LABVIEW™ APPLICATION BUILDER

Version 6.0

The LabVIEW Application Builder is an add-on package you can use to create programs and shared libraries (DLLs) with LabVIEW. Additionally, you can distribute these programs and shared libraries without the LabVIEW development software. Refer to the *LabVIEW 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 the Application Builder. You must use the Application Builder 6.0 with LabVIEW 6.0.

Contents

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**.

If your application or shared library aborts unexpectedly, it might leave files in the temporary directory. Remove old files to avoid using up 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.

LabVIEW Application Builder for Sun runs on Solaris 2.5.1 or later. LabVIEW for HP-UX runs on Hewlett-Packard Model 9000 Series 700 computers with HP-UX 10.20 or later. LabVIEW for Linux runs on Linux/x86 computers with RedHat Linux 5.0 or later.

LabVIEW Application Builder for Linux requires Linux for Intel x86 processors with kernel version 2.0.x or later. LabVIEW runs on most major Linux distributions, such as RedHat, Calderra, SuSE, and Debian.

You will need a minimum of 32 MB of RAM, with 32 MB or more of swap space storage. You need between 65 MB to 150 MB of disk storage space depending on the components you install.

Installing the LabVIEW Application Builder Libraries

The Application Builder is part of the default installation for the LabVIEW Professional Development System configuration. Complete the following installation steps if you purchased the Application Builder 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, turn off the automatic virus checker and run the installer. After installation, check your hard disk for viruses again and turn on the virus checker.

1. Insert the CD.
2. Run `setup.exe`.
3. Change the path, if necessary, to point to your LabVIEW directory, then 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, turn off the automatic virus checker and run the installer. After installation, check your hard disk for viruses again and turn on the virus checker.

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

UNIX

Complete the following steps to install the Application Builder on HP-UX, 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 plan to install these libraries.

1. Mount the CD.

2. Type the following UNIX command for your operating system.

   - **(HP-UX)**

     ```
     cd /cdrom/HP-UX
     ```

     where /cdrom is the directory where you have mounted the CD-ROM.

   - **(Linux)**

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

     where /cdrom is the directory where you have mounted the CD-ROM.

   - **(Solaris 2)**

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

     where /cdrom is the directory where you have mounted the CD-ROM.

3. Run the installation program by typing 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 choose **Tools»Build Application or Shared Library (DLL)**. Verify that your LabVIEW directory contains an AppLibs directory. If this directory is not present, reintsall the Application Builder Libraries according to the preceding steps.

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

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 Version 5.1.1 and 6.0

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

- You now can use the Application Builder to build shared libraries (DLLs).

- **(Windows)** The files necessary for the 3D Graph, DataSocket, and NI Reports are now part of the LabVIEW Run-Time Engine Installation. When you create an installer, you now only choose whether to install the LabVIEW Run-Time Engine.

Changes Introduced between Version 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 had moved since you created them. In 5.1.1, you move your build script files after you create them.

- The Application Builder encountered 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 could only 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 now 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 automatically added 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

The following list describes the changes between versions 5.0 and 5.1.

- In LabVIEW 5.1, the process for building an application was 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.

321809D-01   Jul00