



Aide-mémoire LibreOffice

LibreOffice Basic

Les paramètres d'exécution



v. 2.0 - 02/04/2021

Rédigé avec LibreOffice v. 7.0.5 - Plateforme : Toutes

Connaître les chemins définis dans LibreOffice

Chemins des fichiers utilisateur

Il s'agit des chemins définis dans Outils > Options > LibreOffice > Chemins
 Utilisez le service PathSettings :

```
Dim oPaths As Object
Dim Dirs As Variant
oPaths = CreateUnoService("com.sun.star.util.PathSettings")
Dirs = Split(oPaths.Xxx, ";")
```

où Xxx est la propriété correspondant au répertoire visé, parmi :

La propriété...	... désigne
Addin	Le répertoire qui contient les add-in de l'ancien temps.
AutoCorrect	Les paramètres du dialogue d'autocorrection.
AutoText	Le répertoire de stockage des autotextes.
Backup	Les sauvegardes automatiques sont enregistrées ici.
Basic	Vous trouvez ici les fichiers Basic utilisées par les autopilotes.
Bitmap	Icones des barres d'outils.
Config	Fichiers de configuration.
Dictionary	Lieu de stockage des dictionnaires fournis.
Favorite	Chemin de sauvegarde des marque-pages des dossiers.
Filter	Répertoire de stockage des filtres.
Gallery	Répertoires pour les fichiers multimédia et la base Galerie.
Graphic	Rép. affiché lors de l'ouverture/l'enregistrement d'un graphique.
Help	Chemin vers les fichiers d'aide.
Linguistic	Ici se trouvent les fichiers pour la vérification orthographique.
Module	Chemins vers les modules.
Palette	Chemins vers les fichiers palettes (. sob à .sof).
Plugin	Répertoires de stockage des plugins.
Storage	Stockage des fichiers de messagerie, de newsgroups (ex : serveur FTP).
Temp	L'URL de base vers les fichiers temporaires.
Template	Répertoires dédiés au stockage des modèles.
UIConfig	Répertoires globaux pour le stockage des fichiers de configuration.
UserConfig	Le dossier qui contient les paramètres de l'utilisateur.
Work	Le chemin du dossier de travail, modifiable selon les besoins de l'utilisateur.

BasePathShareLayer ?
 BasePathUserLayer ?

Chemin d'installation d'une extension

Utilisez le singleton « package provider » :
 "/singletons/com.sun.star.deployment.PackageInformationProvider"

```
Dim oInfo As Object, Path As String
oInfo =
    GetDefaultContext.getByname("/singletons/com.sun.star.deployment.P
    ackageInformationProvider")
If Not IsNull(oInfo) Then
    Path = oInfo.getPackageLocation(ExtID)
End If
If (Path <> "") Then Path = Path & "/"
```

où ExtID est l'identificateur unique de l'extension (ex : "com.company.SomeName")

- Path contient soit le répertoire (format URL) soit vide si pas trouvé.
- Vous pouvez aussi utiliser l'expandeur de chaînes, avec la macro UNO_USER_PACKAGES_CACHE (voir plus loin).

Connaître des paramètres d'exécution de LibreOffice

Deux services complémentaires sont disponibles : PathSubstitution et MacroExpander.

Utilisez le service PathSubstitution

```
Dim oSubst As Object, Result As String
oSubst = CreateUnoService("com.sun.star.util.PathSubstitution")
Result = oSubst.getSubstituteVariableValue("$ (nom_variable)")
```

- La variable à substituer est une chaîne de la forme : \$ (nom_variable)
- Le résultat est au format URL.

Variables substituables :

La variable...	... désigne
\$(inst)	Le chemin d'installation de LibreOffice.
\$(prog)	Le chemin d'accès au programme soffice.
\$(user)	Le répertoire d'installation de l'utilisateur.
\$(work)	Le répertoire de travail de l'utilisateur. Sous Windows, il s'agira du répertoire Mes Documents. Sous Unix, c'est le répertoire « home ».
\$(home)	Le répertoire de l'utilisateur. Sous Unix, c'est le répertoire « home ».
\$(temp)	Sous Windows, c'est le répertoire pointé par CSIDL_PERSONAL, p.ex : "Documents and Settings\username\Documents".
\$(path)	Le répertoire temporaire courant.
\$(user_name)	Le contenu de la variable d'environnement PATH.
(depuis LibO 5.2)	Le nom d'utilisateur de la session courante (sans nom de domaine sous Windows).
\$(langid)	Le code de la langue utilisée par LibreOffice. Ex : 1036 pour le français (France).
\$(vlang)	Le code de la langue utilisée par LibreOffice, sous forme textuelle. Ex : "fr" pour le français.

Utilisez le singleton expandeur de (chaînes) macros

"/singletons/com.sun.star.util.theMacroExpander"
 et appelez sa méthode ExpandMacros() :

```
Dim oContext as Object 'objet contexte
Dim oMacroExpand as Object 'macro expandeur
Dim Result As String
oContext = getProcessServiceManager().DefaultContext
oMacroExpand =
    oContext.getValueByName("/singletons/com.sun.star.util.theMacroExp
    ander")

Result = oMacroExpand.ExpandMacros("$UNO_USER_PACKAGES_CACHE")
```

- Le nom d'une chaîne macro doit commencer par le symbole \$
- Les répertoires et fichiers sont retournés au format URL.

Macros

Voir : https://wiki.documentfoundation.org/Development/Environment_variables

et pour le fichier bootstrap :

ORIGIN	Le dossier d'installation de LibreOffice.
SYSUSERCONFIG	Le répertoire des paramètres de l'utilisateur, sous sa session.
UNO_USER_PACKAGES_CACHE	Le répertoire de stockage des extensions.
USERNAME	Le nom du compte de l'utilisateur.

Connaître les paramètres en ligne de commande LibreOffice

Seules les options utiles en utilisation « macro » en ligne de commande sont décrites. L'ensemble est cependant disponible sur cette page (EN) (vérif. 04/2021) :
<https://dnimruoynepo.blogspot.fr/2016/12/command-line-arguments-in-libreoffice.html>

Aide et information

--version	Affiche le numéro de version.
--nstoporarydirectory	(seulement dans le bac à sable MacOS X) Renvoie le chemin du rép. temp. de l'utilisateur courant. Prioritaire sur tous les autres paramètres.

Paramètres généraux

--quickstart[=no]	[Dés]active le démarrage rapide. Une seule valeur est possible après le symbole « égale » (no), ce qui désactive ce service.
--nolockcheck	Désactive la vérification des instances distantes utilisant l'installation.
--infilter={filtre}	Force la saisie d'un type de filtre, si possible. Lorsque ce n'est pas possible, LibreOffice utilise le filtre disponible pour le document. Exemple : --infilter="Calc Office Open XML" --infilter="Text (encoded):UTF8,LF,,," Les noms de filtres sont susceptibles de changer. Ces exemples illustrent l'utilisation des paramètres. Il n'est malheureusement pas facile de connaître la liste des filtres disponibles.
--pidfile={fichier}	Enregistre le pid de soffice.bin dans {fichier}.
--display {affichage}	Définit la variable d'environnement DISPLAY sur les plateformes UNIX à la valeur {affichage}. Ce paramètre n'est pris en charge par le script de démarrage de LibreOffice que sur les plateformes UNIX.

Contrôle de l'interface utilisateur

--nologo	Désactive le splash screen de lancement.
--minimized	Démarre à l'état minimisé. Le splash screen n'est pas affiché.
--nodefault	Démarre et n'affiche rien d'autre que le splash screen (pas de fenêtre initiale)
--invisible	Démarre en mode invisible. Ni le splash screen ni la fenêtre du programme ne sont affichés. LibreOffice, les documents et les dialogues peuvent être contrôlés et ouverts via des appels à l'API. Lorsque ce paramètre est utilisé, LibreOffice ne peut être refermé qu'en utilisant le gestionnaire de tâches (Windows) ou la commande kill (Windows/Unix). --invisible ne peut pas être utilisé avec --quickstart.
--headless	Démarre en mode « headless » qui permet d'utiliser l'application sans aucune interface. Ce mode particulier s'utilise pour contrôler l'application depuis des clients externes par appels API.

invisible vs headless.

--invisible	ne désactive pas l'interface graphique : docs et dialogues affichés à l'écran.
--headless	appelle le « mode silencieux » quand l'IHM n'est pas nécessaire.
--norestore	Désactive le redémarrage et la récupération de fichiers après un plantage système.
--safe-mode	Démarre en mode sans échec. Ce mode permet d'utiliser provisoirement un profil neuf et aide à restaurer une configuration plantée.
--accept={UNO-URL}	Notifie qu'une chaîne « UNO Accept String » est utilisée lors de la création d'une chaîne « UNO Acceptor Threads » à partir de laquelle d'autres programmes peuvent se connecter pour accéder à l'API. {UNO-URL} est de la forme uno:connection-type,params;protocol-name,params;ObjectName. Le code LibreOffice montre que ObjectName est ignoré.
--unaccept={UNO-URL}	Referme un accepteur créé par --accept. Utilisez --unaccept=all pour refermer tous les accepteurs ouverts.

Paramètres pour développeur

--terminate_after_init	Quitte après la fin de l'initialisation (pas de chargement de document).
--eventtesting	Quitte après le chargement des documents.

Création de documents

Ces options permettent de créer des documents vides du type spécifié. Il n'est possible d'utiliser qu'un seul type dans une commande. Lorsqu'un nom de fichier est précisé, LibreOffice tente d'ouvrir ce fichier dans le module indiqué. Si ce n'est pas possible, ce fichier est ouvert dans celui qui convient.

Les options ci-dessous créent des documents vides du type spécifié :

```
--writer      --draw      --base      --math
--calc       --impress   --global    --web
```

Ouverture de fichiers

Ces options définissent la manière dont les noms des fichiers sont traités. Un nouveau traitement commence après les paramètres et s'achève aux suivants. Le traitement par défaut est soit l'ouverture des fichiers pour édition soit la création de nouveaux fichiers à partir de modèles.

```
-n           Traite les documents spécifiés comme des modèles pour créer de
             nouveaux documents.
-o           Ouvre les documents spécifiés pour édition, qu'il s'agisse de
             modèles ou pas.
--pt {Imprimante} Imprime les documents spécifiés vers l'imprimante
             {Imprimante}, après quoi ils sont refermés.
             Le splash screen n'est pas affiché. Si {Imprimante} est spécifié
             plusieurs fois, seule la dernière occurrence est utilisée pour tous
             les documents.
             Notez que le paramètre --printer-name de l'option --print-to-
             file interfère avec {Imprimante}.
-p           Imprime les fichiers spécifiés vers l'imprimante par défaut.
             Le splash screen n'est pas affiché. Si le nom d'un fichier comporte
             des espaces, il doit être spécifié entre guillemets doubles.
--view      Ouvre les fichiers spécifiés en mode visualisation (lecture seule).
--show      Ouvre et démarre la présentation immédiatement pour chacun
             des documents spécifiés. Les fichiers sont refermés après la
             présentation.
             Les fichiers autres que ceux d'Impress sont ouverts dans le mode
             par défaut, quel qu'ait été le mode précédent.
--convert-to Conversion de fichiers par lots (implique --headless).
OutExt[:OutFilterName] OutExt : l'extension cible.
[--outdir output_dir] OutFilterName : le filtre de conversion.
             Si --outdir n'est pas précisé, le répertoire de travail courant est
             utilisé comme répertoire de sortie
             Si --convert-to est utilisé plus d'une fois, c'est la dernière
             valeur de OutExt[:OutFilterName] qui est utilisée.
             Si --outdir est utilisé plus d'une fois, seule sa dernière valeur
             est prise en compte.
             Exemples :
             -- convert-to pdf *.doc
             -- convert-to pdf:writer_pdf_Export --outdir
             /home/user *.doc
             -- convert-to "html:XHTML Writer File:UTF8" *.doc
             -- convert-to "txt:Text (encoded):UTF8" *.doc
             Malheureusement, il n'est pas facile actuellement de connaître la
             liste des valeurs de filtres possibles. L'utilisation de cette option
             est donc délicate malgré sa grande utilité.
```

<https://ask.libreoffice.org/en/question/2641/convert-to-command-line-parameter/>

pointe vers :

<https://cgit.freedesktop.org/libreoffice/core/tree/filter/source/config/fragments/filters>

(EN) (vérif. 10/2017)

```
--print-to-file [--printer-name printer_name] [--outdir output_dir]
             Impression de fichiers par lots.
             Si --outdir n'est pas précisé, le répertoire de travail courant est
             utilisé comme répertoire de sortie
             Si --printer-name ou --outdir sont utilisés plusieurs fois,
             seule la dernière valeur de chacun est prise en compte.
             Notez que {Imprimante} de l'option --pt interfère avec --
             printer-name.
--cat       Envoie le contenu texte des fichiers spécifiés vers la console
             (implique l'option --headless).
             Ne peut pas être utilisée avec --convert-to.
--env:var [=valeur] Définit une variable de bootstrap.
             Par exemple, pour définir un chemin de profil utilisateur :
             -env:UserInstallation=file:///tmp/test
             Malheureusement, il n'est pas facile actuellement de connaître la
             liste des variables disponibles.
```

Appeler une macro en ligne de commande

Syntaxe

L'option --headless provoque l'exécution silencieuse (voir ci-dessus).

Macro globale

```
{soffice} "macro:///library/module/macro[(params)]"
```

Macro dans un document ODF

```
{soffice} chemin/a/doc.odf "macro://./library/module/macro[(params)]"
```

Forme de {soffice}

```
Windows      %programfiles%\libreoffice 5\program\soffice.exe
GNU/Linux    /opt/LibreOffice 5/program/soffice
```

Installer une macro... par macro

Vous n'installez pas une macro mais une **bibliothèque** qui la contient.

Principe

Un fichier « porteur » (Writer, Calc, etc.) contient la macro à installer et une macro d'installation :

- La macro d'installation est placée dans la bibliothèque Standard du document ;
- la macro à installer est *isolée de l'installeur*, donc placée dans sa propre bibliothèque. C'est cette bibliothèque qui sera installée.

Sauf cas particulier, le type du fichier « porteur » (Writer, Calc, etc.) est sans rapport avec la macro à installer. Writer est un bon support car il permet de **documenter** l'installation.

La macro à installer

Placez-la dans sa propre bibliothèque dans le document porteur. C'est cette bibliothèque qui sera installée.

La macro installeuse

Son rôle est de recopier la bibliothèque embarquée vers le conteneur global Mes Macros sur le poste. Installation-type d'une bibliothèque de type « code » :

```
Dim oSrcLib As Object      'bibli source (dans le document)
Dim oDestLib As Object    'bibli cible (dans 'Mes Macros')
Dim i As Integer
Dim SrcModules() As String
'On crée la bibliothèque cible si elle n'existe pas déjà
If Not GlobalScope.BasicLibraries.HasByName("bibli cible") Then
    GlobalScope.BasicLibraries.CreateLibrary("bibli cible")
End If
'On recopie les modules
If BasicLibraries.HasByName("bibli source") Then
    BasicLibraries.LoadLibrary("bibli source")
    oSrcLib = BasicLibraries.GetByName("bibli source")
    oDestLib = GlobalScope.BasicLibraries.GetByName("bibli cible")
    SrcModules = oSrcLib.GetElementNames()
    'Installation des modules 1 par 1
    i = LBound(SrcModules())
    Do While (i <= UBound(SrcModules()))
        If Not oDestLib.HasByName(SrcModules(i)) Then
            oDestLib.InsertByName(SrcModules(i), _
                oSrcLib.GetByName(SrcModules(i)))
        End If
        i = i + 1
    Loop
End If
```

Pour installer une bibliothèque de type « dialogue », remplacez BasicLibraries par DialogLibraries et GlobalScope.BasicLibraries par GlobalScope.DialogLibraries.

Aller plus loin : les extensions

L'étape suivante serait de transformer votre macro pour la diffuser sous forme d'une **extension**.

Cette tâche est ardue. L'outil **ExtensionCompiler** de Bernard Marcelly apporte une aide très précieuse dans ce travail : http://berma.pagesperso-orange.fr/Files_en/ExtensionCompiler.ott.

Crédits

Auteur : Jean-François Nifenecker - jean-francois.nifenecker@laposte.net

Nous sommes comme des nains assis sur des épaules de géants. Si nous voyons plus de choses et plus lointaines qu'eux, ce n'est pas à cause de la perspicacité de notre vue, ni de notre grandeur, c'est parce que nous sommes élevés par eux. (Bernard de Chartres [attr.])

Historique

Version	Date	Commentaires
2.0	06/04/2021	Mise en forme.

Licence

Cet aide-mémoire est placé sous licence

Creative Commons BY-SA v3 (fr)

Informations :

<https://creativecommons.org/licenses/by-sa/3.0/fr/>

