

# Guide de l'utilisateur pour l'Application Builder de LabVIEW™

Version 7.0

L'Application Builder de LabVIEW est un logiciel complémentaire qui sert à créer des applications autonomes et des bibliothèques partagées (DLLs) avec LabVIEW. Vous pouvez distribuer ces applications et bibliothèques partagées sans le système de développement LabVIEW. Reportez-vous à l'*Accord de licence logicielle National Instruments* qui figure sur les CD de distribution de la version PDS (Professional Development System) de LabVIEW et de l'Application Builder pour vous renseigner sur les obligations de licence qu'implique la distribution d'applications et de bibliothèques partagées.

Ce guide de l'utilisateur comprend des instructions sur l'installation de l'Application Builder, décrit la configuration système requise pour les applications créées avec cette version, ainsi que les modifications qui ont été apportées entre les versions précédentes et la version actuelle. Ce guide de l'utilisateur décrit également les mises en garde et les recommandations dont vous devez tenir compte lorsque vous créez un VI dans une application ou une bibliothèque partagée.

Vous devez utiliser l'Application Builder 7.0 avec LabVIEW 7.0.

---

## Pour en savoir plus...

Reportez-vous à l'*Aide LabVIEW* pour consulter des instructions pas à pas sur la création d'une application ou d'une bibliothèque partagée et les mises en garde à prendre en considération avant de construire une application ou une bibliothèque partagée.

---

---

LabVIEW™, National Instruments™, NI™, et ni.com™ sont des marques déposées de National Instruments Corporation. Les noms de produits et de sociétés cités sont des marques déposées par leurs propriétaires respectifs. Pour obtenir la liste des brevets protégeant les produits National Instruments, veuillez vous référer, selon le cas : à la rubrique **Aide»Brevets** de votre logiciel, au fichier `patents.txt` sur votre CD et/ou à [ni.com/patents](http://ni.com/patents).

Jun 2003  
323591A-01

[ni.com](http://ni.com)

© 2003 National Instruments Corp. Tous droits réservés. Printed in Ireland.



# Contenu

---

Configuration système requise .....	2
Mac OS X .....	2
Mac OS 9.x ou version antérieure .....	3
UNIX .....	3
Installation de l'Application Builder .....	3
Windows .....	3
Mac OS .....	4
UNIX .....	4
Vérification de l'installation de l'Application Builder .....	5
Mises en garde et recommandations lors de la construction d'applications et de bibliothèques partagées .....	5
Modifications apportées à l'Application Builder .....	7
Modifications apportées entre les versions 6.1 et 7.0 .....	7
Modifications apportées entre les versions 6.0 et 6.1 .....	8

## Configuration système requise

---

Les applications et les bibliothèques partagées que vous créez avec l'Application Builder exigent approximativement la même configuration système que le système de développement LabVIEW. Les exigences en mémoire dépendent de la taille de l'application ou de la bibliothèque partagée. Typiquement, les applications et les bibliothèques partagées exigent à peu près la même quantité de mémoire que pour exécuter des VIs dans le système de développement LabVIEW.

Les applications et les bibliothèques partagées LabVIEW utilisent un répertoire pour stocker les fichiers temporaires. Certains fichiers temporaires ont une taille importante, c'est pourquoi National Instruments vous recommande d'avoir plusieurs méga-octets d'espace disque disponibles pour ce répertoire temporaire. Vous pouvez afficher ou modifier ce répertoire temporaire en sélectionnant **Outils»Options** et en sélectionnant ensuite **Chemins** à partir du menu déroulant du haut.



**Remarque** Si l'application ou la bibliothèque partagée abandonne son exécution de façon soudaine, elle risque de laisser des fichiers dans le répertoire temporaire. Supprimez les anciens fichiers du répertoire temporaire pour libérer de l'espace disque.

## Mac OS X

Pour construire une bibliothèque partagée, vous devez avoir Developer Tools d'août 2002 ou version ultérieure, y compris Project Builder, installé. Reportez-vous au site Apple Developer Connection à [developer.apple.com](http://developer.apple.com) pour télécharger la version correcte.

## Mac OS 9.x ou version antérieure

Pour construire une bibliothèque partagée, vous devez avoir Macintosh Programmer's Workshop (MPW) installé, y compris ToolServer. MPW doit avoir un compilateur MrC dans son répertoire d'outils. Le MPW qui est installé par CodeWarrior ne possède pas ce compilateur. Reportez-vous au site Apple Developer Connection à [developer.apple.com](http://developer.apple.com) pour télécharger la version correcte.

Pour appeler une bibliothèque partagée LabVIEW à partir d'une application CodeWarrior, vous devez avoir alloué suffisamment de mémoire à cette application. Comme l'application utilise le moteur d'exécution de LabVIEW qui à lui seul accapare 4 Mo, vous devez allouer un minimum de 5 Mo aux bibliothèques partagées que vous construisez. Le volume de mémoire par défaut est de 384 Ko et l'erreur que vous obtenez si vous n'allouez pas suffisamment de mémoire n'indique pas d'où vient le problème.

## UNIX

Les applications et les bibliothèques partagées LabVIEW qui affichent des faces-avant nécessitent un serveur X Window, comme OpenWindows, CDE ou X11R6. Ces applications et bibliothèques partagées ne requièrent pas d'interface utilisateur graphique particulière (GUI) comme Motif ou OpenLook, car LabVIEW utilise `xlib` pour créer sa propre interface GUI.

**(Sun)** Application Builder pour Sun fonctionne sous Solaris 2.6 ou version ultérieure.

**(Linux)** Application Builder pour Linux fonctionne sous Linux pour les processeurs Intel x86 avec version de noyau 2.0.x, 2.2.x ou 2.4.x et la bibliothèque Run-Time GNU C 2.1.92 ou version ultérieure (`glibc`).

## Installation de l'Application Builder

---

L'installation par défaut de la version PDS de LabVIEW inclut l'Application Builder. Suivez les étapes suivantes pour installer l'Application Builder si vous l'avez acheté séparément.

## Windows

Effectuez les étapes suivantes pour installer l'Application Builder.



**Remarque** Certains programmes de détection de virus interfèrent avec le programme d'installation. Désactivez tous les programmes de détection automatique de virus avant de procéder à l'installation. Après l'installation, vérifiez votre disque dur pour vous assurer

qu'il ne contient pas de virus et activez tout programme de détection de virus que vous aviez désactivé.

1. Insérez le CD d'installation.
2. Exécutez `setup.exe`.
3. Changez le chemin d'accès à votre répertoire LabVIEW, le cas échéant, et cliquez sur le bouton **Installer**.
4. Vérifiez l'installation comme décrit à la section [Vérification de l'installation de l'Application Builder](#) de ce document.

## Mac OS

Effectuez les étapes suivantes pour installer l'Application Builder.



**Remarque** Certains programmes de détection de virus interfèrent avec le programme d'installation. Désactivez tous les programmes de détection automatique de virus avant de procéder à l'installation. Après l'installation, vérifiez votre disque dur pour vous assurer qu'il ne contient pas de virus et activez tout programme de détection de virus que vous aviez désactivé.

1. Insérez le CD d'installation.
2. Double-cliquez sur l'icône **LabVIEW AppLibs Installer**.
3. Après que vous avez cliqué sur le bouton **Installer**, l'installateur vous invite à sélectionner un dossier de destination. Sélectionnez le dossier LabVIEW.
4. Vérifiez l'installation comme décrit à la section [Vérification de l'installation de l'Application Builder](#) de ce document.

## UNIX

Effectuez les étapes suivantes pour installer l'Application Builder sous Linux ou Sun. Des privilèges racine ne sont pas nécessaires pour installer ces bibliothèques, mais vous devez avoir la possibilité d'écrire dans le répertoire LabVIEW où vous souhaitez installer les bibliothèques.

1. Insérez le CD d'installation.
2. Entrez la commande UNIX suivante adaptée au système d'exploitation :
  - **(Linux)**  
`cd /mnt/cdrom/linux`  
où `cdrom` est le répertoire dans lequel vous installez le CD.

- **(Solaris 2)**  
volcheck  
cd /cdrom/cdrom0/solaris2  
où *cdrom* est le répertoire dans lequel vous installez le CD.
3. Lancez l'installeur en entrant la commande suivante :  
./INSTALL
  4. Suivez les instructions qui s'affichent à l'écran.
  5. Vérifiez l'installation comme décrit à la section *Vérification de l'installation de l'Application Builder* de ce document.

## Vérification de l'installation de l'Application Builder

---

Lancez LabVIEW une fois que vous avez installé l'Application Builder et sélectionnez **Outils»Construire une application ou une bibliothèque partagée (DLL)**. Vérifiez que le répertoire LabVIEW contient un répertoire `applibs`. Si ce répertoire n'est pas présent, reportez-vous à la section *Installation de l'Application Builder* de ce document et réinstallez l'Application Builder.

Si les bibliothèques sont installées correctement, le répertoire `examples` contient le fichier `appbuild.llb`.

## Mises en garde et recommandations lors de la construction d'applications et de bibliothèques partagées

---

La liste suivante décrit certaines des mises en garde et recommandations à prendre en considération lors de la construction d'un VI dans une application ou une bibliothèque partagée que vous voulez distribuer aux utilisateurs :

- Certaines propriétés et méthodes du VI Serveur ne sont pas prises en charge dans le moteur d'exécution de LabVIEW. Évitez d'utiliser ces propriétés et méthodes dans le VI que vous voulez construire à l'intérieur d'une application ou bibliothèque partagée.
- Assurez-vous que les paramètres de la boîte de dialogue **Propriétés du VI** sont corrects. Par exemple, vous pourriez vouloir configurer le VI pour qu'il s'exécute lors de son ouverture ou vous pourriez vouloir cacher les boutons de la barre de menus.
- Pensez à distribuer de la documentation avec l'application ou la bibliothèque partagée de façon à ce que les utilisateurs possèdent les informations dont ils ont besoin pour utiliser l'application ou la

bibliothèque partagée. Reportez-vous au chapitre 5, *Creating Documentation*, du manuel *LabVIEW Development Guidelines*, pour obtenir de plus amples informations sur le développement de la documentation utilisateur.



**Remarque** Ne distribuez *pas* la documentation sur LabVIEW avec les applications ou bibliothèques partagées que vous créez. La documentation LabVIEW est protégée par les lois du copyright.

- Si le VI charge d'autres VIs de manière dynamique en utilisant le VI Serveur ou les Nœuds d'appel par référence, vous devez ajouter les VIs chargés dynamiquement en cliquant sur le bouton **Ajouter un VI chargé dynamiquement** sur l'onglet **Fichiers source** de la boîte de dialogue **Construire une application ou une bibliothèque partagée (DLL)**.
- Si le VI charge d'autres VIs dynamiquement à l'aide du VI serveur ou de Nœuds d'appel par référence, assurez-vous que l'application ou la bibliothèque partagée crée correctement les chemins vers les VIs. Lorsque vous incluez les VIs chargés dynamiquement dans l'application ou la bibliothèque partagée, les chemins vers les VIs changent. Par exemple, si vous construisez `test.vi` dans une application, son chemin est `C:\..\Application.exe\test.vi`, où `C:\..\Application.exe` représente le chemin vers l'application et son nom de fichier.
- Si le VI utilise la fonction Chemin du VI actuel, assurez-vous que la fonction se comporte comme prévu dans l'application ou la bibliothèque partagée. Dans une application ou une bibliothèque partagée, la fonction Chemin du VI actuel renvoie le chemin du VI dans le fichier de l'application et traite ce dernier comme s'il s'agissait d'une bibliothèque de VIs. Par exemple, si vous construisez `test.vi` dans une application, la fonction renvoie un chemin `C:\..\Application.exe\test.vi`, où `C:\..\Application.exe` représente le chemin vers l'application et son nom de fichier.
- Lorsque vous fermez toutes les faces-avant d'une application, celle-ci s'arrête. Si le VI que vous avez construit dans l'application contient du code qui s'exécute après que la dernière face-avant ne se ferme, ce code ne s'exécute pas dans l'application. Évitez d'écrire du code de diagramme qui s'exécute après la fermeture de la dernière face-avant.
- Si le VI utilise des menus d'exécution personnalisés, assurez-vous que tous les éléments de menu d'application que le VI utilise sont disponibles dans le moteur d'exécution de LabVIEW.

- Pensez à distribuer un fichier de préférences qui contient vos paramètres d'environnement de travail sous LabVIEW dans l'application.
- Envisagez également de créer une boîte de dialogue **À propos** pour afficher des informations concernant l'application.
- Vous devez distribuer le moteur d'exécution de LabVIEW avec l'application ou la bibliothèque partagée, sauf si ce moteur est déjà installé sur la machine cible.

Reportez-vous à l'*Aide LabVIEW* pour obtenir des informations complémentaires sur chacune de ces mises en garde et de ces recommandations lors de la construction d'une application ou d'une bibliothèque partagée, y compris la liste complète des propriétés et des méthodes du VI Serveur non prises en charge par le moteur d'exécution LabVIEW.

## Modifications apportées à l'Application Builder

---

Les sections suivantes décrivent les modifications qui ont été introduites entre chacune des versions les plus récentes de l'Application Builder.

Reportez-vous aux *Notes d'information Application Builder LabVIEW* pour obtenir de plus amples informations sur les modifications apportées entre les versions 5.0 et 6.0.

### Modifications apportées entre les versions 6.1 et 7.0

La liste suivante décrit les changements apportés entre les versions 6.1 et 7.0 :

- Si une application ou une bibliothèque partagée nécessite un support de port série ou parallèle ou des informations sur la configuration matérielle, vous pouvez inclure ce support et ces informations dans l'installateur en cochant les cases appropriées de la boîte de dialogue **Paramètres avancés de l'installateur**.
- Si le Module LabVIEW Real-Time est installé sur votre ordinateur, vous pouvez inclure le support pour le Module LabVIEW Real-Time dans l'installateur en cochant la case **Support pour LabVIEW RT** de la boîte de dialogue **Paramètres avancés de l'installateur**.
- Si vous voulez qu'un programme s'exécute après que l'installation est terminée, mais avant que l'installateur ne se ferme, cochez la case **Attente synchrone** de la boîte de dialogue **Paramètres avancés de l'installateur**.

- Si vous utilisez la propriété Application : arguments de la ligne de commande pour lire les arguments de la ligne de commande définis par l'utilisateur transmis lors du lancement d'une application ou d'une bibliothèque partagée, vous pouvez cocher la case **Transmettre tous les arguments de ligne de commande à l'application** sur l'onglet **Paramètres appli.** de la boîte de dialogue **Construire une application ou une bibliothèque partagée (DLL)** pour transmettre tous les arguments en tant qu'arguments définis par l'utilisateur, de sorte que vous n'avez pas à entrer deux traits d'union avant les arguments définis par l'utilisateur sur la ligne de commande.
- Lorsque vous construisez une application ou une bibliothèque partagée à partir d'un VI qui contient des commandes ou des types de définitions personnalisés, l'Application Builder déconnecte les commandes et les définitions de type afin que vous n'avez pas à inclure les fichiers avec l'application ou la bibliothèque partagée. Si le VI contient un VI polymorphe, l'Application Builder supprime toutes les instances inutilisées du VI polymorphe, ce qui réduit la taille de l'application ou de la bibliothèque partagée.
- Les bibliothèques partagées que vous construisez dans LabVIEW utilisent les types ANSI au lieu des types de données LabVIEW dans l'en-tête généré.

## Modifications apportées entre les versions 6.0 et 6.1

La liste suivante décrit les changements apportés entre les versions 6.0 et 6.1 :

- La boîte de dialogue **Paramètres de destination** de LabVIEW 6.0 se compose désormais de deux boîtes de dialogue distinctes. La première boîte de dialogue, **Paramètres de destination de la construction**, accessible à partir de l'onglet **Fichiers Source** de la boîte de dialogue **Construire une application ou une bibliothèque partagée (DLL)**, décrit les paramètres de destination de la construction fichier par fichier. La seconde boîte de dialogue, **Paramètres de destination de l'installation**, accessible à partir de l'onglet **Paramètres installeur** de la boîte de dialogue **Construire une application ou une bibliothèque partagée (DLL)**, décrit les paramètres de destination de l'installeur fichier par fichier.
- Une fois que vous avez construit une application ou une bibliothèque partagée et enregistré le fichier du script de construction, vous pouvez cliquer sur la flèche près du bouton **Chargement** pour afficher la liste des scripts de constructions les plus récemment utilisés.
- **(Windows)** LabVIEW génère les installeurs MSI (Microsoft Installer). Vous pouvez choisir parmi 10 répertoires d'installation.



- **(Windows)** Les boutons **Taille du support** et **Espace supplémentaire sur le premier disque (ko)** ont été supprimés de l'onglet **Paramètres installateur** de la boîte de dialogue **Construire une application ou une bibliothèque partagée (DLL)** car l'installateur Microsoft (MSI) ne prend pas en charge le fractionnement sur plusieurs disques.
- **(Windows)** L'option **Créer un programme de désinstallation** a été supprimée de la boîte de dialogue **Paramètres avancés de l'installateur** car l'Application Builder crée toujours un désinstallateur. Vous pouvez en outre configurer l'installateur de sorte qu'il n'installe que les parties du moteur d'exécution LabVIEW que vous voulez.
- Pour commander une application à distance, vous devez inclure l'utilitaire Gestionnaire de licences NI dans l'installateur en cliquant sur le bouton **Avancé** dans l'onglet **Paramètres installateur** de la boîte de dialogue **Construire une application ou une bibliothèque partagée (DLL)** et en cochant la case **Support pour licence de face-avant distante**. Pour obtenir des informations complémentaires sur le Gestionnaire de licences NI, sélectionnez **Démarrer»Programmes»National Instruments»Gestionnaire de licences NI**.
- **(Windows)** Le nombre de langues pour lesquelles vous pouvez créer un désinstallateur n'est plus 12, mais 4 langues (anglais, français, allemand et japonais).
- Si vous développez un exécutable ou une bibliothèque partagée, vous devez distribuer tous les fichiers texte de codes d'erreur pertinents, qu'ils soient définis par l'utilisateur ou provenant de tiers, avec votre exécutable ou bibliothèque partagée. 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 l'utilitaire d'une société tierce pour créer un installateur, repérez les fichiers texte de codes d'erreur que vous voulez distribuer et configurez le sous-répertoire d'installation qui convient pour ces fichiers. Les fichiers texte de codes d'erreur définis par l'utilisateur se trouvent dans le répertoire `labview\user.lib\errors` et les fichiers provenant de tiers 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 installateur pour une bibliothèque partagée, installez les fichiers texte de codes d'erreur dans le répertoire `National Instruments\shared\errors`. Chacun de ces répertoires peut également contenir un sous-répertoire pour chaque langue avec les codes d'erreur traduits.