

# Développez et déployez une application GTK+ sous Windows

par Nicolas Joseph ([home](#)) ([Blog](#))

Date de publication : 30 Janvier 2008

Dernière mise à jour :

Cet article va vous expliquer comment développer puis déployer une application écrite grâce à la bibliothèque GTK+ sous Windows.

Public concerné

I - Introduction

II - Installation

II-A - Code::Blocks

II-B - GTK+

II-C - Test

III - Déploiement

IV - Changer le thème de GTK+

V - Conclusion

VI - Remerciements

## Public concerné



## I - Introduction

Sous Linux pour développer une application à l'aide de GTK+, c'est relativement simple. Pour cela, il suffit de s'appuyer sur les outils fournis par votre distribution : apt-get ou yum, par exemple, pour l'installation des bibliothèques, les autotools pour la compilation et quelques dizaines d'éditeurs de texte, IDE ou autre RAD pour le développement. Il en va de même pour la distribution de votre application.

L'un des (nombreux) avantages de GTK+ est la possibilité de disposer du même outil sous Linux et sous Windows. Cependant les utilisateurs et développeurs disposant de ce dernier ne sont pas les mieux servis : aucun système pour gérer les dépendances, une ligne de commande pauvre, ...

Dans ce tutoriel je vais donc vous présenter comment installer le nécessaire pour développer une application à l'aide de GTK+ et ensuite comment créer un installeur pour distribuer facilement votre création.

## II - Installation

### II-A - Code::Blocks


Pour vos développements utilisant le C ou le C++ sous Windows, je vous conseille l'IDE **Code::Blocks**. Son installation est extrêmement simple, téléchargez et installez les paquets suivant :

- **MinGW-5.1.3.exe**
- **gdb-6.6.tar.bz2**
- **Code::Blocks 8.02**

Installez *mingw* (gcc, g++ et make), à la racine du disque C: (C:\MinGW) et décompressez gdb au même endroit. Ensuite installez Code::Blocks dans le répertoire des programmes (C:\Programs Files\CodeBlocks). Voilà vous pouvez dès à présent utiliser Code::Blocks.


Passons maintenant à l'installation des bibliothèques GTK+.

### II-B - GTK+

L'installation est encore plus simple. Allez sur le site officiel de GTK+ : <http://gtk.org/download-windows.html> et téléchargez tous les packages **binaries** et **dev** sans oublier les dépendances requises, marquées d'un  :

- GLib
- GTK+
- Pango
- ATK
- Cairo
- zlib
- gettext-runtime
- libpng
- libjpeg
- libtiff

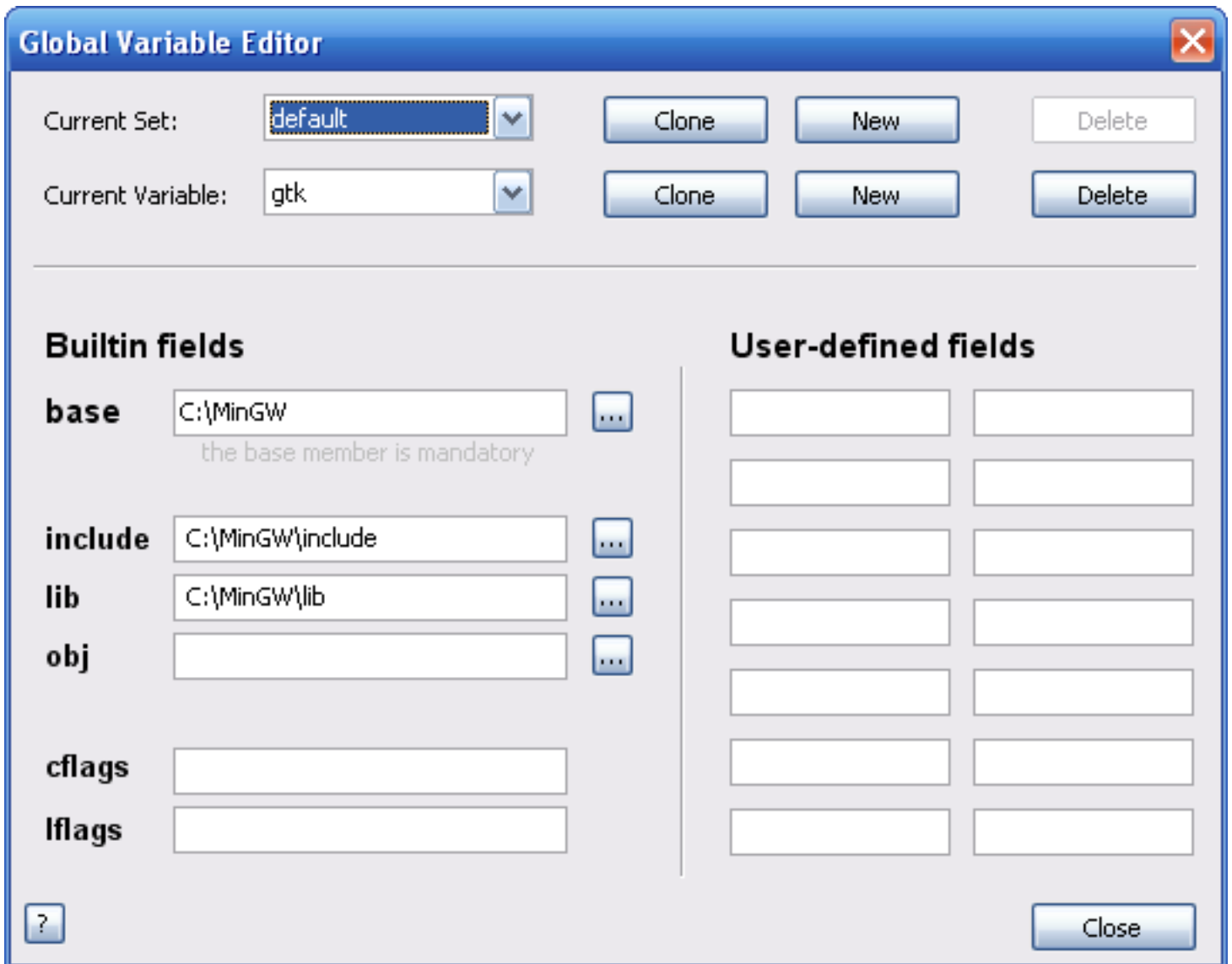
Et dézipper tout à la racine du répertoire de mingw !

 *Bizarrement le fichier **zlib.dll** se retrouve à la racine, pour éviter tout problème, déplacez le fichier dans le sous-répertoire **bin**.*

### II-C - Test

Maintenant il nous reste plus qu'à tester tout ça ! Commencez par lancer Code::Blocks. Vérifiez dans *Settings* -> *Compiler and debugger* onglet *Toolchain executables* que le répertoire correspond bien à celui choisi (par défaut C:\MinGW).

Ensuite *Settings* -> *Global Variable...* pour créer une variable qui permettra à Code::Blocks de trouver nos fichiers d'entête et nos bibliothèques lors de la création d'un nouveau projet. Voici la configuration par défaut :



Pour finir *File* -> *New* -> *Project...*, choisissez le template *GTK+ project*. Lorsque l'assistant de configuration vous demande l'emplacement de GTK+, entrez simplement `${#gtk}`.

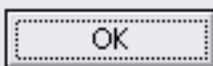
Pressez la touche F9 et admirez le résultat !

### III - Déploiement

Maintenant votre application compile et s'exécute correctement, mais uniquement sous Code::Blocks. Essayez d'exécuter directement votre programme (dans le sous-répertoire bin\Debug de votre projet) et vous devriez obtenir un beau message d'erreur :

#### Composant introuvable

Cette application n'a pas pu démarrer car libglib-2.0-0.dll est introuvable. La réinstallation de cette application peut corriger ce



Tout simplement parce que vos dll ne se trouvent pas dans l'un des répertoires présent dans la variable d'environnement PATH. Vous pouvez bien sûr l'ajouter mais qu'en sera-t-il lorsque vous voudrez distribuer votre programme ? Vous souhaitez faire subir la même punition aux utilisateurs de vos programmes ? Pas très motivant pour utiliser votre programme. Il existe bien sûr la solution de lier les bibliothèques statiquement à votre programme, mais au cours de mes tests je n'ai jamais réussi... De plus cela vous oblige à publier votre projet sous licence libre.

Nous allons donc partir sur une solution plus propre et plus professionnelle basée sur un exécutable auto-extractable créé grâce à **inno setup**.

Une fois installé, lancez le programme. Vous obtenez un "simple" éditeur de fichier texte. Voici un fichier type qui contient l'ensemble des fichiers dont un programme utilisant GTK+ à besoin pour s'exécuter.

N'oubliez pas de créer un fichier COPYING.txt contenant la licence de votre programme.

```
; Script generated by the Inno Setup Script Wizard.
; SEE THE DOCUMENTATION FOR DETAILS ON CREATING INNO SETUP SCRIPT FILES!

[Setup]
AppName=test
AppVerName=test 0.1
AppPublisher=developpez.com
AppPublisherURL=http://www.developpez.com/
AppSupportURL=http://www.developpez.com/
AppUpdatesURL=http://www.developpez.com/
DefaultDirName={pf}\test
DefaultGroupName=test
AllowNoIcons=yes
LicenseFile=COPYING.txt
OutputBaseFilename=setup
Compression=lzma
SolidCompression=yes

[Languages]
Name: "french"; MessagesFile: "compiler:Languages\French.isl"

[Tasks]
Name: "desktopicon"; Description: "{cm:CreateDesktopIcon}"; GroupDescription:
"{cm:AdditionalIcons}"; Flags: unchecked
Name: "quicklaunchicon"; Description: "{cm:CreateQuickLaunchIcon}"; GroupDescription:
"{cm:AdditionalIcons}"; Flags: unchecked
```

**[Files]**

```
Source: "C:\Documents and Settings\gege2061\Mes documents\test\bin\Release\test.exe"; DestDir:
"{app}"; Flags: ignoreversion

; GTK+ dependencies
; DLL
Source: "C:\MinGW\bin\libcairo-2.dll"; DestDir: "{app}"; Flags: ignoreversion
Source: "C:\MinGW\bin\libpangocairo-1.0-0.dll"; DestDir: "{app}"; Flags: ignoreversion
Source: "C:\MinGW\bin\jpeg62.dll"; DestDir: "{app}"; Flags: ignoreversion
Source: "C:\MinGW\bin\libtiff3.dll"; DestDir: "{app}"; Flags: ignoreversion
Source: "C:\MinGW\bin\libpng13.dll"; DestDir: "{app}"; Flags: ignoreversion
Source: "C:\MinGW\bin\zlib1.dll"; DestDir: "{app}"; Flags: ignoreversion
Source: "C:\MinGW\bin\intl.dll"; DestDir: "{app}"; Flags: ignoreversion
Source: "C:\MinGW\bin\libatk-1.0-0.dll"; DestDir: "{app}"; Flags: ignoreversion
Source: "C:\MinGW\bin\libgdk_pixbuf-2.0-0.dll"; DestDir: "{app}"; Flags: ignoreversion
Source: "C:\MinGW\bin\libgdk-win32-2.0-0.dll"; DestDir: "{app}"; Flags: ignoreversion
Source: "C:\MinGW\bin\libglib-2.0-0.dll"; DestDir: "{app}"; Flags: ignoreversion
Source: "C:\MinGW\bin\libgmodule-2.0-0.dll"; DestDir: "{app}"; Flags: ignoreversion
Source: "C:\MinGW\bin\libgobject-2.0-0.dll"; DestDir: "{app}"; Flags: ignoreversion
Source: "C:\MinGW\bin\libgthread-2.0-0.dll"; DestDir: "{app}"; Flags: ignoreversion
Source: "C:\MinGW\bin\libgtk-win32-2.0-0.dll"; DestDir: "{app}"; Flags: ignoreversion
Source: "C:\MinGW\bin\libpango-1.0-0.dll"; DestDir: "{app}"; Flags: ignoreversion
Source: "C:\MinGW\bin\libpangoft2-1.0-0.dll"; DestDir: "{app}"; Flags: ignoreversion
Source: "C:\MinGW\bin\libpangowin32-1.0-0.dll"; DestDir: "{app}"; Flags: ignoreversion

; .mo
Source: "C:\MinGW\lib\locale\fr\LC_MESSAGES\atk10.mo"; DestDir: "{app}\lib\locale\fr\LC_MESSAGES";
Flags: ignoreversion
Source: "C:\MinGW\share\locale\fr\LC_MESSAGES\glib20.mo"; DestDir:
"{app}\share\locale\fr\LC_MESSAGES"; Flags: ignoreversion
Source: "C:\MinGW\share\locale\fr\LC_MESSAGES\gtk20.mo"; DestDir:
"{app}\share\locale\fr\LC_MESSAGES"; Flags: ignoreversion
Source: "C:\MinGW\share\locale\fr\LC_MESSAGES\gtk20-properties.mo"; DestDir:
"{app}\share\locale\fr\LC_MESSAGES"; Flags: ignoreversion

;
Source: "C:\MinGW\etc\gtk-2.0\gdk-pixbuf.loaders"; DestDir: "{app}\etc\gtk-2.0"; Flags:
ignoreversion
Source: "C:\MinGW\etc\gtk-2.0\gtk.immodules"; DestDir: "{app}\etc\gtk-2.0"; Flags: ignoreversion
Source: "C:\MinGW\etc\pango\pango.modules"; DestDir: "{app}\etc\pango"
Source: "C:\MinGW\etc\pango\pango.alias"; DestDir: "{app}\etc\pango"

; optional: let the user make the app look more Windows-like
Source: "C:\MinGW\lib\gtk-2.0\2.10.0\engines\libwimp.dll"; DestDir:
"{app}\lib\gtk-2.0\2.10.0\engines"
Source: "C:\MinGW\lib\gtk-2.0\2.10.0\engines\libpixmap.dll"; DestDir:
"{app}\lib\gtk-2.0\2.10.0\engines"
Source: "C:\MinGW\lib\gtk-2.0\2.10.0\immodules\im-am-et.dll"; DestDir:
"{app}\lib\gtk-2.0\2.10.0\immodules"
Source: "C:\MinGW\lib\gtk-2.0\2.10.0\immodules\im-cedilla.dll"; DestDir:
"{app}\lib\gtk-2.0\2.10.0\immodules"
Source: "C:\MinGW\lib\gtk-2.0\2.10.0\immodules\im-cyrillic-translit.dll"; DestDir:
"{app}\lib\gtk-2.0\2.10.0\immodules"
Source: "C:\MinGW\lib\gtk-2.0\2.10.0\immodules\im-ime.dll"; DestDir:
"{app}\lib\gtk-2.0\2.10.0\immodules"
Source: "C:\MinGW\lib\gtk-2.0\2.10.0\immodules\im-inuktitut.dll"; DestDir:
"{app}\lib\gtk-2.0\2.10.0\immodules"
Source: "C:\MinGW\lib\gtk-2.0\2.10.0\immodules\im-ipa.dll"; DestDir:
"{app}\lib\gtk-2.0\2.10.0\immodules"
Source: "C:\MinGW\lib\gtk-2.0\2.10.0\immodules\im-multipress.dll"; DestDir:
"{app}\lib\gtk-2.0\2.10.0\immodules"
Source: "C:\MinGW\lib\gtk-2.0\2.10.0\immodules\im-thai.dll"; DestDir:
"{app}\lib\gtk-2.0\2.10.0\immodules"
Source: "C:\MinGW\lib\gtk-2.0\2.10.0\immodules\im-ti-er.dll"; DestDir:
"{app}\lib\gtk-2.0\2.10.0\immodules"
Source: "C:\MinGW\lib\gtk-2.0\2.10.0\immodules\im-ti-et.dll"; DestDir:
"{app}\lib\gtk-2.0\2.10.0\immodules"
```

```
Source: "C:\MinGW\lib\gtk-2.0\2.10.0\immodules\im-viqr.dll"; Destdir:  
"{app}\lib\gtk-2.0\2.10.0\immodules"
```

[Run]

```
Filename: "{app}\test.exe"; Description: "{cm:LaunchProgram,test}"; Flags: nowait postinstall  
skipifsilent
```

Pour obtenir votre installateur, tapez ctrl+F9 et vous obtenez un exécutable setup.exe dans le sous-répertoire *Output*. Exécutez-le et une fois terminé lancez de nouveau votre programme... Admirez le résultat !

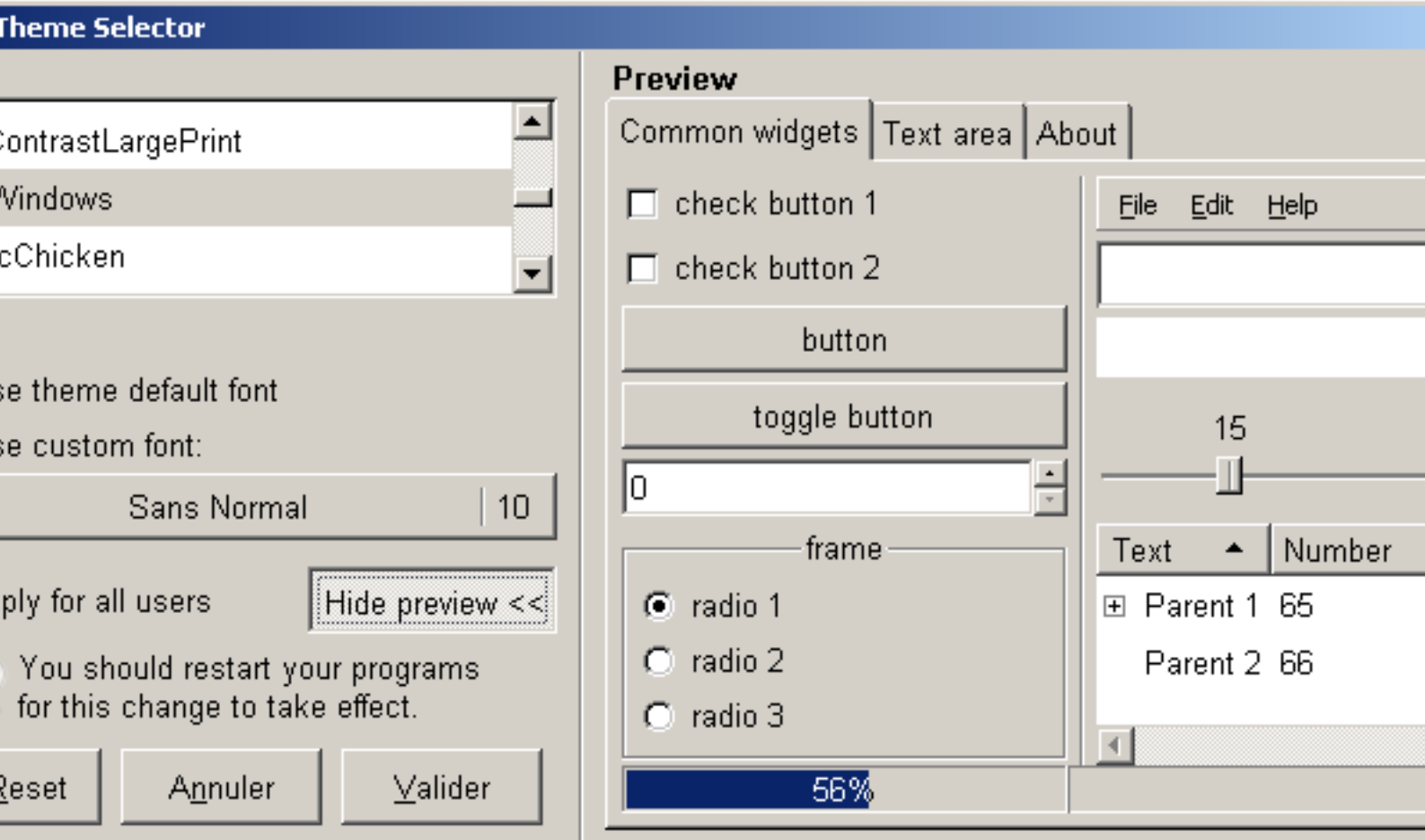
## IV - Changer le thème de GTK+

Pour ceux d'entre vous qui utilisent un autre thème que le thème classique de windows XP, vous aurez remarqué que l'apparence de vos applications n'est pas des plus belles.

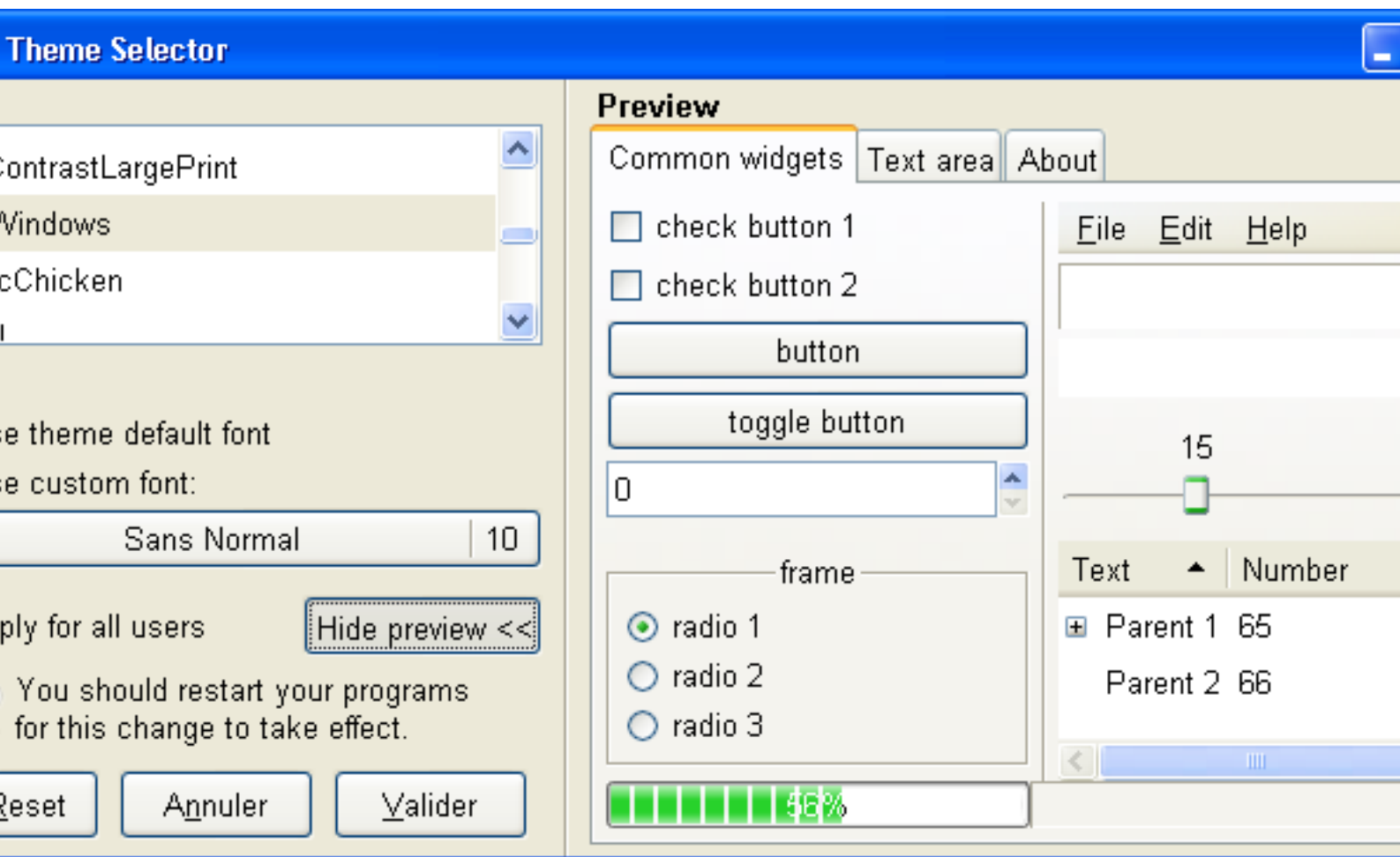
Heureusement GTK+ dispose d'un système de thème très puissant qui vous permet de parler ce problème. Il faut juste s'avoir l'utiliser.

- Pour commencer, téléchargez l'installateur **gtk2-themes-2008-01-21-ash.exe** (vérifiez qu'il s'agit bien de la dernière version sur le site **GTK+ for Windows Runtime Environment**),
- Installez le gestionnaire de thème dans le même répertoire que MinGW,
- Les exécutables sont installés à la racine du répertoire, ils faut donc déplacer les deux fichiers gtk2\_prefs.exe et gtk.ico dans le répertoire bin/ (pensez à modifier les raccourcis présent dans le menu démarrer en conséquence),
- Exécuter le programme gtk2\_prefs.exe et choisissez votre thème. Pour un thème qui colle à celui de Windows, préférez MS-Windows.

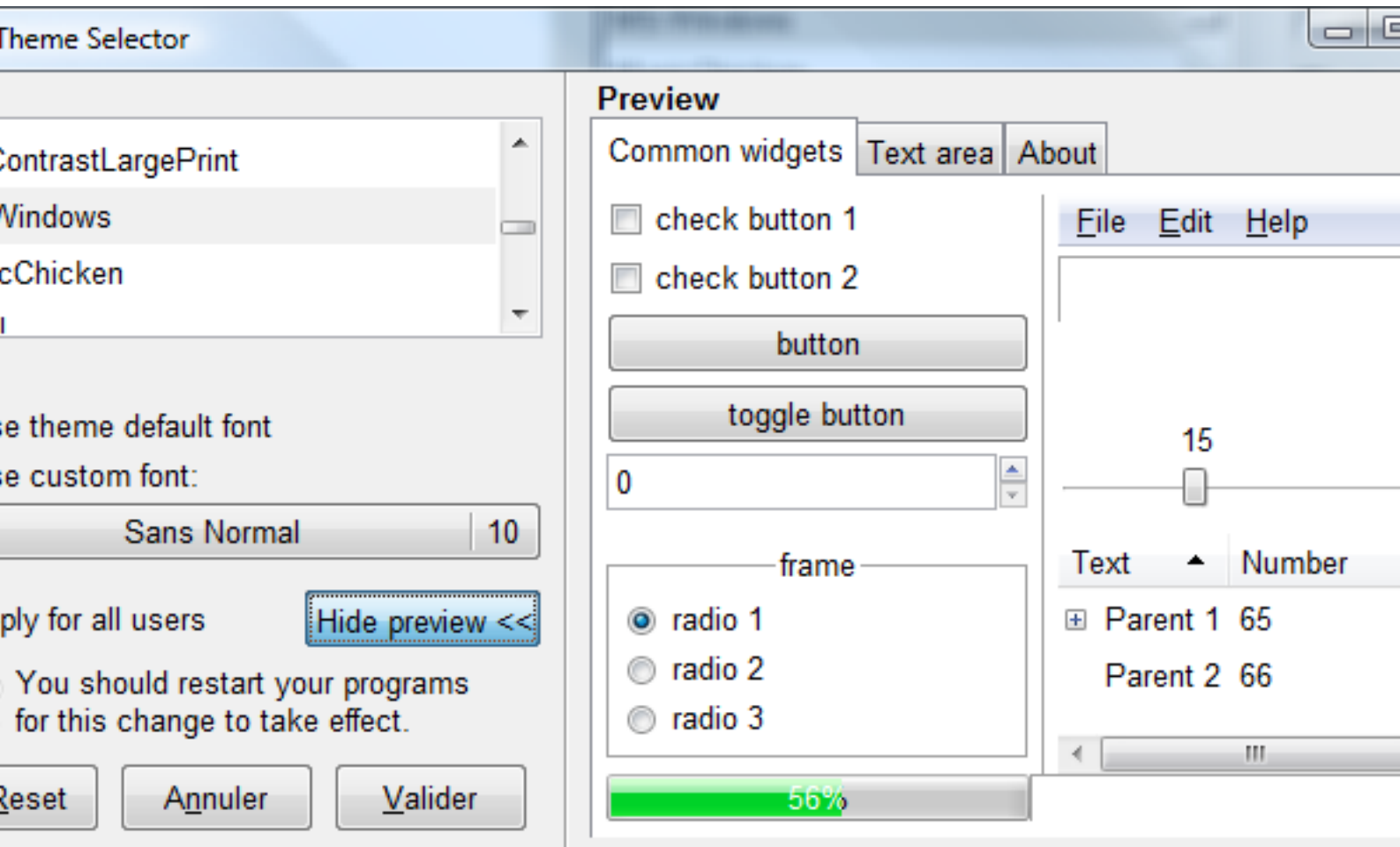
Voici le résultat, en changeant uniquement le thème dans le panneau de configuration de Windows :



*Thème classique*



Thème XP



Thème Vista

## V - Conclusion

Voilà vous pouvez maintenant faire profiter aux utilisateurs Windows de vos créations GTK+. Même si ce n'est pas aussi puissant qu'un système de paquet, cela simplifie grandement l'installation pour les futurs utilisateurs.

L'utilisation faite ici de inno setup est extrêmement basique, si vous souhaitez aller plus loin je vous conseille de lire les deux tutoriels de [ThierryAIM](#) : [distribuer vos applications VB6 avec InnoSetup](#) et [la section \[Code\] de InnoSetup](#). Et vous pouvez bien sûr poser vos questions sur le [forum approprié](#).

## VI - Remerciements

Merci à **buchs** par sa relecture orthographique de cet article.

