine that you want.
- To control your application remotely, you must include the NI License Manager utility in your installer by clicking the **Advanced** button in the **Installer Settings** tab and placing a checkmark in the **Remote Panel License Support** checkbox. For more information about the NI License Manager, select **Start»Programs» National Instruments»LabVIEW 6.1»NI License Manager Utility**.
- **(Windows)** The number of languages that you can create installers for has decreased from 12 to 4 (English, French, German, and Japanese).
- If you are developing an application or shared library, you must distribute any relevant user-defined or third-party error code text files with the application or shared library. If you use the Application Builder to create an installer for the application or shared library, the Application Builder prompts you to select the error code text files you want to distribute. The Application Builder also configures the installation subdirectory for the files.

  If you use a third-party utility to create an installer, locate the error code text files you want to distribute and configure the proper installation subdirectory for the files. User-defined error code text

files are located in `labview\user.lib\errors`, and third-party files are located in `labview\projects\errors`. If you are creating an installer for an application, install the error code text files in a `user.lib\errors` subdirectory of the application directory. If you are creating an installer for a shared library, install the error code text files in the `National Instruments\shared\errors directory`. Each of these directories also can contain a language subfolder with translated error codes.

Changes Introduced between Versions 5.1.1 and 6.0

The following list describes the changes between versions 5.1.1 and 6.0:

- You can use the Application Builder to build shared libraries (DLLs).
- **(Windows)** The files necessary for the 3D Graph, DataSocket, and NI Reports are part of the LabVIEW Run-Time Engine installation. When you create an installer, you only select whether to install the LabVIEW Run-Time Engine.

Changes Introduced between Versions 5.1 and 5.1.1

The following list describes the changes between versions 5.1 and 5.1.1:

- Previously, the Application Builder generated errors if you tried to open script files that moved after you created them. In 5.1.1, you can move your build script files after you create them.
- The Application Builder does not encounter problems with DLLs referenced by name.
- When adding support files to an application, you can add the contents of an entire directory using the **Add Directory Contents** button on the **Source Files** tab of the **Application Builder** dialog box.
- You can add an entire VI library (`.llb`) to an application as a support file, whereas before, you only could add files from a VI library. You access this option through the **Source Files** tab.
- You can select more than one item in listboxes under the **Source Files** tab and in the **Custom Destinations** dialog box. You can remove files easily and set destination settings easily.
- Installer settings allow you to set the **Installation destination** and the **Relative path** within that destination's directory. The only place you can install the executable for the application is the installation directory.
- The list of VIs in the **VI Settings** tab is alphabetized.

- **(Windows)** You can specify the directory in which to create the installer.
- **(Windows)** If the ActiveX server is enabled for the application, the type library is added automatically to the destinations file list. You cannot modify anything about the build or installation destination of the type library. If the ActiveX server is enabled, you cannot modify the **Replace existing files** installer setting for the executable because the executable version *must* match the type library version.

Changes Introduced between Version 5.0 and 5.1

In LabVIEW 5.1, the process for building an application is streamlined. Previously, you had to save your VIs to a library, then build an application using the **Build Application** dialog box. Further, to build an installer in Windows you had to use the **Create Distribution Kit** dialog box.

In LabVIEW 5.1, you use the **Build Application** dialog box to do all of these operations. You configure the application to various settings within the tabs on the **Build Application** dialog box. After you define these settings, you save them in a script so that you can easily rebuild the application.

321809E-01 Nov01