

Client DepotFox, propriétés et méthodes

Le client DepotFox est constitué de 3 classes, stockées dans la bibliothèque `foxhisto.vcx`

La classe de base est `_DepotFox`, elle est basée sur `Custom`. Elle n'est pas destinée à être instanciée comme objet directement utilisable.

Les deux classes qui sont instanciées comme objet client sont `DepotFox` et `Options`. La classe `Option` est instanciée automatiquement par la classe `DepotFox`, mais peut aussi être instanciée en tant qu'objet autonome.

Les tableaux suivants présentent uniquement les propriétés et méthodes spécifiques des 3 classes (les propriétés et méthodes de `Custom` qui ne sont pas modifiées ou utilisées ne figurent pas dans ces tableaux)

_DepotFox

Propriétés (`_DepotFox`)

Nom	Description
ChaineConnexion	la chaine de connexion utilisée pour accéder à la base de données de DépotFox. <i>Valeur par défaut :</i> "Driver={SQL Server Native Client 11.0}; Server=Machine\InstanceSQL;Database=master;Trusted_Connection=yes;"
IdleTimeout	la durée de validité d'une connexion, en minutes. Les connexions sont désactivées après ce délai. <i>Valeur par défaut :</i> 10
IsConnectStringValid	Indique si la chaine de connexion est valide ; passe à .T. après vérification de la chaine de connexion. <i>Valeur par défaut :</i> .F.
NomDatabase	le nom de la database de DépotFox <i>Valeur par défaut :</i> ZZtest
NomInstanceSQL	le nom de l'instance SQL hébergeant la base de données pour DépotFox <i>Valeur par défaut :</i> machine\SQLexpress
TypesFichiersFox	les noms descriptifs des types de fichiers traités par Dépotfox. Ces noms doivent correspondre aux types déclarés dans la procédure stockée <code>Merge_FoxTable_depuis_depotpourfox</code> <i>Valeur par défaut :</i> données, database, projet, form, classe, rapport, étiquette, menu, programme, menusprog, queryprog
TypesFichiersFoxCourts	les noms abrégés des types de fichiers traités par Dépotfox <i>Valeur par défaut :</i> dbf, dbc, pjx, scx, vcx, frx, lbx, mnx, prg, mpr, qpr
TypesFichiersNonFox[1, 3]	Le tableau des noms des types de fichiers (autres que fox) traités par DépotFox. Cette liste provient de la table des types de la database de base <i>Pas de valeur par défaut</i>

Méthodes (_DepotFox)

Nom	Description
Init	<p>LPARAMETERS <code>lcChaineConnexion AS STRING, lcNomDatabase AS STRING, lcNomInstanceSQL AS STRING</code></p> <p>définit <code>NomDatabase</code> et <code>NomInstanceSQL</code> si on reçoit la chaîne de connexion, définit <code>ChaineConnexion</code> si on reçoit <code>NomDatabase</code> et <code>NomInstanceSQL</code></p>
GetTypesNonFox	Récupère sur le serveur les listes des noms et extensions des types de fichiers non fox traités par DepotFox, et actualise la propriété tableau <code>TypesFichiersNonFox[1,3]</code>
SetChaineConnexion	Définit la chaîne de connexion à partir du nom de l'instance et du nom de la database, et vérifie cette chaîne
SetNomDatabase	définit le nom du database à partir d'une chaîne de connexion valide
SetNomInstanceSQL	définit le nom de l'instance à partir d'une chaîne de connexion valide
VerifieListeTypesFichiersFox	<p>LPARAMETERS <code>Type_Table_IN AS STRING, SansFiltre AS Boolean</code></p> <p>Vérifie si la liste reçue en paramètre (chaîne de mots séparés par des virgules) ne contient que des types fox autorisés, substitue les abréviations par le nom complet, met en forme, et retourne un booléen</p>

DepotFox

Propriétés (DepotFox)

Nom	Description
CheminDossier	<p>le dossier virtuel pour déposer les fichiers fox en vue de leur archivage</p> <p><i>Valeur par défaut</i> : <code>\\Machine\sqlexpress2016\ZZ_FS\depotfox</code></p>
DateDerniereSauvegardeSQLarchive	<p>Si vous ne sauvegardez pas votre base de données SQL, le fichier LOG augmentera indéfiniment!</p> <p><i>Valeur par défaut</i> : <code>=date(1900,01,01)</code></p>
DateDerniereSauvegardeSQLdepot	<p>Si vous ne sauvegardez pas votre base de données SQL, le fichier LOG augmentera indéfiniment!</p> <p><i>Valeur par défaut</i> : <code>=date(1900,01,01)</code></p>
EtatDossierVirtuel	<p>Affiche l'état du dossier virtuel de dépôt partagé. S'il est Inactif, le dossier n'est plus visible ni accessible sur le réseau (son contenu est conservé)</p> <p><i>Valeur par défaut</i> : <code>Actif</code></p>
NbFichiersDansArchive	<p>le nombre de fichiers dans la table d'archive</p> <p><i>Valeur par défaut</i> : <code>0</code></p>
NbFichiersDansDepot	<p>le nombre de fichiers dans le dossier de dépôt</p> <p><i>Valeur par défaut</i> : <code>0</code></p>

NbFichiersDansHistorique	le nombre de fichiers dans la table d'historique <i>Valeur par défaut : 0</i>
NbFichiersExtraits	Le nombre de fichiers qui viennent d'être extraits depuis la table d'historique vers un emplacement physique <i>Valeur par défaut : 0</i>
NbNouveauxFichiersDansArchive	le nombre de fichiers archivés par la dernière opération d'archivage depuis le dépôt <i>Valeur par défaut : 0</i>
NbNouveauxFichiersDansHistorique	le nombre de fichiers historisés par la dernière opération d'archivage depuis le dépôt <i>Valeur par défaut : 0</i>
TailleLOGarchive	la taille du fichier de log de la base de données SQL d'archivage <i>Valeur par défaut : 0</i>
TailleLOGdepot	la taille du fichier de log de la base de données SQL de dépôt <i>Valeur par défaut : 0</i>
TailleMDFarchive	la taille du fichier de données de la base de données SQL d'archivage <i>Valeur par défaut : 0</i>
TailleMDFdepot	la taille du fichier de données de la base de données SQL de dépôt <i>Valeur par défaut : 0</i>

Méthodes (DepotFox)

Nom	Description
Init	<code>LPARAMETERS lcChaineConnexion AS STRING, lcNomDatabase AS STRING, lcNomInstanceSQL AS STRING</code> En surcharge de la classe de base, instancie un objet Options en tant que sous-objet contenu
ArchiveVersHistoriqueFichiers	Archive dans la table d'historique les fichiers non-fox présents dans le dossier virtuel partagé "dépotfox" dont le type figure dans la table des types et actualise les propriétés NbNouveauxFichiersDansArchive et NbNouveauxFichiersDansHistorique
ArchiveVersHistoriqueFox	Archive dans la table d'historique les fichiers fox présents dans le dossier virtuel partagé "dépotfox" et actualise les propriétés NbNouveauxFichiersDansArchive et NbNouveauxFichiersDansHistorique
ArchiveVersHistoriqueTous	Archive dans les deux tables d'historique depuis le répertoire de dépôt et actualise les propriétés NbNouveauxFichiersDansArchive et NbNouveauxFichiersDansHistorique
DeleteTypeFichierNonFox	<code>LPARAMETERS m.Ext AS STRING , m.ConnexionHandle AS INTEGER</code> Supprime de la table SQL des types de fichiers la ligne correspondant au paramètre obligatoire ext

ExtraitFichier	<p><code>LPARAMETERS lcDestination AS STRING</code></p> <p>Extrait le fichier sélectionné depuis l'historique, et le recrée à l'emplacement passé en paramètre (si le dossier de destination n'est pas spécifié, on en crée un dans le répertoire courant)</p>
GetEtatDossierVirtuel	Récupère sur le serveur SQL le statut SINGLE_USER ou MULTI_USER de la database de dépôt, et actualise la propriété locale <code>EtatDossierVirtuel</code>
GetHistoriqueFox	<p><code>LPARAMETERS DateHeureDebut AS DATETIME, DateHeureFin AS DATETIME</code></p> <p>Obtient les fichiers Fox depuis la table d'historique, et les place dans un curseur VFP pour extraction par la méthode <code>ExtraitFichier</code>. Les paramètres sont optionnels : pas de paramètres= tout l'historique 2 paramètres de valeurs différentes = FROM... TO 2 paramètres de même valeur = AS OF 1 seul paramètre = FROM... TO complété automatiquement</p>
GetHistoriqueNonFox	<p><code>LPARAMETERS DateHeureDebut AS DATETIME, DateHeureFin AS DATETIME</code></p> <p>Obtient les fichiers non Fox depuis la table d'historique, et les place dans un curseur VFP pour extraction par la méthode <code>ExtraitFichier</code>. Les paramètres sont optionnels : pas de paramètres= tout l'historique 2 paramètres de valeurs différentes = FROM... TO 2 paramètres de même valeur = AS OF 1 seul paramètre = FROM... TO complété automatiquement</p>
InsertTypeFichierNonFox	<p><code>LPARAMETERS m.Ext AS STRING, m.Nom AS STRING, m.Actif AS Boolean, m.ConnexionHandle AS INTEGER</code></p> <p>Ajoute un nouvel enregistrement dans la table SQL des types de fichiers (paramètres requis <code>Ext</code>, <code>Nom</code>, optionnel <code>Actif</code>)</p>
SauvegardeSQL	Effectue une sauvegarde des 2 bases de données SQL à l'emplacement défini par défaut sur SQL Server
SetAllSQLprop	interroge la base de données SQL et met à jour les propriétés de l'objet instancié
SetCheminDossier	récupère le nom du dossier virtuel partagé de dépôtfox pour la database utilisée, et actualise la propriété <code>NbFichiersDansDepot</code>
SetDateDerniereSauvegardeSQL	Actualise la propriété <code>DateDerniereSauvegardeSQL</code> avec la valeur obtenue sur le serveur
SetDossierVirtuel	<p><code>LPARAMETERS Actif_ou_Inactif AS STRING</code></p> <p>Active ou Désactive (met hors ligne) le dossier virtuel partagé, actualise les propriétés <code>EtatDossierVirtuel</code> et <code>CheminDossier</code></p>
SetNbFichiersDansArchive	Obtient le nombre de fichiers dans les tables d'archivage (sans l'historique), et met à jour la propriété <code>NbFichiersDansArchive</code>
SetNbFichiersDansDepot	Obtient le nombre de fichiers Fox dans la FileTable SQL de dépôt, et met à jour la propriété <code>NbFichiersDansDepot</code>

SetNbFichiersDansHistorique	Obtient le nombre de fichiers stockés dans l'historique des tables d'archivage, met à jour la propriété NbFichiersDansHistorique
SetTailleFichiers	Actualise les propriétés TailleMDFarchive, TailleMDFdepot, TailleLOGarchive, et TailleLOGdepot avec les informations recueillies sur le serveur SQL
SetTypesFichiersNonFox	crée un curseur local TYPES_NON_FOX à partir de la propriété tableau TypesFichiersNonFox[1,3], pour préparer les insert/update/delete qui seront envoyés à la table des types sur le serveur
UpdateTousTypesFichiersNonFox	met à jour la table des types sur le serveur, à partir des modifications faites localement sur le curseur TYPES_NON_FOX
UpdateTypeFichierNonFox	<pre> LPARAMETERS m.OldExt AS STRING, ; && ancienne extension m.NewExt AS STRING, ; && nouvelle extension m.OldNom AS STRING, ; && ancien nom m.NewNom AS STRING, ; && nouveau nom m.OldAct AS Boolean, ; && ancien actif m.NewAct AS Boolean, ; && nouveau actif m.ConnexionHandle AS INTEGER </pre> <p>Met à jour dans la table SQL des types de fichiers la ligne correspondant au paramètre obligatoire OldExt, avec un WhereType=3 (champ clé et champs modifiés)</p>
ZapHistoriqueFichiers	Supprime définitivement tout le contenu de l'historique Fichiers non Fox
ZapHistoriqueFox	Supprime définitivement tout le contenu de l'historique Fox

Options

Méthodes (Options)

Nom	Description
DeleteProjet	<pre> LPARAMETERS m.Niveau AS STRING, ConnexionHandle AS INTEGER </pre> <p>Supprime le projet correspondant au paramètre obligatoire Niveau dans la table SQL des projets</p>
DeleteProjetFichier	<pre> LPARAMETERS m.chemin AS STRING, m.TypeFichier AS STRING, m.NomFichier AS STRING, m.niveau AS STRING, m.ConnexionHandle AS INTEGER </pre> <p>Supprime le rattachement d'un fichier NON Fox à un projet ; les 4 premiers paramètres sont obligatoires (chemin, TypeFichier, NomFichier, niveau).</p>
DeleteProjetFoxTable	<pre> LPARAMETERS m.chemin AS STRING, m.tyetable AS STRING, m.nom_table AS STRING, m.niveau AS STRING, m.ConnexionHandle AS INTEGER </pre> <p>Supprime le rattachement d'un fichier Fox à un projet ; les 4 premiers paramètres sont obligatoires (chemin, TypeTable, NomTable, niveau).</p>

DeleteVersionFichier	<p>LPARAMETERS m.Chemin AS STRING, m.TypeFichier AS STRING, m.NomFichier AS STRING, m.Numero AS STRING, m.Debut AS DATETIME, m.ConnexionHandle AS INTEGER</p> <p>Supprime une version de la table SQL des versions ; les 5 premiers paramètres sont obligatoires (Chemin, TypeFichier, NomFichier, Numero, Debut).</p>
DeleteVersionFoxTable	<p>LPARAMETERS m.Chemin AS STRING, m.TypeTable AS STRING, m.NomTable AS STRING, m.Numero AS STRING, m.Debut AS DATETIME, m.ConnexionHandle AS INTEGER</p> <p>Supprime une version de la table SQL des versions ; les 5 premiers paramètres sont obligatoires (Chemin, TypeTable, NomTable, Numero, Debut).</p>
GetFichiersFoxVersions	Lit la vue de jointure des fichiers fox et des versions avec tous les détails, la place dans un curseur VFP
GetFichiersNonFoxVersions	Lit la vue de jointure des fichiers non fox et des versions avec tous les détails, la place dans un curseur VFP
GetProjets	Lit la table des projets hiérarchisés, et la place dans un curseur VFP
GetTousFichiersVersions	Lit la vue de jointure de tous les fichiers (fox et nonfox) et des versions avec tous les détails, la place dans un curseur VFP
InsertProjet	<p>LPARAMETERS m.nom_projet AS STRING , m.Niveau AS STRING , m.commentaire AS STRING, ConnexionHandle AS INTEGER</p> <p>Ajoute un nouveau projet dans la table SQL des projets ; les 2 premiers paramètres sont obligatoires (nom_projet, Niveau)</p>
InsertProjetFichier	<p>LPARAMETERS m.Chemin AS STRING, m.TypeFichier AS STRING, m.NomFichier AS STRING, m.Niveau AS STRING, ConnexionHandle AS INTEGER</p> <p>Ajoute un rattachement d'un fichier NON fox à un projet ; les 4 premiers paramètres sont obligatoires (Chemin, TypeFichier, NomFichier, Niveau).</p>
InsertProjetFoxTable	<p>LPARAMETERS m.Chemin AS STRING, m.TypeTable AS STRING, m.NomTable AS STRING, m.Niveau AS STRING, ConnexionHandle AS INTEGER</p> <p>Ajoute un rattachement d'un fichier NON fox à un projet ; les 4 premiers paramètres sont obligatoires (Chemin, TypeTable, NomTable, Niveau).</p>
InsertVersionFichier	<p>LPARAMETERS m.Chemin AS STRING, m.TypeFichier AS STRING, m.NomFichier AS STRING, m.Numero AS STRING, m.Debut AS DATETIME, m.ConnexionHandle AS INTEGER</p> <p>Ajoute un nouveau numéro pour un fichier NON fox dans la table SQL des Versions ; les 4 premiers paramètres sont obligatoires (Chemin, TypeFichier, NomFichier, Numero).</p>

InsertVersionFoxTable	<p>LPARAMETERS m.Chemin AS STRING, m.TypeTable AS STRING, m.NomTable AS STRING, m.Numero AS STRING, m.Debut AS DATETIME, m.ConnexionHandle AS INTEGER</p> <p>Ajoute un nouveau numéro de version pour un fichier fox dans la table SQL des Versions ; les 4 premiers paramètres sont obligatoires (Chemin, TypeTable, NomTable, Numero).</p>
IsHierarchyValid	<p>LPARAMETERS Hierarchy AS STRING</p> <p>Vérifie (et corrige si possible) que la chaîne passée en paramètre représente bien un HierarchyID de SQL Server ; le paramètre est obligatoire. Une représentation correcte est de la forme /1/1/22/nn/</p>
IsNumVersionValid	<p>LPARAMETERS NumVersion AS STRING</p> <p>Vérifie (et corrige si possible) que la chaîne passée en paramètre au format calculé dans la table des versions représente bien un HierarchyID de SQL Server ; le paramètre est obligatoire. Une représentation correcte est de la forme 1.1.22.nnn</p>
SearchTablesFichiersProjets	<p>LPARAMETERS Nom_Projet_EGALE AS STRING, Type_IN AS STRING, Nom_LIKE AS STRING</p> <p>Lit la vue des tables, fichiers et projets hiérarchisés, et la place dans un curseur VFP ; tous les paramètres sont optionnels.</p>
SetProjetsFox	<p>Lit la vue de jointure des fichiers Fox et des projets, la place dans un curseur VFP pour pouvoir être modifiée ; ATTENTION ! la connexion au serveur reste ouverte, pensez à personnaliser le Timeout selon vos besoins!</p>
SetProjetsNonFox	<p>Lit la vue de jointure des fichiers NON Fox et des projets, la place dans un curseur VFP pour pouvoir être modifiée ; ATTENTION ! la connexion au serveur reste ouverte, pensez à personnaliser le Timeout selon vos besoins!</p>
SetVersionsFox	<p>Lit la vue des versions des fichiers fox, et la place dans un curseur VFP pour pouvoir être modifiée ; ATTENTION ! la connexion au serveur reste ouverte, pensez à personnaliser le Timeout selon vos besoins!</p>
SetVersionsNonFox	<p>Lit la vue des versions des fichiers NON fox, et la place dans un curseur VFP pour pouvoir être modifiée ; ATTENTION ! la connexion au serveur reste ouverte, pensez à personnaliser le Timeout selon vos besoins!</p>
UpdateProjet	<p>LPARAMETERS m.nom_projet AS STRING , m.Niveau AS STRING , m.commentaire AS STRING, m.WhereNiveau AS STRING , ConnexionHandle AS INTEGER</p> <p>Met à jour un projet dans la table SQL des projets ; les 4 premiers paramètres sont obligatoires (nom_projet, Niveau, commentaire, WhereNiveau).</p>

UpdateProjetFichier	<p><code>LPARAMETERS m.Chemin AS STRING, m.TypeFichier AS STRING, m.NomFichier AS STRING, m.Niveau AS STRING, m.WhereNiveau AS STRING, ConnexionHandle AS INTEGER</code></p> <p>Met à jour un rattachement d'un fichier NON fox à un projet ; les 4 premiers paramètres sont obligatoires (Chemin, TypeFichier, NomFichier, Niveau).</p>
UpdateProjetFoxTable	<p><code>LPARAMETERS m.Chemin AS STRING, m.TypeTable AS STRING, m.NomTable AS STRING, m.Niveau AS STRING, m.WhereNiveau AS STRING, ConnexionHandle AS INTEGER</code></p> <p>Met à jour un rattachement d'un fichier fox à un projet ; les 4 premiers paramètres sont obligatoires (Chemin, TypeTable, NomTable, Niveau).</p>
UpdateTousFichiersFoxProjets	<p>Met à jour la table de jointure entre fichiers fox et projets (rattachement d'un fichier à un ou plusieurs projets) à partir du curseur VFP obtenu par SetProjetsFox</p>
UpdateTousFichiersFoxVersions	<p>Met à jour la table des versions personnalisées des fichiers fox à partir du curseur VFP obtenu par SetVersionsFox</p>
UpdateTousFichiersNonFoxProjets	<p>Met à jour la table de jointure entre fichiers NON fox et projets (rattachement d'un fichier à un ou plusieurs projets) à partir du curseur VFP obtenu par SetProjetsNonFox</p>
UpdateTousFichiersNonFoxVersions	<p>Met à jour la table des versions personnalisées des fichiers NON fox à partir du curseur VFP obtenu par SetVersionsNonFox</p>
UpdateTousProjets	<p>Met à jour la table SQL des projets à partir du curseur VFP obtenu par GetProjets</p>
UpdateVersionFichiers	<p><code>LPARAMETERS m.OldNumero AS STRING, ; && l'ancien numéro m.NewNumero AS STRING, ; && le nouveau numéro m.OldDebut AS DATETIME, ; && l'ancienne date de début de la version m.NewDebut AS DATETIME, ; && la nouvelle date de début de la version m.WhereChemin AS STRING, ; m.WhereTypeFichier AS STRING, ; m.WhereNomFichier AS STRING</code></p> <p>Modifie le numéro et/ou la date de début d'une version dans la table SQL des Versions ; les paramètres OldNumero, OldDebut, WhereChemin, WhereTypeFichier, et WhereNomFichier sont obligatoires, au moins 1 des paramètres NewNumero et NewDebut doit être renseigné.</p>
UpdateVersionFoxTable	<p><code>LPARAMETERS m.OldNumero AS STRING, ; && l'ancien numéro m.NewNumero AS STRING, ; && le nouveau numéro m.OldDebut AS DATETIME, ; && l'ancienne date de début de la version m.NewDebut AS DATETIME, ; && la nouvelle date de début de la version m.WhereChemin AS STRING, ; m.WhereTypeTable AS STRING, ; m.WhereNomTable AS STRING</code></p> <p>Modifie le numéro et/ou la date de début d'une version dans la table SQL des Versions ; les paramètres OldNumero, OldDebut, WhereChemin, WhereTypeTable, et WhereNomTable sont obligatoires, au moins 1 des paramètres NewNumero et NewDebut doit être renseigné.</p>