

XlsxToolBx / DocxToolBx

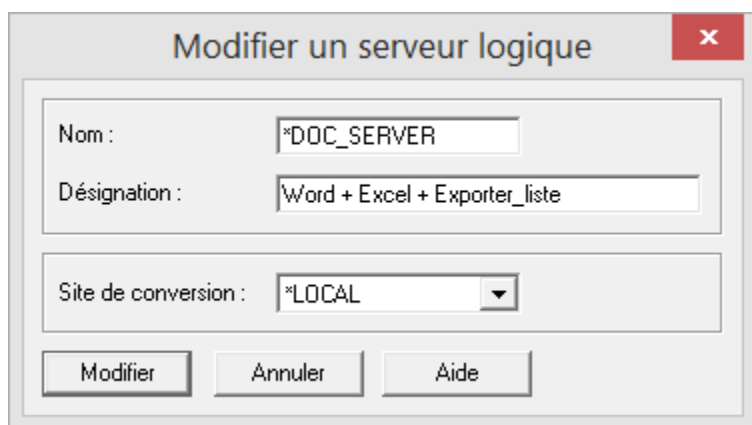
CONFIGURATION

Les XlsxToolBx et DocxToolBx passent par le Middleware java.

Il faut donc prévoir une configuration client/serveur spécifique.

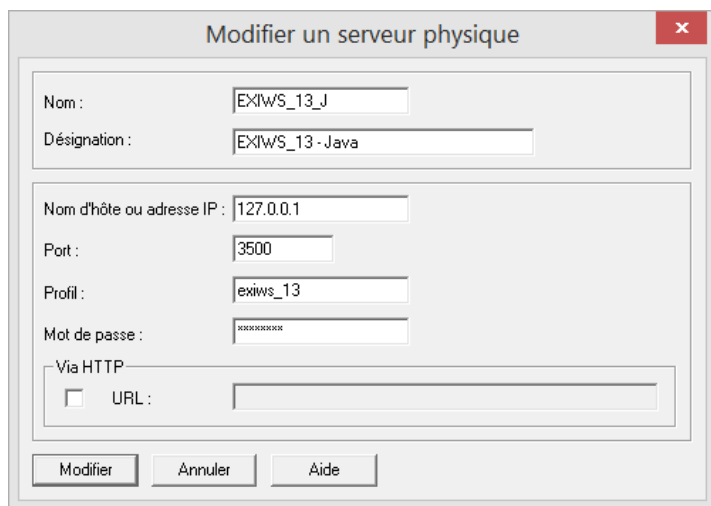
Le **MWCLIENT.INI** utilisé est celui qui se trouve dans le contexte d'exécution de votre programme (par défaut, au même endroit que les fichiers générés ou sous la racine d'Adélia pour les programmes Visual; ou sous la racine du site Cloud/Web, dans WEB-INF\Classes pour les programmes Cloud ou Web).

Créez un serveur logique *DOC_SERVER.



The dialog box titled "Modifier un serveur logique" has a red close button in the top right corner. It contains three input fields: "Nom :" with the value "*DOC_SERVER", "Désignation :" with the value "\Word + Excel + Exporter_liste", and "Site de conversion :" with a dropdown menu showing "*LOCAL". At the bottom, there are three buttons: "Modifier", "Annuler", and "Aide".

Créez un serveur physique sous le port 3500 (Java) :



The dialog box titled "Modifier un serveur physique" has a red close button in the top right corner. It contains several input fields: "Nom :" with the value "EXIWS_13_J", "Désignation :" with the value "EXIWS_13 - Java", "Nom d'hôte ou adresse IP :" with the value "127.0.0.1", "Port :" with the value "3500", "Profil :" with the value "exiws_13", and "Mot de passe :" with a masked value "XXXXXXXXXX". There is a checkbox labeled "Via HTTP" which is currently unchecked. Below it, there is a label "URL :" followed by an empty text field. At the bottom, there are three buttons: "Modifier", "Annuler", and "Aide".

Associez les 2 avec *NONE comme base de données :

Modifier une association

Serveur logique : *DOC_SERVER

Serveur physique : EXIWS_13_J

BD utilisée : *NONE

Adélia Web

Ignorer le contrôle de validation ☐

Modifier Annuler Aide

Le **MWSERVER.INI** utilisé est celui sous la racine d'Adélia, dans Javarun, du serveur défini dans la configuration du client.

Vous pouvez changer son emplacement, en spécifiant un chemin différent dans le CfgConfiguration.properties (qui se trouve sous %adeliws%\javarun).

Créez un utilisateur de mêmes nom et mot de passe que celui précisé au niveau de la définition du serveur physique. Il n'a pas besoin de contexte particulier :

Modifier un utilisateur

Profil exiws_13 ☒ Valide

Désignation : EXIWS_13

Mot de passe xxxxxxx

Contexte

Modifier Annuler Aide

Puis, arrêtez et redémarrez le démon Middleware (service Adélia Middleware Java).

EXEMPLE (XlsxToolBx)

```

CHARGER_DLL 'XlsxToolBx.dll'

SI Mode = 'OUVERTURE'
    nomFichier = 'Class_001.XLSX'
    APPELER_DLL 'XlsxToolBx' 'WorkbookOpen' nomFichier handleClasseur rc
FIN

SI Mode = 'CREATION'
    Format = 2
    APPELER_DLL 'XlsxToolBx' 'WorkbookCreate' Format handleClasseur rc
FIN

```

=> Le handleClasseur est le n° interne du classeur rendu par la fonction WorkbookCreate.

Cette fonction admet 2 paramètres :

- le type de classeur (en entrée)
- le n° interne (handle) du classeur (en sortie).
- le code retour de la fonction

```

* Créer la feuille
APPELER_DLL 'XlsxToolBx' 'SheetAdd' handleClasseur 'Feuille' 0 rc

```

=> Pour créer la 1ère feuille, mettez 0 comme position (1 ne passe pas car il n'y a pas encore de feuille).

```

* Ajouter des valeurs
APPELER_DLL 'XlsxToolBx' 'SelectionSetAlphaValueArray' handleClasseur plage LIGNEENTETE 50 NB_ELEMENTS rc

```

=> 50, car LIGNEENTETE est un tableau de ALPHA(50).

```

* Enregistrer
nomFichier = 'Class_002.XLSX'
APPELER_DLL 'XlsxToolBx' 'WorkbookSaveAs' handleClasseur nomFichier rc

```

EXEMPLE (DocxToolBx)

```

CHARGER_DLL 'DocxToolBx.dll'

SI Mode = 'CREATION'
  APPELER_DLL 'DocxToolBx' 'DocCreate' Modele handleDocument rc
FIN

SI Mode = 'OUVERTURE'
  APPELER_DLL 'DocxToolBx' 'DocOpen' Fichier handleDocument rc
FIN

* Récupérer le Handle de la partie principale
APPELER_DLL 'DocxToolBx' 'DocGetMainDocumentPart' handleDocument handleMainPart rc

* Insérer le premier paragraphe
WW_TXT_PARA = *BLANK
APPELER_DLL 'DocxToolBx' 'ParagraphInsert' handleMainPart _DOCX_POS_FIRST 0 *BLANK WW_TXT_PARA
handleParagraphe1 rc

```

=> Pour créer le 1er paragraphe, mettez 0 comme référence de position (1 ne passe pas car il n'y a pas encore de paragraphe).

*BLANK signifie que ce paragraphe aura comme style 'Normal'. Vous pouvez à la place, mettre le nom du style souhaité.

```

* Insérer un autre paragraphe
WW_TXT_PARA = *BLANK
APPELER_DLL 'DocxToolBx' 'ParagraphInsert' handleMainPart _DOCX_POS_AFTER handleParagraphe1 *BLANK WW_TXT_PARA
handleParagraphe2 rc

```

=> Vous pouvez utiliser le même handle de paragraphe. Vous devez utiliser un handle différent si vous voulez travailler sur l'un de ces paragraphes ultérieurement dans le programme.

```

* Changer le style
APPELER_DLL 'DocxToolBx' 'ParagraphSetStyle' handleParagraphe1 'Titrel' rc

* Exporter au format PDF
APPELER_DLL 'DocxToolBx' 'DocExportPDF' handleDocument 'D:\Temp\Test.pdf' rc

* Enregistrer
APPELER_DLL 'DocxToolBx' 'DocSave' handleDocument rc

* Fermer
APPELER_DLL 'DocxToolBx' 'DocClose' handleDocument rc

```

PLAGES (XlsxToolBx)

Les plages de cellules doivent être spécifiée sous la forme A1:C3 (colonne A ligne 1 à colonne C ligne 3).

La plage peut comporter le nom de la feuille de calcul référencée, la sélection est alors de la forme Feuil1!A1:C3.

Si le nom de la feuille de calcul est omis, la sélection est considérée comme relative à la feuille active du classeur.

Vous pouvez traiter plusieurs plages de cellule simultanément.

Dans ce cas, séparez les plages par le caractère ";". Par exemple : Feuil1!A1:C3;Feuil2!A1:C3.

Exemple : *modification de la fonte de la première cellule de la feuille Feuil1*

```
APPELER_DLL 'XlsxToolBx' 'StyleSetFont' handleClasseur 'Feuil1!A1' 'Comic Sans MS' rc
```

EN GÉNÉRAL

Vous n'êtes pas obligé d'avoir Excel / Word installé ni sur votre poste ni sur le serveur physique où tourne le démon Java.

- **DLL**

Il ne faut surtout pas oublier de charger la DLL au début.

- **Démon Java**

En cas d'erreur, vous pouvez aussi essayer en démarrant le démon Middleware Java en fenêtre.

Arrêtez le service Adélia Middleware Java.

Prenez une fenêtre de commande DOS.

Exécutez " cd %adeliws%\javarun ".

Puis, exécutez " daemon.bat ".

- **Chemins**

Le chemin est précisé par rapport à la machine où tourne le Middleware Java. Il doit donc être connu par ce poste et accessible (en lecture/écriture selon vos besoins) par l'utilisateur qui a démarré le Middleware java.

Pour les chemins des classeurs Excel et documents Word , il faut doubler les anti-slash, car on est dans un contexte Java.

Pour les chemins réseau, utilisez plutôt les chemins UNC (\\...) et évitez les connecteurs réseau (P:\...). De plus, le service Adélia Middleware Java ne doit pas être démarré avec "Compte système local", mais avec un profil réseau qui ait les droits d'accès et d'écriture sur l'axe réseau où se trouvent vos classeurs / documents.

Exemple avec un chemin local :

```
nomFichier = 'D:\\Temp\\Class_001.XLSX'
```

Exemple avec un chemin réseau :

```
nomFichier = '\\\\LeServeur\\Partage\\Class_002.XLSX'
```

Par défaut, si vous ne précisez pas de chemin au fichier, le .docx / .xlsx sera créé / lu sous la racine d'Adélia dans Javarun (contexte d'exécution du démon Middleware Java).

- **Format des classeurs / documents**

Les XlsxToolBx et DocxToolBx gèrent les classeurs / documents au format .xlsx / .docx.

L'ouverture de classeur / document créés avec une ancienne version du Pack Office tente de convertir les .xls / .doc dans ce format. Or, le processus de conversion est partiel et des pertes de données ou de format peuvent survenir.

Il faudrait, dans un premier temps et par sécurité, reprendre vos .xls / .doc dans une version récente d'Excel / Word, avant de pouvoir les gérer avec la XlsxToolBx / DocxToolBx.