

LABVIEW™ APPLICATION BUILDER

Version 6.1

LabVIEW Application Builder ist ein Zusatzpaket, das Sie verwenden können, um mit LabVIEW selbständig (ohne Entwicklungsumgebung) ausführbare Programme und Shared Libraries (DLLs) zu erstellen. Zusätzlich können Sie diese Anwendungen und Shared Libraries ohne die LabVIEW-Entwicklungssoftware vertreiben. In der *LabVIEW-Softwarelizenzvereinbarung* finden Sie die Lizenzbedingungen, die für den Vertrieb von Programmen gelten.

In diesen Versionshinweisen finden Sie Installationsanweisungen. Darüber hinaus werden die Systemanforderungen für Applikationen beschrieben, die mit dieser Version von Application Builder erstellt werden. Sie müssen den Application Builder 6.1 installiert haben, um ihn mit LabVIEW 6.1 verwenden zu können.

Inhaltsverzeichnis

Erforderliche Systemkonfiguration	2
UNIX	2
Macintosh.....	3
Installieren der LabVIEW Application Builder-Bibliotheken	3
Windows	3
Macintosh.....	4
UNIX	4
Prüfen der Application Builder-Installation.....	5
Häufig gestellte Fragen	5
Wo befindet sich der Installer der LabVIEW Run-Time Engine?... 5	
Ist es möglich, die Run-Time-Engine nicht zu installieren und statt dessen die lvrt.dll mit der Anwendung ins gleiche Verzeichnis zu legen?	5
Wie kann ich eine benutzerdefiniertes Informationsfenster (Über...) zu meiner Applikation hinzufügen?	6
Warum zeigt meine Applikation nicht' das benutzerdefinierte Symbol, das ich erstellt habe?.....	6

LabVIEW™, National Instruments™, NI™ und ni.com™ sind Warenzeichen von National Instruments Corporation. Die Produkt- und Firmennamen sind Warenzeichen oder Handelsbezeichnungen der jeweiligen Unternehmen. Patent-Informationen für National Instruments Produkte erhalten Sie auf folgende Weise: Über die Menüoption **Hilfe>>Patente** in Ihrer Software, in der Datei `patents.txt` auf der jeweiligen CD und/oder im Internet unter www.ni.com/patents.

Änderungen an den Application Builder-Bibliotheken	6
Änderungen zwischen Version 6.0 und 6.1	6
Änderungen zwischen Version 5.1.1 und 6.0	8
Änderungen zwischen Version 5.1 und 5.1.1	8
Änderungen zwischen Version 5.0 und 5.1	9

Erforderliche Systemkonfiguration

Für die Applikationen und DLLs, die Sie mit Application Builder erstellen, gelten etwa dieselben Anforderungen wie für das LabVIEW-Entwicklungssystem. Die Speichieranforderungen hängen von der Größe der Applikation oder der DLL ab. Typischerweise ist für Applikationen und DLLs etwa derselbe Speicherplatz erforderlich, der auch zum Ausführen von VIs im Entwicklungssystem benötigt wird.

Für LabVIEW-Applikationen und –DLLs wird ein Verzeichnis zum Speichern temporärer Dateien verwendet. Einige temporäre Dateien sind relativ groß, daher sollten Sie am besten einige Megabyte Speicherplatz für das Verzeichnis bereitstellen, in dem temporäre Dateien gespeichert werden. Sie können das temporäre Verzeichnis wechseln, indem sie über **Werkzeuge»Optionen** die Option **Pfade** aus dem Pull-Down-Menü auswählen.

Wenn eine Applikation oder Shared Library/DLL unerwartet beendet wird, können Dateien im Verzeichnis für die temporären Dateien zurückbleiben. Entfernen sie alte Dateien, um Speicherplatz auf dem Datenträger freizugeben.

UNIX

Für LabVIEW-Applikationen und Shared Libraries, mit denen Frontpanels angezeigt werden, ist ein X Windows System-Server wie OpenWindows 3, HP-VUE oder X11R6 erforderlich. Für diese Applikationen und Shared Libraries ist keine spezielle grafische Benutzeroberfläche (GUI), zum Beispiel Motif oder OpenLook, erforderlich, weil LabVIEW mit `xlib` eine eigene GUI erzeugt.

(Sun) LabVIEW Application Builder für Sun läuft unter Solaris 2.5.1 oder höher.

(Linux) LabVIEW für Linux läuft auf Linux/ x86-Computern unter RedHat Linux 5.0 oder höher. LabVIEW Application Builder für Linux benötigt Linux für Intel x86-Prozessoren mit Kernel-Version 2.0.x oder höher. LabVIEW läuft unter den bekannten Linux-Distributionen wie RedHat, Caldera, SuSE und Debian.

Sie benötigen mindestens 32 MB RAM und mindestens 32 MB Swap-Speicherplatz. In Abhängigkeit von den Komponenten, die Sie installieren, sind zwischen 65 MB und 150 MB Festplattenspeicherplatz erforderlich.

Macintosh

Um eine Shared Library zu erstellen, müssen Sie Macintosh Programmer's Workshop (MPW) installiert haben. MPW muss ausserdem in seinem Tools-Verzeichnis einen MrC Compiler enthalten. Der von Metrowerks installierte MPW enthält diesen Compiler nicht. In der Apple Developer Connection unter developer.apple.com können Sie die richtige Version herunterladen.

Wenn Sie eine in LabVIEW erstellte Shared Library mit einer CodeWarrior-Anwendung aufrufen möchten, müssen Sie für diese Anwendung genügend Speicherplatz allokkieren. Die Anwendung benutzt die LabVIEW Run-Time-Engine, die ungefähr 4 MB groß ist, deshalb sollten Sie mindestens 5 MB für Ihre Applikation zur Verfügung stellen. Die Voreinstellung für bereitgestellten Speicher ist 384 KB und der Fehler, der bei zu wenig allokiertem Speicher auftritt, weist nicht auf die Quelle des Problems hin.

Installieren der LabVIEW Application Builder-Bibliotheken

Die Standardinstallation des LabVIEW Professional Development Systems enthält den Application Builder. Führen Sie die nachfolgenden Schritte durch, um den Application Builder zu installieren, wenn Sie ihn separat erworben haben.

Windows

Führen Sie die folgenden Schritte durch, um den Application Builder zu installieren.



Hinweis Einige Virenschutzprogramme stören das Installationsprogramm. Prüfen Sie den Vertriebsdatenträger auf Viren, bevor Sie mit der Installation beginnen. Deaktivieren Sie dann die automatische Virenerkennung und starten Sie den Installer. Prüfen Sie die Festplatte nach der Installation erneut auf Viren und aktivieren Sie dann die Virenerkennung.

1. Legen Sie die CD ein.
2. Führen Sie `setup.exe` aus.

3. Ändern Sie gegebenenfalls den Pfad auf das entsprechende LabVIEW-Verzeichnis und klicken Sie anschließend auf die Schaltfläche **Install**.

Macintosh

Führen Sie die folgenden Schritte durch, um den Application Builder zu installieren.



Hinweis Einige Virenschutzprogramme stören das Installationsprogramm. Prüfen Sie den Vertriebsdatenträger auf Viren, bevor Sie mit der Installation beginnen. Deaktivieren Sie dann die automatische Virenerkennung und starten Sie den Installer. Prüfen Sie die Festplatte nach der Installation erneut auf Viren und aktivieren Sie dann die Virenerkennung.

1. Legen Sie die CD ein.
2. Doppelklicken Sie auf das Symbol **LabVIEW AppLibs Installer**.
3. Nachdem Sie auf die Schaltfläche **Install** geklickt haben, werden Sie aufgefordert, einen Zielordner auszuwählen. Wählen Sie den LabVIEW-Ordner aus.



Hinweis Um eine Shared Library mit LabVIEW zu erstellen, *müssen* Sie eine vollständige Installation des MPW, inklusive des ToolServers auf Ihrem Rechner haben. Das Programm ist als kostenloser Download in der Apple Developer Connection unter `developer.apple.com` erhältlich. Eine Installation des Code Warrior MPW ist *nicht* ausreichend. Dieser enthält nicht die notwendigen Werkzeuge.

UNIX

Führen Sie die folgenden Schritte durch, um den Application Builder unter Linux oder Sun zu installieren. Es sind keine Root-Benutzerrechte erforderlich, um diese Bibliotheken zu installieren, aber Sie müssen über die Schreibberechtigung für das LabVIEW-Verzeichnis verfügen, in dem diese Bibliotheken installiert werden sollen.

1. Mounten Sie die CD.
2. Geben Sie den folgenden UNIX-Befehl für das Betriebssystem ein.

- **(Linux)**

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

wobei /cdrom das Verzeichnis ist, in dem Sie die CD-ROM gemountet haben.

- **(Solaris 2)**

```
volcheck
```

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

wobei /cdrom das Verzeichnis ist, in dem Sie die CD-ROM gemountet haben.

3. Führen Sie das Installationsprogramm aus, indem Sie den folgenden Befehl eingeben:
`./INSTALL`
4. Befolgen Sie die Anweisungen auf dem Bildschirm.

Prüfen der Application Builder-Installation

Starten Sie LabVIEW, nachdem Sie Application Builder installiert haben, und klicken Sie auf **Werkzeuge»Applikation oder “Shared Library” (DLL) erzeugen**. Überzeugen Sie sich, dass das Verzeichnis LabVIEW ein Verzeichnis `AppLibs` enthält. Wenn dieses Verzeichnis nicht vorhanden ist, installieren Sie die Application Builder-Bibliotheken erneut, wie unter [Installieren der LabVIEW Application Builder-Bibliotheken](#) beschrieben.

Wenn die Bibliotheken ordnungsgemäß installiert sind, enthält das Verzeichnis `examples` das Beispiel `appbuild.llb`.

Häufig gestellte Fragen

Das folgende Kapitel enthält Antworten zu häufig gestellten Fragen über den LabVIEW Application Builder.

Wo befindet sich der Installer der LabVIEW Run-Time Engine?

Auf der LabVIEW CD, der Installer der LabVIEW Run-Time Engine befindet sich in dem Verzeichnis `LVRunTimeEng`. (**Macintosh and UNIX**) Der Installer der Run-Time-Engine befindet sich in dem Verzeichnis `redist\runTime`. Falls Sie den Application Builder separat erworben haben, befindet sich der Installer der Run-Time-Engine auf der Application Builder CD an der selben Stelle.

Sie können den Installer der Run-Time-Engine auch von der National Instruments Web-Seite unter `ni.com` herunterladen.

Ist es möglich, die Run-Time-Engine nicht zu installieren und statt dessen die `lvrt.dll` mit der Anwendung ins gleiche Verzeichnis zu legen?

Wenn die Run-Time-Engine installiert ist, dann wurden die `lvrt.dll` (die DLL der Engine) und ihre Unterverzeichnisse per Voreinstellung nach `Programme\National Instruments\shared\LabVIEW Run-Time\Version` kopiert. Kopieren der Dateien und ihrer Unterverzeichnisse in das gleiche Verzeichnis der Applikation genügt, da die Applikation im eigenen Verzeichnis nach der DLL der

Run-Time-Engine sucht. Allerdings verlieren Sie mit dieser Methode Modularität, da jede Anwendung ein eigenes Duplikat der `lvrt.dll` im eigenen Verzeichnis benötigt. Falls Sie diese Option nutzen möchten, binden Sie die entsprechenden Dateien und ihre Unterverzeichnisse als Support-Dateien zur Anwendung hinzu.

Wie kann ich eine benutzerdefiniertes Informationsfenster (Über...) zu meiner Applikation hinzufügen?

Schreiben Sie ein VI mit einem Frontpanel, das Ihre Applikation beschreibt. Der Name des VIs muss mit `About` beginnen. Binden Sie dieses VI als ein Dynamisches VI ein, wenn Sie die Applikation erstellen. Dies stellt die **About** Option im **Hilfe** Menü Ihrer Applikation so ein, dass Ihr Informations-VI ausgeführt wird, wenn Sie `Über...` anklicken.

Warum zeigt meine Applikation nicht' das benutzerdefinierte Symbol, das ich erstellt habe?

LabVIEW kann schwarz-weiße und 16-farbige Symbole importieren, jeweils mit einer Auflösung von 16×16 und 32×32. Dies ergibt 4 mögliche Symbole. Wenn die Symboldatei nicht alle vier Symbole enthält, benutzt LabVIEW für die fehlenden Symbole die Standardsymbole. Mit einem Symbol-Editor können Sie alle Symbol-Typen definieren und dies somit unterbinden.

Änderungen an den Application Builder-Bibliotheken

In den folgenden Abschnitten werden die Änderungen beschrieben, die in die neuesten Application Builder-Versionen eingeflossen sind.

Änderungen zwischen Version 6.0 und 6.1

In der folgenden Liste werden die Änderungen zwischen Version 6.0 und 6.1 beschrieben:

- Das **Ziel** Dialogfenster besteht jetzt aus zwei getrennten Dialogfenstern. Das erste Dialogfenster, **Einstellungen für Build-Ziel**, unter dem **Quelldatei**-Reiter, beschreibt die Build-Ziele der einzelnen Dateien. Das zweite Dialogfenster, **Einstellungen für Installer Dateien**, unter dem **Installer Einstellungen**-Reiter, beschreibt die Installer Einstellungen der einzelnen Dateien.
- An der Laden-Schaltfläche befindet sich ein Kurzwahl-Menü zur Auswahl der **am häufigsten genutzten** Build-Skripte.
- **(Windows)** LabVIEW erstellt Microsoft Installer (MSI). Sie können 10 verschiedene Installer-Verzeichnisse auswählen.

- **(Windows)** Die **Datenträgergröße** und **Platz auf erster Diskette reservieren (kB)**-Schaltflächen wurden aus den **Installer Einstellungen** entfernt, weil der Microsoft Installer (MSI) kein Disk-Spanning unterstützt.
- **(Windows)** Die Option **Erstelle Uninstaller** wurde von dem Menü **Fortgeschrittene Installer Einstellungen** entfernt, weil der Application Builder jetzt immer einen Uninstaller erstellt. Weiterhin gibt es Auswahloptionen, mit denen Sie bestimmen können, welche Teile der LabVIEW Run-Time-Engine Sie installieren möchten.
- Um die Anwendung netzwerkfähig zu machen, müssen Sie die NI License Manager utility in den Installer integrieren. Klicken Sie dazu **Fortgeschritten** in dem Reiter mit den **Installer Einstellungen** und setzen Sie den Haken für **Unterstützung für lizenzierte netzsteuerbare Panels** . Weitere Informationen zum NI Licence Manager bekommen Sie unter **Start»Programme»National Instruments»LabVIEW 6.1»NI Licence Manger Utility**.
- **(Windows)** Die Anzahl der Sprachen für die Erstellung der Installer wurde von 12 auf 4 herabgesetzt (Englisch, Französisch, Deutsch und Japanisch).
- Wenn Sie eine Anwendung oder Shared Library erstellen, müssen Sie alle relevanten benutzerdefinierten oder von Drittherstellern zur Verfügung gestellten Fehlercodes oder Textdateien mit der Applikation oder Shared Library zur Verfügung stellen. Wenn Sie den Application Builder einsetzen, um einen Installer für eine Anwendung oder eine Shared Library zu erstellen, dann fordert der Application Builder sie dazu auf, die Fehlercodes und Textdateien, die Sie vertreiben möchten, auszuwählen. Der Application Builder erstellt auch das Installations-Unterverzeichnis für diese Dateien.

Wenn Sie eine Applikation eines Drittherstellers verwenden, um einen Installer zu erstellen, finden Sie heraus, wo sich die Fehlercodes oder Textdateien befinden, die Sie vertreiben möchten und konfigurieren Sie die passenden Unterverzeichnisse für die Dateien.

Benutzerdefinierte Fehlercode-Textdateien finden sie unter `labview\user.lib\errors\`, die Dateien der Dritthersteller werden unter `labview\projects\errors` gespeichert. Wenn Sie einen Installer für eine Applikation erstellen, dann installieren Sie die Fehlercode-Textdateien in ein Unterverzeichnis von `user.lib\errors` des Applikationsverzeichnisses. Wenn Sie einen Installer für eine Shared Library erstellen, dann installieren sie die Fehlercode-Textdateien in das Verzeichnis `National Instruments\shared\errors`. Jedes dieser Verzeichnisse kann auch ein Sprach-Unterverzeichnis mit übersetzten Fehlercodes enthalten.

Änderungen zwischen Version 5.1.1 und 6.0

In der folgenden Liste werden die Änderungen zwischen Version 5.1.1 und 6.0 beschrieben.

- Sie können den Application Builder verwenden, um Shared Libraries (DLLs) zu erzeugen.
- **(Windows)** Die Dateien, die für 3D-Graphen, DataSocket sowie NI-Report erforderlich sind, sind jetzt Teil der Installation der LabVIEW Run-Time Engine. Wenn Sie ein Installationsprogramm erstellen, wählen Sie nur, ob die LabVIEW Run-Time Engine installiert wird.

Änderungen zwischen Version 5.1 und 5.1.1

In der folgenden Liste werden die Änderungen zwischen Version 5.1 und 5.1.1 beschrieben.

- Application Builder hat zuvor Fehlermeldungen angezeigt, wenn versucht wurde, Skriptdateien zu öffnen, die seit der Erstellung verschoben wurden. In Version 5.1.1 können Skriptdateien nun auch verschoben werden, nachdem sie erstellt wurden.
- Der Application Builder hat keine Probleme mit DLLs, die per Namen referenziert sind.
- Wenn einer Applikation Hilfsdateien hinzugefügt werden, können Sie den Inhalt eines gesamten Verzeichnisses mit Hilfe der Schaltfläche **Verzeichnisinhalt hinzufügen** auf der Registerkarte **Quelldateien** des Dialogfelds **Applikation oder Shared Library (DLL) erzeugen** hinzufügen.
- Sie können eine komplette VI-Bibliothek (.lib) in eine Applikation einfügen, während in früheren Versionen nur einzelne Dateien einer VI-Bibliothek als Hilfsdatei hinzugefügt werden konnten. Auf diese Option können Sie über die Registerkarte **Quelldateien** zugreifen.
- Sie können mehr als ein Element in den Listenfeldern auf der Registerkarte **Quelldateien** und im Dialogfeld **Zieleinstellungen** markieren. Dateien lassen sich schnell entfernen und Zieleinstellungen problemlos festlegen.
- Mit den Einstellungen des Installationsprogramms können Sie das **Installationsverzeichnis** und den **Relativen Pfad** im Zielverzeichnis festlegen. Die Applikation kann ausschließlich im Installationsverzeichnis installiert werden.
- Die Liste der VIs auf der Registerkarte **VI-Einstellungen** ist jetzt alphabetisch sortiert.
- **(Windows)** Sie können das Verzeichnis festlegen, in dem das Installationsprogramm generiert werden soll.

- **(Windows)** Wenn der ActiveX-Server für diese Applikation aktiviert ist, wird die Typbibliothek automatisch in die Dateiliste des Ziels eingefügt. Build oder Installationsziel der Typbibliothek können nicht geändert werden. Wenn der ActiveX-Server aktiviert ist, können Sie die Einstellung **Bestehende Dateien ersetzen** des Installationsprogramms für das ausführbare Programm nicht ändern, weil die ausführbare Version der Version der Typbibliothek entsprechen *muss*.

Änderungen zwischen Version 5.0 und 5.1

In LabVIEW 5.1 ist der Erstellungsprozess für Applikationen verbessert worden. Zuvor mussten VIs in einer Bibliothek gespeichert werden, und anschließend musste eine Applikation mit Hilfe des Dialogfelds **Build Application** erstellt werden. Um ein Installationsprogramm unter Windows zu erstellen, musste weiterhin das Dialogfeld **Create Distribution Kit** verwendet werden.

In LabVIEW 5.1 wird für alle diese Vorgänge nur das Dialogfeld **Build Application** verwendet. Die Applikation kann über die Registerkarten des Dialogfelds **Build Application** konfiguriert werden. Sobald Sie die Einstellungen festgelegt haben, können Sie diese als Skript speichern, damit Sie die Applikation ohne Schwierigkeiten später erneut erstellen können.