

L'APPLICATION BUILDER DE LABVIEW™

Version 6.1

L'Application Builder de LabVIEW est un module complémentaire que vous pouvez utiliser pour créer des exécutables et des bibliothèques partagées (DLL) avec LabVIEW. De plus, vous pouvez distribuer ces exécutables et ces bibliothèques partagées sans le logiciel de développement LabVIEW. Reportez-vous au *Contrat de Licence Logiciel National Instruments* en ce qui concerne les licences exigées pour distribuer des exécutables.

Ces notes d'information contiennent des instructions d'installation et décrivent les configurations système nécessaires pour les applications créées avec cette version de l'Application Builder. Vous devez utiliser l'Application Builder 6.1 avec LabVIEW 6.1.

Contenu

Configuration système nécessaire	2
UNIX	2
Macintosh.....	3
Installation des bibliothèques de l'Application Builder de LabVIEW ...	3
Windows	3
Macintosh.....	4
UNIX	4
Vérification de l'installation de l'Application Builder	5
Questions fréquemment posées.....	5
Où puis-je trouver le programme d'installation du moteur d'exécution de LabVIEW ?	5
Puis-je copier la DLL lvrtd.dll dans le même répertoire que mon exécutable au lieu d'installer le moteur d'exécution ?.....	5
Comment puis-je inclure une boîte de dialogue " À Propos... " personnalisée dans mon application ?	6
Pourquoi mon application n'utilise-t-elle pas l'icône personnalisée que j'ai spécifiée ?.....	6

Changements apportés aux bibliothèques de l'Application Builder	6
Différences entre la version 6.0 et la version 6.1	6
Différences entre la version 5.1.1 et la version 6.0	8
Différences entre la version 5.1 et la version 5.1.1	8
Différences entre la version 5.0 et la version 5.1	9

Configuration système nécessaire

Les exécutables et les bibliothèques partagées que vous créez avec l'Application Builder exigent approximativement la même configuration que le système de développement LabVIEW. Les besoins en mémoire dépendent de la taille de votre exécutable ou de votre bibliothèque partagée. En général, les exécutables et les bibliothèques partagées ont besoin d'une quantité de mémoire environ égale à celle qui est nécessaire pour exécuter les VIs dans le système de développement.

Les applications et les bibliothèques partagées de LabVIEW utilisent un répertoire pour stocker des fichiers temporaires. Certains fichiers temporaires ont une taille importante et il est préférable de disposer de plusieurs méga-octets d'espace disque pour ce répertoire temporaire. Vous pouvez visualiser ou changer le répertoire temporaire en sélectionnant **Outils»Options** et en sélectionnant **Chemins** à partir du menu déroulant.

En cas d'interruption inopinée de l'application ou de la bibliothèque partagée, il est possible qu'il reste des fichiers dans le répertoire temporaire. Supprimez les anciens fichiers pour libérer de l'espace disque.

UNIX

Les applications et les bibliothèques partagées de LabVIEW qui affichent des faces-avant requièrent un serveur X Windows System, comme OpenWindows 3, HP-VUE ou X11R6. Ces applications et bibliothèques partagées n'ont pas besoin d'interface utilisateur graphique spécifique (GUI), telle que Motif ou OpenLook, vu que LabVIEW utilise `xlib` pour créer sa propre interface utilisateur graphique.

(Sun) L'Application Builder de LabVIEW pour Sun fonctionne sous Solaris 2.5.1 ou version ultérieure.

(Linux) LabVIEW pour Linux fonctionne sur les machines Linux/x86 avec RedHat Linux 5.0 ou version ultérieure. L'Application Builder de LabVIEW pour Linux a besoin de Linux pour processeurs Intel x86 avec la version de noyau 2.0.x ou ultérieure. LabVIEW fonctionne sur la plupart des distributions majeures de Linux, telles que RedHat, Caldera, SuSE et Debian.

Vous avez besoin au minimum de 32 Mo de mémoire vive, avec 32 Mo ou plus d'espace de permutation de mémoire. Il vous faut entre 65 Mo et 150 Mo d'espace disque de stockage, en fonction des composantes que vous installez.

Macintosh

Pour construire une bibliothèque partagée (DLL), vous devez avoir Macintosh Programmer's Workshop (MPW) installé. MPW doit avoir un compilateur MrC situé dans son répertoire tools. Le MPW installé par Metrowerks ne contient pas ce compilateur. Reportez-vous au site Web d'Apple Developer Connection à l'adresse apple.com/developer pour télécharger la bonne version.

Pour appeler une bibliothèque partagée (DLL) construite avec LabVIEW à partir d'une application CodeWarrior, vous devez avoir alloué suffisamment de mémoire à cette application. Vu que l'application utilise le moteur d'exécution de LabVIEW qui possède lui-même une taille d'environ 4 Mo, vous devez disposer d'au moins 5 Mo pour l'application que vous construisez. La quantité de mémoire par défaut est de 384 Ko et l'erreur résultant d'un manque de mémoire allouée ne vous indique pas la source du problème.

Installation des bibliothèques de l'Application Builder de LabVIEW

Par défaut, l'installation de la version Système de développement professionnel de LabVIEW inclut l'Application Builder. Suivez la procédure d'installation ci-après si vous avez acheté l'Application Builder séparément.

Windows

Suivez la procédure d'installation ci-après pour installer l'Application Builder.



Remarque Certains programmes de détection de virus peuvent affecter le programme d'installation. Vérifiez que le support de distribution ne contient aucun virus avant de commencer l'installation. Ensuite, désactivez la détection automatique de virus et exécutez l'installateur. Après l'installation, vérifiez à nouveau l'absence de virus sur votre disque dur et activez le programme de détection de virus.

1. Insérez le CD d'installation.
2. Exécutez `setup.exe`.
3. Changez le chemin, si nécessaire, pour indiquer votre répertoire LabVIEW et cliquez sur le bouton **Installer**.

Macintosh

Suivez la procédure d'installation ci-après pour installer l'Application Builder.



Remarque Certains programmes de détection de virus peuvent affecter le programme d'installation. Vérifiez que le support de distribution ne contient aucun virus avant de commencer l'installation. Ensuite, désactivez la détection automatique de virus et exécutez l'installateur. Après l'installation, vérifiez à nouveau l'absence de virus sur votre disque dur et activez le programme de détection de virus.

1. Insérez le CD d'installation.
2. Double-cliquez sur l'icône **LabVIEW AppLibs Installer**.
3. Une fois que vous avez cliqué sur le bouton **Install**, vous êtes invité à sélectionner un dossier de destination. Sélectionnez votre dossier LabVIEW.



Remarque Pour construire une bibliothèque partagée (DLL) avec LabVIEW, vous *devez* avoir une installation complète de MPW incluant ToolServer sur votre machine. Vous pouvez le télécharger gratuitement à partir du site Web d'Apple Developer Connection à l'adresse apple.com/developer. Une installation de CodeWarrior MPW n'est *pas* suffisante car il ne contient pas le jeu d'outils nécessaire.

UNIX

Suivez la procédure ci-après pour installer l'Application Builder sur Linux ou Sun. Aucun privilège de racine n'est nécessaire à l'installation de ces bibliothèques, mais vous devez pouvoir écrire dans le répertoire LabVIEW où vous prévoyez d'installer ces bibliothèques.

1. Montez le CD d'installation.
2. Tapez la commande UNIX suivante pour votre système d'exploitation.
 - **(Linux)**

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

où `cdrom` est le répertoire de montage du CD.
 - **(Solaris 2)**

```
volcheck
```

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

où `cdrom` est le répertoire de montage du CD.
3. Exécutez le programme d'installation en tapant la commande suivante:

```
./INSTALL
```
4. Suivez les instructions qui apparaissent sur votre écran.

Vérification de l'installation de l'Application Builder

Lancez LabVIEW après avoir installé l'Application Builder et choisissez **Outils»Construire une application ou une DLL....** Vérifiez que votre répertoire LabVIEW contient un répertoire AppLibs. Si ce répertoire n'est pas présent, réinstallez les bibliothèques Application Builder en suivant la section [Installation des bibliothèques de l'Application Builder de LabVIEW](#).

Si les bibliothèques sont correctement installées, le répertoire `examples` contient un exemple `appbuild.lib`.

Questions fréquemment posées

La section suivante contient les réponses aux questions fréquemment posées à propos de l'Application Builder de LabVIEW.

Où puis-je trouver le programme d'installation du moteur d'exécution de LabVIEW ?

Sur le CD de LabVIEW, le programme d'installation du moteur d'exécution de LabVIEW se situe dans le répertoire `LVRunTimeEng`. **(Macintosh et UNIX)** Le programme d'installation du moteur d'exécution de LabVIEW se situe dans le répertoire `redist\runTime`. Si vous avez acheté l'Application Builder séparément, vous trouverez également sur le CD le programme d'installation du moteur d'exécution de LabVIEW dans les répertoires de mêmes noms.

Vous pouvez aussi télécharger le programme d'installation du moteur d'exécution de LabVIEW sur le site Web de National Instruments à l'adresse `ni.com`.

Puis-je copier la DLL `lvrt.dll` dans le même répertoire que mon exécutable au lieu d'installer le moteur d'exécution ?

Si le moteur d'exécution est installé, `lvrt.dll` (la DLL du moteur) et ses sous-répertoires sont copiés dans `Program Files\National Instruments\shared\LabVIEW Run-Time\Version` par défaut. Copier les fichiers et les sous-répertoires dans le même répertoire que votre exécutable suffit parce que ce dernier regarde dans son répertoire quand il recherche la DLL du moteur. Vous perdez une certaine modularité avec cette méthode car chaque exécutable nécessite une duplication de `lvrt.dll` dans son propre répertoire. Si vous utilisez cette option, incluez les fichiers et ses sous-répertoires comme fichiers support de votre application.

Comment puis-je inclure une boîte de dialogue “ À Propos... ” personnalisée dans mon application ?

Développez un VI avec une face-avant décrivant votre application. Le nom du VI doit commencer par `About`. Incluez ce VI comme un VI chargé dynamiquement quand vous construisez votre exécutable. Ceci active l’option **À Propos** dans le menu **Aide** de votre exécutable afin d’exécuter le VI `About`.

Pourquoi mon application n’utilise-t-elle pas l’icône personnalisée que j’ai spécifiée ?

LabVIEW peut importer des icônes en noir et blanc ou des icônes 16 couleurs dans chacune des deux résolutions 16×16 et 32×32, ce qui donne un total de quatre icônes possibles. Quand le fichier d’icônes ne contient pas les quatre icônes réunies, LabVIEW utilise son icône par défaut à la place de celle manquante. Évitez cela en utilisant l’éditeur d’icône pour spécifier tous les types d’icônes.

Changements apportés aux bibliothèques de l’Application Builder

Les sections suivantes décrivent les changements introduits dans les versions les plus récentes de l’Application Builder.

Différences entre la version 6.0 et la version 6.1

La liste suivante décrit les différences entre la version 6.0 et la version 6.1 :

- Les paramètres de **Destination** sont définis dans deux boîtes de dialogue. La première boîte de dialogue, intitulée **Paramètres de destination de la construction** et accessible à partir de l’onglet **Fichiers source**, décrit les paramètres de destination de la construction fichier par fichier. La seconde boîte de dialogue, intitulée **Paramètres de destination de l’installation** et accessible à partir de l’onglet **Paramètres installateur**, décrit les paramètres de l’installateur fichier par fichier.
- Un menu déroulant des scripts les plus récemment utilisés est situé à côté du bouton **Chargement**.
- **(Windows)** LabVIEW génère des programmes d’installation de type “Microsoft Installer” (appelés aussi “MSI”). Vous pouvez choisir parmi dix répertoires d’installation.

- **(Windows)** Les boutons **Taille du support** et **Espace supplémentaire sur le premier disque (ko)** sont retirés de l'onglet **Paramètres installateur** car Microsoft Installer ne supporte pas l'agrandissement de l'espace disque.
- **(Windows)** L'option **Créer un programme de désinstallation** a été retirée de la boîte de dialogue **Paramètres avancés de l'installateur** car l'Application Builder crée toujours un programme de désinstallation. De plus, il y a des cases à cocher que vous pouvez utiliser pour installer uniquement les composantes du moteur d'exécution de LabVIEW qui vous intéressent.
- Pour contrôler votre application à distance, vous devez inclure l'utilitaire NI License Manager Utility dans votre programme d'installation en cliquant sur le bouton **Avancé** dans l'onglet **Paramètres installateur** et en cochant la case **Support pour licence de face-avant distante**. Pour plus d'informations à propos de cet utilitaire, sélectionnez **Démarrer»Programmes»National Instruments»LabVIEW 6.1»NI License Manager Utility**.
- **(Windows)** Le nombre de langues pour lesquelles vous pouvez créer des programmes d'installation est passé de 12 à 4 (anglais, français, allemand et japonais).
- Si vous êtes en train de développer une application ou une bibliothèque partagée, vous devez distribuer tous les fichiers texte de codes d'erreur définis par l'utilisateur ou provenant d'une autre société. Si vous utilisez l'Application Builder pour créer un programme d'installation pour un exécutable ou une bibliothèque partagée, l'Application Builder vous invite à sélectionner les fichiers texte de codes d'erreur que vous souhaitez distribuer. L'Application Builder configure aussi le sous-répertoire d'installation pour ces fichiers.

Si vous utilisez un utilitaire tiers pour créer un programme d'installation, localisez les fichiers texte de codes d'erreur que vous souhaitez distribuer et configurez le sous-répertoire d'installation approprié pour ces fichiers. Les fichiers texte de codes d'erreur définis par l'utilisateur sont situés dans `labview\user.lib\errors` et les fichiers tiers sont situés dans `labview\projects\errors`. Si vous créez un programme d'installation pour une application, installez les fichiers texte de codes d'erreur dans le sous-répertoire `user.lib\errors` du répertoire de l'application. Si vous créez un programme d'installation pour une bibliothèque partagée, installez les fichiers texte de codes d'erreur dans le répertoire `National Instruments\shared\errors`. Chacun de ces répertoires peut aussi contenir un sous-répertoire de langue contenant les codes d'erreur traduits.

Différences entre la version 5.1.1 et la version 6.0

La liste suivante décrit les changements entre la version 5.1.1 et la version 6.0 :

- Vous pouvez utiliser l'Application Builder pour construire des bibliothèques partagées (DLL).
- **(Windows)** Les fichiers nécessaires au support de graphes 3D, connexions DataSocket et NI Reports font partie de l'installation du moteur d'exécution de LabVIEW. Lorsque vous créez un programme d'installation, vous choisissez uniquement si vous voulez installer le moteur d'exécution de LabVIEW.

Différences entre la version 5.1 et la version 5.1.1

La liste suivante décrit les changements entre la version 5.1 et la version 5.1.1 :

- Auparavant, l'Application Builder générait des erreurs si vous essayiez d'ouvrir des fichiers de script qui avaient été déplacés depuis leur création. Dans 5.1.1, vous pouvez déplacer vos fichiers de script de construction après les avoir créés.
- L'Application Builder ne rencontre pas de problème avec des DLL référencées par leur nom.
- Lorsque vous ajoutez des fichiers support à une application, vous pouvez ajouter le contenu d'un répertoire entier en utilisant le bouton **Ajouter le contenu d'un répertoire...** situé sur l'onglet **Fichiers source** de la boîte de dialogue **Construire une application ou une DLL**.
- Vous pouvez ajouter une bibliothèque entière de VIs (.lib) à une application en tant que fichier support, alors qu'auparavant vous ne pouviez ajouter que des fichiers provenant d'une bibliothèque de VIs. Vous accédez à cette option au moyen de l'onglet **Fichiers source**.
- Vous pouvez sélectionner plusieurs éléments dans les listes déroulantes situées sur l'onglet **Fichiers source** et dans la boîte de dialogue obtenue en cliquant sur le bouton **Destinations personnalisées...** Vous pouvez supprimer facilement des fichiers et définir aisément des paramètres de destination.
- Les paramètres de l'installateur vous permettent de définir la **Destination de l'installation** et le **Chemin relatif** au sein du répertoire de cette destination. Le seul endroit où vous pouvez installer l'exécutable pour l'application est le répertoire d'installation.
- La liste des VIs sous l'onglet **Paramètres VIs** est classée par ordre alphabétique.

- **(Windows)** Vous pouvez spécifier le répertoire dans lequel vous créez le programme d'installation.
- **(Windows)** Si le serveur ActiveX est activé pour l'application, la bibliothèque de types est automatiquement ajoutée à la liste des fichiers de destination. Vous ne pouvez en rien modifier la destination de la construction ou de l'installation de la bibliothèque de types. Si le serveur ActiveX est activé, vous ne pouvez pas modifier le paramètre du programme d'installation **Remplacer les fichiers existants** par l'exécutable car la version de ce dernier *doit* correspondre à la version de la bibliothèque de types.

Différences entre la version 5.0 et la version 5.1

Dans LabVIEW 5.1, le processus de construction d'une application est simplifié. Auparavant, vous deviez enregistrer vos VIs dans une bibliothèque, puis construire une application en utilisant la boîte de dialogue **Construction de l'application**. De plus, pour construire un programme d'installation sous Windows, vous deviez utiliser la boîte de dialogue **Créer un kit de distribution**.

Dans LabVIEW 5.1, vous utilisez la boîte de dialogue **Construction de l'application** pour effectuer toutes ces opérations. Vous configurez l'application à l'aide de divers paramètres sous les onglets de la boîte de dialogue **Construction de l'application**. Une fois que vous avez défini ces paramètres, vous les enregistrez dans un script, ce qui vous permet de reconstruire facilement l'application.