

Manuel de l'utilisateur

# Extension sODBC



**Somi·t**   
Small Office Management Integrated Tools  
*Solutions de gestion intégrée pour l'entreprise*

# MANUEL D'UTILISATION DE L'EXTENSION sODBC POUR FILEMAKER PRO

Par Henri et Mathieu Frappier



## TABLE DES MATIÈRES

TABLE DES MATIÈRES .....	3
PRÉAMBULE.....	8
INTRODUCTION .....	8
Résumé .....	8
Installation de l'extension .....	8
Installation des fonctions personnalisées .....	9
Fichier d'exemples.....	10
<b>FONCTIONS EXTERNES .....</b>	<b>11</b>
Liste des fonctions externes de l'extension sODBC.....	11
sODBC_Open( "" ; ConnectionString ).....	11
sODBC_Close( "" ).....	11
sODBC_Query( "" ; Query ).....	12
sODBC_NonQuery( "" ; Query ).....	12
sODBC_QueryA( "" ; Query ; Get ( FileName ) ; ScriptName ).....	12
sODBC_NonQueryA( "" ; Query ; Get ( FileName ) ; ScriptName ).....	13
sODBC_FieldSeparator .....	13
sODBC_RecordSeparator.....	13
sODBC_GetSchema( "" ; SchemaName ; AttributeFilter ).....	14
sODBC_Register( Name ; Key ) .....	14
sODBC_Version( { Format } ) .....	15
Encodage Texte .....	15
Codes d'erreurs et leur signification pour l'extension sODBC .....	16
<b>FONCTIONS PERSONNALISÉES – ACOMBA.....</b>	<b>17</b>
Liste des fonctions personnalisées pour le logiciel Acomba .....	17



Aco.AccountStatement.ASAmountDue ( CuNumber ; InInvoiceNumber ) .....	18
Aco.Charter.ChCategory ( ChNumber ) .....	18
Aco.Charter.ChDescription ( ChNumber ) .....	18
Aco.ControlCustomer.Interest .....	18
Aco.Customer.Exist ( CuNumber ) .....	18
Aco.Customer.NewUpdate ( CuNumber ; CuSortKey ; CuName ; CuCareOf ; CuAddress ; CuCity ; CuPostalCode ; CuISOCountryCode ; CuPhoneNumber1 ; CuPhoneExtension1 ; CuPhoneNumber2 ; CuEMail ; CuWebAddress ; CuReceivable ; CuTaxGroupNumber ; CuTermNumber ; CuInterest ; CuMarginLimit ; CuComment ) .....	19
Aco.CustomerPaymentLine.RecCardPos ( CPInvoiceARCP ) .....	19
Aco.InvoiceAR.Exist ( InCustomerSupplierNumber ; InInvoiceNumber ) .....	19
Aco.InvoiceAP.Exist ( InSupplierNumber ; InInvoiceNumber ) .....	20
Aco.Supplier.Exist ( SuNumber ) .....	20
Aco.Supplier.NewUpdate ( SuNumber ; SuSortKey ; SuName ; SuCareOf ; SuAddress ; SuCity ; SuPostalCode ; SuISOCountryCode ; SuPhoneNumber1 ; SuPhoneExtension1 ; SuPhoneNumber2 ; SuEMail ; SuWebAddress ; SuPurchaseAccountNumber ; SuTaxGroupNumber ; SuPaymentTermNumber ; SuComments ) .....	20
Aco.SupplierTrans.Balance ( SuNumber ; InInvoiceNumber ) .....	21
Aco.TransAR.Begin .....	21
Aco.TransAR.Cancel .....	21
Aco.TransAR.End .....	21
Aco.TransARHeader.New ( InInvoiceNumber ; InCustomerSupplierNumber ; InTransactionType ; InDate ; InAllPaid ; InTaxGroupNumber ; InReference ; InDescription ; TANumPosts ) .....	21
Aco.TransARDetail.Update ( PACharterNumber ; PAAmount ; PaDescription ; PAIsTax ; TANum ) .....	22
Aco.TransAP.Begin .....	22
Aco.TransAP.Cancel .....	22
Aco.TransAP.End .....	22



Aco.TransAPHeader.New ( InInvoiceNumber ; InSupplierNumber ; InType ; InDate ; InAmount ; InSubTotal ; InReference ; InDescription ; TAnumPosts ) .....	22
Aco.TransAPDetail.Update ( PACharterNumber ; PAAmount ; PaDescription ; PAIsTax ; TaNum ) .....	23
Aco.Utills.CityProvince ( City ; Province ) .....	23
Aco.Utills.DebitCreditAR ( Amount ; TransactionType ; TaNum ) .....	23
Aco.Utills.DebitCreditAP ( Amount ; TransactionType ; TaNum ) .....	23
Aco.Utills.DecimalSeparator ( Number ) .....	23
Aco.Utills.ESQ ( Text ) .....	24
Aco.Utills.ISOCountryCodeEn ( Country ) .....	24
Aco.Utills.ISOCountryCodeFr ( Country ) .....	24
Aco.Utills.Phone ( Text ) .....	24
<b>FONCTIONS PERSONNALISÉES – SAGE 50 (Simple Comptable) .....</b>	<b>25</b>
Liste des fonctions personnalisées pour le logiciel Sage 50 .....	25
s50.sASUserld .....	26
s50sASOrgld.....	26
s50.Customer.Exist (sName).....	26
s50.Customer.New ( sName ; sCntcName ; sStreet1 ; sStreet2 ; sCity ; sProvState ; sCountry ; sPostalZip ; sPhone1 ; sPhone2 ; sFax ; dCrLimit ; sEmail ; sWebSite ; nLangPref ; dtSince ; sMemo).....	26
s50.Customer.Update ( lld ; sName ; sCntcName ; sStreet1 ; sStreet2 ; sCity ; sProvState ; sCountry ; sPostalZip ; sPhone1 ; sPhone2 ; sFax ; sEmail ; sWebSite ; sMemo ) .....	27
s50.Invoice.New ( sName ; lCustomrld ; dtDate ; sReference ; sTaxCode ; dSubTotal ; dTotal ; dTotalTaxA ; lAccountTaxA ; dTotalTaxB ; lAccountTaxB ; sComment ) .....	28
s50.InvoiceDetail.New ( sInvoiceNumber ; sPartCode ; lQuantity ; dPrice ; sTaxCode ; dAmountTaxA ; lAccountTaxA ; dAmountTaxB ; lAccountTaxB ) .....	28
s50.InvoiceDetailAccount.New ( sInvoiceNumber ; lAccount ; sDescription ; lQuantity ; dPrice ; sTaxCode ; dAmountTaxA ; lAccountTaxA ; dAmountTaxB ; lAccountTaxB ) .....	29



s50.Invoice.Freight ( sInvoiceNumber ; dFreightAmount ; sFreightTaxCode ; dFreightTaxA ; IFreightAccountTaxA ; dFreightTaxB ; IFreightAccountTaxB ) .....	29
s50.Invoice.Commit .....	29
s50.Invoice.Rollback .....	30
s50.InvoicePayment.New ( sInvoiceNumber ; lCustomrId ; dtDate ; dAmount ; lType ; lBankAccount ; sReference ; sComment ) .....	30
s50.Purchase.New ( sName ; lVendorId ; dtDate ; sReference ; sDescription ; sTaxCode ; dSubTotal ; lChargeAccount ; dTotalTaxA ; lAccountTaxA ; dTotalTaxB ; lAccountTaxB ; sNonTaxableDescription ; dNonTaxableAmount ; lNonTaxableAccount ; dTotal ; sComment) .....	30
s50.Vendor.exist ( sName ) .....	31
s50.Vendor.New ( sName ; sCntcName ; sStreet1 ; sStreet2 ; sCity ; sProvState ; sCountry ; sPostalZip ; sPhone1 ; sPhone2 ; sFax ; sEmail ; sWebSite ; nLangPref ; dtSince ; sMemo) .	31
s50.Vendor.Update ( lId ; sName ; sCntcName ; sStreet1 ; sStreet2 ; sCity ; sProvState ; sCountry ; sPostalZip ; sPhone1 ; sPhone2 ; sFax ; sEmail ; sWebSite ; sMemo ) .	31
s50.Utills.CheckSqlSyntax ( Sql ) .....	32
s50.Utills.Date ( fmDate ) .....	32
s50.Utills.DateFirstOfLastMonth ( fmDate ) .....	32
s50.Utills.ESQ ( Text ) .....	32
s50.Utills.FormatDecimal ( Number ) .....	33
s50.Utills.getNextId ( TableId ) .....	33
s50.Utills.getNextInvoiceNumber. ....	33
s50.Utills.getNextReceiptNumber .....	33
s50.Utills.getInvoiceBalance ( sInvoiceNumber ; lCustomrId ) .	34
s50.Utills.getPurchaseBalance ( sInvoiceNumber ; lVendorId ) .....	34
s50.Utills.getTaxCodeId ( sTaxCode ) .....	34
s50.Utills.getTaxInfo ( sTaxCode ; lTaxAccountNumber ) .	34
s50.Utills.setNextId ( TableId ) .....	34



s50.Utills.setNextInvoiceNumber ( Number ).....	34
s50.Utills.setNextReceiptNumber ( Number ).....	34
<b>RÉFÉRENCES</b> .....	<b>35</b>
Notes techniques.....	35
Dépannage (« Troubleshooting »).....	36
À propos de FileMaker, Inc.....	36



## PRÉAMBULE

- Ce manuel assume que vous possédez les connaissances de base concernant la plateforme de développement de bases de données FileMaker Pro, de l'utilisation de ses extensions et de ses fonctions externes et personnalisées.
- Ce manuel, ainsi que les fichiers qui y sont inclus, sont distribués sous licence d'emploi et ne peuvent être copiés, photocopiés, reproduits, traduits ou convertis en format électronique, en totalité ou en partie, sans autorisation écrite obtenue au préalable auprès de Somi-t inc.
- Le terme « FileMaker » utilisé dans les descriptions de ce manuel réfère généralement aux versions des logiciels FileMaker Pro et FileMaker Pro Advanced.
- Somi-t est fier d'être membre de la FileMaker Business Alliance (FBA) depuis 2006.
- Somi-t inc. n'assume aucune responsabilité découlant d'erreurs ou d'inexactitude provenant du présent manuel.
- N'hésitez pas à communiquer avec nous à [plugin@somi-t.com](mailto:plugin@somi-t.com)
- Site Web : <http://www.somi-t.com> ou <http://www.SuivezLeGuide.ca>
- Cette version du manuel a été mise à jour le 24 novembre 2014.

## INTRODUCTION

### Résumé

- **L'extension sODBC permet à votre solution FileMaker Pro de communiquer avec n'importe quelle source ODBC de données installée sur votre système Windows ou Mac OS X.**
- Après l'installation de l'extension sODBC, il vous sera facile d'utiliser les onze fonctions externes qui s'affichent automatiquement dans le dialogue de calcul de FileMaker Pro. Vous pourrez alors utiliser les fonctions SQL telles que <Select>, <Update>, <Delete> et <Insert> ;
- L'extension sODBC vous permettra de contrôler les actions d'ouverture et de fermeture de la connexion ODBC par FileMaker contrairement à l'action de script <Exécuter SQL> qui ouvre et ferme la connexion automatiquement sans vous consulter ;
- Vous pourrez utiliser la syntaxe du langage SQL proposée par le pilote ODBC de la base de données ciblée.

### Installation de l'extension sous Windows

- Résumé des étapes d'installation :
  - 1) Le pilote sODBC.fmx doit être installé dans le répertoire C:\Program Files\FileMaker\FileMaker Pro [version]\Extensions\ avant de démarrer le logiciel FileMaker Pro.
  - 2) Le pilote ODBC de l'application doit être installé sur l'ordinateur conformément aux recommandations du fabricant.
  - 3) Vérifier si la source des données est visible dans le Panneau de configuration \ Source de données ODBC \ Onglet Pilotes  
[EN : Control Panel\ODBC Data Source\ onglet Drivers]



- 4) Utiliser User DSN ou System DSN pour ajouter (ADD) la source ODBC à la liste.
- 5) La source ODBC est maintenant disponible pour sODBC à l'aide des fonctions externes de FileMaker.

## Installation de l'extension sous Mac OS X

- Résumé des étapes d'installation :
  - 1) Installer le iODBC Framework dans Mac OS X disponible à cette adresse <http://www.iodbc.org/downloads/iODBC/iodbc-sdk-3.52.7-macosx-10.5.dmg>
  - 2) Le pilote sODBC.fplugin doit être installé dans le répertoire /Application/FileMaker Pro [version]/Extensions/ avant de démarrer le logiciel FileMaker Pro.
  - 3) Le pilote ODBC de l'application doit être installé sur l'ordinateur conformément aux recommandations du fabricant.
  - 4) Bien que le gestionnaire de connexion ODBC "ODBC Manager" est disponible dans le dossier /Application/Utilities...  
Nous vous recommandons d'installer "Openlink ODBC Administrator" disponible à cette adresse <http://www.iodbc.org/dataspace/iodbc/wiki/iODBC/Downloads>
  - 5) Utiliser User DSN ou System DSN pour ajouter (ADD) la source ODBC à la liste.
  - 6) La source ODBC est maintenant disponible pour sODBC à l'aide des fonctions externes de FileMaker.

## Installation des fonctions personnalisées

- Exclusivement pour les utilisateurs de FileMaker Pro Advanced ! Vous pouvez importer ou encore copier et coller nos fonctions personnalisées dans un différent fichier à partir de nos fichiers FileMaker inclus avec l'ensemble correspondant au logiciel cible.
- Il n'est pas possible pour les utilisateurs de FileMaker Pro d'importer les fonctions personnalisées.
- Étapes à exécuter pour importer les fonctions personnalisées :
  - 1) Ouvrez le fichier destination dans lequel vous souhaitez importer les fonctions personnalisées ;
  - 2) Dans le menu Fichier, sélectionnez > Gérer > Fonctions personnalisées ;
  - 3) Dans la boîte de dialogue Gérer les fonctions personnalisées, cliquez sur Importer ;
  - 4) Ouvrez le fichier qui contient les fonctions personnalisées à importer ;
  - 5) Si un nom de compte et un mot de passe vous sont demandés à l'ouverture du fichier, saisissez-les dans la boîte de dialogue Mot de passe, puis cliquez sur OK ;
  - 6) Dans la boîte de dialogue Importer les fonctions personnalisées, sélectionnez les fonctions à importer puis cliquez sur OK ;
  - 7). La boîte de dialogue Récapitulatif d'importation s'affiche ;
  - 8) Cliquez sur Ouvrir fichier journal pour afficher le fichier journal d'importation ou cliquez sur OK pour fermer la boîte de dialogue Récapitulatif d'importation ;
  - 9) Fermez la boîte de dialogue Gérer les fonctions personnalisées.



## Fichier d'exemples

- Les bases de données FileMaker Pro contenant les fonctions personnalisées, ne sont fournies que dans les ensembles de fonctions personnalisées vendus séparément de l'extension sODBC pour FileMaker. Par exemple, avec l'ensemble des fonctions personnalisées pour Acomba et Sage 50.



## FONCTIONS EXTERNES

### Liste des fonctions externes de l'extension sODBC

- sODBC\_Open( "" ; ConnectionString )
- sODBC\_Close( "" )
- sODBC\_Query( "" ; Query )
- sODBC\_NonQuery( "" ; Query )
- sODBC\_QueryA( "" ; Query ; Get ( FileName ) ; ScriptName )
- sODBC\_NonQueryA( "" ; Query ; Get ( FileName ) ; ScriptName )
- sODBC\_FieldSeparator
- sODBC\_RecordSeparator
- sODBC\_GetSchema( "" ; SchemaName ; AttributeFilter )
- sODBC\_Register( Name ; Key )
- sODBC\_Version( { Format } )

### sODBC\_Open( "" ; ConnectionString )

- Description : Ouvre la connexion avec la source ODBC de données.
- Paramètre "ConnectionString" : Paramètre de connexion à la source ODBC de données.
  - Exemple :
    - sODBC\_Open( "" ; "DSN=MonAcomba;UID=Supervisor;PWD=DEMO;" )
    - sODBC\_Open( "" ; "DSN=MonFp7;UID=test;PWD=test;" )
    - sODBC\_Open( "" ; "DSN=Northwind;" )
    - sODBC\_Open( "" ; "Driver={Microsoft Access Driver (\*.mdb)};DBQ=c:\bin\Northwind.mdb" )
- Résultat : Renvoie la valeur 1 si la connexion a été établie avec succès sinon renvoie le code d'erreur précédé par "!!".

### sODBC\_Close( "" )

- Description : Ferme la connexion avec la source ODBC de données.
- Paramètre : Non utilisé, laisser vide avec "".
- Résultat : Renvoie la valeur 1 si la connexion a été fermée avec succès sinon renvoie le code d'erreur précédé par "!!".



### sODBC\_Query( "" ; Query )

- Description : Exécute une requête SQL du type SELECT.
- Paramètre "" : Non utilisé sous Windows, laisser vide avec "".
  - Pour la version Mac, vous pouvez changer l'encodage du texte d'entrée et de sortie. Voir la rubrique "Encodage Texte" dans cette section du document.
- Paramètre "Query" : Requête SQL à exécuter.
  - Exemple :
    - sODBC\_Query("", "SELECT \* FROM Contacts WHERE Status='Actif'")
- Résultat : Renvoie le résultat de la requête. Si plus d'un champ est sélectionné, le séparateur de champ sera utilisé. Également, si plus d'un enregistrement est sélectionné, le séparateur d'enregistrement sera utilisé. Voir sODBC\_FieldSeparator et sODBC\_RecordSeparator.

### sODBC\_NonQuery( "" ; Query )

- Description : Exécute une requête SQL du type INSERT, UPDATE ou DELETE.
- Paramètre "" : Non utilisé, laisser vide avec "".
- Paramètre "Query" : Requête SQL à exécuter.
  - Exemple
    - sODBC\_NonQuery("", "UPDATE Contacts SET Status='Inactif' WHERE ID=101")
    - sODBC\_NonQuery("", "INSERT INTO Contacts (Prénom, Nom) VALUES ('Linda', 'L'Amour')")
- Résultat : Renvoie le nombre d'enregistrement modifié par la requête.

### sODBC\_QueryA( "" ; Query ; Get ( FileName ) ; ScriptName )

#### WINDOWS SEULEMENT

- Description : Comme sODBC\_Query mais la requête s'exécute dans son propre fil d'exécution (thread), libérant ainsi l'interface utilisateur.
- Paramètre "" : Non utilisé, laisser vide avec "".
- Paramètre "Query" : Requête SQL à exécuter.
- Paramètre "Get ( FileName )" : Nom du fichier où se trouve le script défini par ScriptName.
- Paramètre "ScriptName" : Nom du script FileMaker à exécuter après l'exécution de la requête SQL.
- Résultat : Aucun



- Note : Pour accéder au résultat de la requête à l'intérieur du script ScriptName, utilisez la fonction Get ( ScriptParameter ) de FileMaker.

### sODBC\_NonQueryA( "" ; Query ; Get ( FileName ) ; ScriptName )

#### WINDOWS SEULEMENT

- Description : Comme sODBC\_NonQuery mais la requête s'exécute dans son propre fil d'exécution (thread), libérant ainsi l'interface utilisateur.
- Paramètre "" : Non utilisé, laisser vide avec "".
- Paramètre "Query" : Requête SQL à exécuter.
- Paramètre "Get ( FileName )" : Nom du fichier où se trouve le script défini par ScriptName.
- Paramètre "ScriptName" : Nom du script FileMaker à exécuter après l'exécution de la requête SQL.
- Résultat : Aucun
- Note : Pour accéder au résultat de la requête à l'intérieur du script ScriptName, utilisez la fonction Get ( ScriptParameter ) de FileMaker.

### sODBC\_FieldSeparator

- Description : Renvoie le séparateur de champ utilisé et/ou assigne le séparateur de champ à utiliser.
- Paramètre : Chaîne de caractère(s) à utiliser comme séparateur.
  - Exemple :
    - sODBC\_FieldSeparator
    - sODBC\_FieldSeparator( Char(31) )
    - sODBC\_FieldSeparator("<MonSeparateur1>")
- Résultat : Renvoie le séparateur de champ utilisé.
- Note : Par défaut, le séparateur de champ est le code ASCII 31, équivalent à la fonction Char(31) de FileMaker.

### sODBC\_RecordSeparator

- Description : Renvoie le séparateur d'enregistrement utilisé et/ou assigne le séparateur d'enregistrement à utiliser.
- Paramètre : Chaîne de caractère(s) à utiliser comme séparateur.
  - Exemple :
    - sODBC\_RecordSeparator



- sODBC\_RecordSeparator( Char(30) )
  - sODBC\_RecordSeparator("<MonSeparateur2>")
- Résultat : Renvoie le séparateur d'enregistrement utilisé.
- Note : Par défaut, le séparateur d'enregistrement par défaut est le code ASCII 30, équivalent à la fonction Char(30) de FileMaker.

### sODBC\_GetSchema( "" ; SchemaName ; AttributeFilter )

#### WINDOWS SEULEMENT

- Description : Permet d'accéder aux informations de schéma de la source de données.
- Paramètre "SchemaName" : Nom du schéma à interroger.
  - Valeurs possibles :
    - "Restrictions"
    - "Tables"
    - "Columns"
    - "Views"
    - "ViewColumns"
    - "Users"
    - "DataTypes"
    - "ReservedWords"
    - "Procedures"
    - "ProcedureParameters"
    - "Indexes"
    - "IndexColumns"
    - "ForeignKeys"
    - "DataSourceInformation"
    - "MetaDataCollections"
- Paramètre "AttributeFilter" : Filtrer les résultats selon le nom de l'attribut.
  - Exemple
    - sODBC\_GetSchema("", "Tables"; "TABLE\_NAME")
- Résultat : Renvoie les informations de schéma sous forme de liste de valeur. Si le filtre d'attributs n'est pas utilisé, les attributs seront séparés par le séparateur de champ. Voir sODBC\_FieldSeparator.

### sODBC\_Register( Name ; Key )

- Description : Enregistre l'extension. Son état passe du mode démonstrateur à celui d'enregistré.
- Paramètre "Name" : Nom de l'organisation que vous avez fourni à Somi-t lors de l'achat de l'extension.
- Paramètre "Key" : Clé d'enregistrement fournie par Somi-t.



## sODBC\_Version( { Format } )

- Description : Retourne le numéro de la version et/ou le nom de l'extension.
- Paramètre : "", "short", "long"

## Encodage Texte

La version Mac du plug-in sODBC offre la possibilité de modifier l'encodage du texte.

L'encodage par défaut utilisé par le plug-in sODBC version Mac est ISO\_8859\_1. Pour modifier l'encodage du texte d'entrée, utilisez le mot clé "inputEncoding=" suivi de l'encodage désiré. Pour modifier l'encodage du texte de sortie, utilisez le mot clé "outputEncoding=" suivi de l'encodage désiré.

Exemples avec sODBC\_Query() :

- `sODBC_Query("inputEncoding=UTF8";"SELECT * FROM Province WHERE Nom = 'Québec'")`
- `sODBC_Query("outputEncoding=UTF8";"SELECT * FROM Province WHERE Nom = 'Québec'")`
- `sODBC_Query("inputEncoding=UTF8;outputEncoding=UTF8";"SELECT * FROM Province WHERE Nom = 'Québec'")`

Exemple avec sODBC\_NonQuery() :

- `sODBC_NonQuery("inputEncoding=UTF8";"UPDATE Province SET nom = 'Québec' WHERE nom = 'Quebec'")`



Liste des encodages disponibles :

Native	UTF8	ASCII_DOS	ASCII_Windows	ASCII_Mac
ISO_8859_1	ISO_8859_2	ISO_8859_3	ISO_8859_4	ISO_8859_5
ISO_8859_6	ISO_8859_7	ISO_8859_8	ISO_8859_9	ISO_8859_15
ShiftJIS_Mac	ShiftJIS_Win	Korean_Mac	Korean_Win	Korean_Johab
ChineseTrad_Mac	ChineseTrad_Win	ChineseSimp_Mac	ChineseSimp_Win	Cyrillic_Mac
Cyrillic_Win	CentralEurope_Mac	EasternEurope_Win	Turkish_Mac	Turkish_Win
Baltic_Win	Arabic_Mac	Arabic_Win	Greek_Mac	Greek_Win
Hebrew_Mac	Hebrew_Win			

### Codes d'erreurs et leur signification pour l'extension sODBC

- !!-101 : Mauvaise licence, (\_Open)
- !!-1 : Non connecté (\_Open, \_Query, \_NonQuery, \_GetSchema)
- !!-2 : Paramètre absent (\_Open, \_Query, \_NonQuery, \_GetSchema)
- !!-3 : Erreur spécifique de connexion ouverte (\_Open)
- !!-4 : Erreur spécifique de commande pour une requête (\_Query)
- !!-5 : Erreur spécifique de lecture pour une requête (\_Query)
- !!-6 : Erreur spécifique de commande pour une requête (\_NonQuery)
- !!-7 : Erreur spécifique de lecture pour une requête (\_NonQuery)
- !!-8 : Erreur spécifique de fermeture de connexion (\_Close)
- !!-10 : Erreur de schéma (\_GetSchema)



## FONCTIONS PERSONNALISÉES – ACOMBA

### Liste des fonctions personnalisées pour le logiciel Acomba

- Aco.AccountStatement.ASAmountDue ( CuNumber ; InInvoiceNumber )
- Aco.Charter.ChCategory ( ChNumber )
- Aco.Charter.ChDescription ( ChNumber )
- Aco.ControlCustomer.Interest
- Aco.Customer.Exist ( CuNumber )
- Aco.Customer.NewUpdate ( CuNumber ; CuSortKey ; CuName ; CuCareOf ; CuAddress ; CuCity ; CuPostalCode ; CuISOCountryCode ; CuPhoneNumber1 ; CuPhoneExtension1 ; CuPhoneNumber2 ; CuEMail ; CuWebAddress ; CuReceivable ; CuTaxGroupNumber ; CuTermNumber ; CuInterest ; CuMarginLimit ; CuComment )
- Aco.CustomerPaymentLine.RecCardPos ( CPIInvoiceARCP )
- Aco.InvoiceAP.Exist ( InSupplierNumber ; InInvoiceNumber )
- Aco.InvoiceAR.Exist ( InCustomerSupplierNumber ; InInvoiceNumber )
- Aco.Supplier.Exist ( SuNumber )
- Aco.Supplier.NewUpdate ( SuNumber ; SuSortKey ; SuName ; SuCareOf ; SuAddress ; SuCity ; SuPostalCode ; SuISOCountryCode ; SuPhoneNumber1 ; SuPhoneExtension1 ; SuPhoneNumber2 ; SuEMail ; SuWebAddress ; SuPurchaseAccountNumber ; SuTaxGroupNumber ; SuPaymentTermNumber ; SuComments )
- Aco.SupplierTrans.Balance ( SuNumber ; InInvoiceNumber )
- Aco.TransAP.Begin
- Aco.TransAP.Cancel
- Aco.TransAP.End
- Aco.TransAPDetail.Update ( PACharterNumber ; PAAmount ; PaDescription ; PAIsTax ; TaNum )
- Aco.TransAPHeader.New ( InInvoiceNumber ; InSupplierNumber ; InType ; InDate ; InAmount ; InSubTotal ; InReference ; InDescription ; TANumPosts )
- Aco.TransAR.Begin
- Aco.TransAR.Cancel
- Aco.TransAR.End
- Aco.TransARDetail.Update ( PACharterNumber ; PAAmount ; PaDescription ; PAIsTax ; TaNum )
- Aco.TransARHeader.New ( InInvoiceNumber ; InCustomerSupplierNumber ; InTransactionType ; InDate ; InAllPaid ; InTaxGroupNumber ; InReference ; InDescription ; TANumPosts )
- Aco.Utils.CityProvince ( City ; Province )



- Aco.Utills.DebitCreditAR ( Amount ; TransactionType ; TaNum )
- Aco.Utills.DebitCreditAP ( Amount ; TransactionType ; TaNum )
- Aco.Utills.DecimalSeparator ( Number )
- Aco.Utills.ESQ ( Text )
- Aco.Utills.ISOCountryCodeEn ( Country )
- Aco.Utills.ISOCountryCodeFr ( Country )
- Aco.Utills.Phone ( Text )

#### Aco.AccountStatement.ASAmountDue ( CuNumber ; InInvoiceNumber )

- Description : Renvoie le solde d'une facture
- Paramètre "CuNumber" : Numéro du client
- Paramètre "InInvoiceNumber" : Numéro de la facture

#### Aco.Charter.ChCategory ( ChNumber )

- Description : Renvoie la catégorie du compte comptable
- Paramètre "ChNumber" : Numéro de compte comptable

#### Aco.Charter.ChDescription ( ChNumber )

- Description : Renvoie la description du compte comptable
- Paramètre "ChNumber" : Numéro de compte comptable

#### Aco.ControlCustomer.Interest

- Description : Renvoie le taux d'intérêt par défaut

#### Aco.Customer.Exist ( CuNumber )

- Description : Renvoie le numéro du client interne à Acomba (RecCardPos) si le client existe (>0)
- Paramètre "CuNumber" : Numéro du client



**Aco.Customer.NewUpdate ( CuNumber ; CuSortKey ; CuName ; CuCareOf ; CuAddress ; CuCity ; CuPostalCode ; CuISOCountryCode ; CuPhoneNumber1 ; CuPhoneExtension1 ; CuPhoneNumber2 ; CuEMail ; CuWebAddress ; CuReceivable ; CuTaxGroupNumber ; CuTermNumber ; CuInterest ; CuMarginLimit ; CuComment )**

- Description : Ajout ou mise à jour des informations d'un client
- Paramètre "CuNumber" : Numéro du client
- Paramètre "CuSortKey" : Clé de tri du client
- Paramètre "CuName" : Nom du client
- Paramètre "CuCareOf" : Nom du contact (A/S)
- Paramètre "CuAddress" : Adresse du client
- Paramètre "CuCity" : Ville du client ( voir Aco.Utils.CityProvince() )
- Paramètre "CuPostalCode" : Code postal du client
- Paramètre "CuISOCountryCode" : Pays du client ( code à deux lettres, voir Aco.Utils.ISOCountryCodeFr() )
- Paramètre "CuPhoneNumber1" : Numéro de téléphone #1 du client
- Paramètre "CuPhoneExtension1" : Extension du téléphone #1
- Paramètre "CuPhoneNumber2" : Numéro de téléphone #2 du client
- Paramètre "CuEMail" : Courriel du client
- Paramètre "CuWebAddress" : Site Internet du client
- Paramètre "CuReceivable" : Numéro du compte à recevoir pour ce client
- Paramètre "CuTaxGroupNumber" : Numéro du groupe de taxes Acomba pour ce client
- Paramètre "CuTermNumber" : Terme de paiement pour ce client
- Paramètre "CuMarginLimit" : Marge de crédit pour ce client
- Paramètre "CuComment" : Commentaire pour ce client

**Aco.CustomerPaymentLine.RecCardPos ( CPIInvoiceARCP )**

- Description : Renvoie le numéro d'un paiement interne a Acomba (RecCardPos)
- Paramètre "CPIInvoiceARCP" : Numéro de facture interne à Acomba (RecCardPos)

**Aco.InvoiceAR.Exist ( InCustomerSupplierNumber ; InInvoiceNumber )**

- Description : Renvoie le numéro de facture client interne à Acomba (>0)
- Paramètre "InCustomerSupplierNumber" : Numéro du client
- Paramètre "InInvoiceNumber" : Numéro de la facture client



### Aco.InvoiceAP.Exist ( InSupplierNumber ; InInvoiceNumber )

- Description : Renvoie le numéro de facture fournisseur interne à Acoma (>0)
- Paramètre "InSupplierNumber" : Numéro du fournisseur
- Paramètre "InInvoiceNumber" : Numéro de la facture fournisseur

### Aco.Supplier.Exist ( SuNumber )

- Description : Renvoie le numéro du fournisseur interne à Acoma (RecCardPos) si le fournisseur existe (>0)
- Paramètre "SuNumber" : Numéro du fournisseur

### Aco.Supplier.NewUpdate ( SuNumber ; SuSortKey ; SuName ; SuCareOf ; SuAddress ; SuCity ; SuPostalCode ; SuISOCountryCode ; SuPhoneNumber1 ; SuPhoneExtension1 ; SuPhoneNumber2 ; SuEMail ; SuWebAddress ; SuPurchaseAccountNumber ; SuTaxGroupNumber ; SuPaymentTermNumber ; SuComments )

- Description : Ajout ou mise à jour des informations d'un fournisseur
- Paramètre "SuNumber" : Numéro du fournisseur
- Paramètre "SuSortKey" : Clé de tri du fournisseur
- Paramètre "SuName" : Nom du fournisseur
- Paramètre "SuCareOf" : Nom du contact (A/S)
- Paramètre "SuAddress" : Adresse du fournisseur
- Paramètre "SuCity" : Ville du fournisseur ( voir Aco.Utils.CityProvince() )
- Paramètre "SuPostalCode" : Code postal du fournisseur
- Paramètre "SuISOCountryCode" : Pays du fournisseur ( code à deux lettres, voir Aco.Utils.ISOCountryCodeFr() )
- Paramètre "SuPhoneNumber1" : Numéro de téléphone #1 du fournisseur
- Paramètre "SuPhoneExtension1" : Extension du téléphone #1
- Paramètre "SuPhoneNumber2" : Numéro de téléphone #2 du fournisseur
- Paramètre "SuEMail" : Courriel du fournisseur
- Paramètre "SuWebAddress" : Site Internet du fournisseur
- Paramètre "SuPurchaseAccountNumber" : Numéro de compte payable pour ce fournisseur
- Paramètre "SuTaxGroupNumber" : Numéro du groupe de taxes Acoma pour ce fournisseur
- Paramètre "SuPaymentTermNumber" : Terme de paiement du fournisseur
- Paramètre "SuComments" : Commentaire pour ce fournisseur



### Aco.SupplierTrans.Balance ( SuNumber ; InInvoiceNumber )

- Description : Retourne la balance à payer pour une transaction fournisseur
- Paramètre "SuNumber" : Numéro du fournisseur
- Paramètre "InInvoiceNumber" : Numéro de la transaction fournisseur

### Aco.TransAR.Begin

- Description : Débute une transaction de type client

### Aco.TransAR.Cancel

- Description : Annule une transaction précédemment ouverte avec Aco.TransAR.Begin

### Aco.TransAR.End

- Description : Termine une transaction précédemment ouverte avec Aco.TransAR.Begin

### Aco.TransARHeader.New ( InInvoiceNumber ; InCustomerSupplierNumber ; InTransactionType ; InDate ; InAllPaid ; InTaxGroupNumber ; InReference ; InDescription ; TANumPosts )

- Description : Créer l'entête d'une transaction/facture client
- Paramètre "InInvoiceNumber" : Numéro de la facture client
- Paramètre "InCustomerSupplierNumber" : Numéro du client
- Paramètre "InTransactionType" : Type de transaction
- Paramètre "InDate" : Date de la transaction
- Paramètre "InAllPaid" : 1 Si la facture est payée en entier ou 0 sinon
- Paramètre "InTaxGroupNumber" : Numéro du groupe de taxe Acoma
- Paramètre "InReference" : Référence de la facture
- Paramètre "InDescription" : Description de la facture
- Paramètre "TANumPosts" : Nombre de détails de facture, généralement 1 détail pour le compte à recevoir, 1 détail par item plus 1 détail par taxe.

Ex. : TANumPosts = 5, pour une facture de 1.00 \$ + taxes

1 – Compte à recevoir 1.14 \$

2 – Pomme 1 x 0.50 \$

3 - Banane 1 x 0.50 \$

4 – TPS 0.05 \$ (5 %)

5 – TVQ 0.09 \$ (8.5 %)



### **Aco.TransARDetail.Update ( PACHarterNumber ; PAAmount ; PaDescription ; PAIsTax ; TaNum )**

- Description : Créer un détail de facture pour une transaction/facture client
- Paramètre "PACHarterNumber" : Numéro du compte comptable
- Paramètre "PAAmount" : Montant
- Paramètre "PaDescription" : Description ( généralement la description du compte comptable, voir Aco.Charter.ChDescription() )
- Paramètre "PAIsTax" : 1 si ce détail est une taxe, 0 si non
- Paramètre "TaNum" : Numéro séquentiel du détail de facture

### **Aco.TransAP.Begin**

- Description : Débute une transaction de type fournisseur

### **Aco.TransAP.Cancel**

- Description : Annule une transaction précédemment ouverte avec Aco.TransAP.Begin

### **Aco.TransAP.End**

- Description : Termine une transaction précédemment ouverte avec Aco.TransAP.Begin

### **Aco.TransAPHeader.New ( InInvoiceNumber ; InSupplierNumber ; InType ; InDate ; InAmount ; InSubTotal ; InReference ; InDescription ; TAnumPosts )**

- Description : Crée l'entête d'une transaction/facture fournisseur
- Paramètre "InInvoiceNumber" : Numéro de la facture fournisseur
- Paramètre "InSupplierNumber" : Numéro du fournisseur
- Paramètre "InType" : Type de transaction
- Paramètre "InDate" : Date de la transaction
- Paramètre "InAmount" : Montant total de la transaction
- Paramètre "InSubTotal" : Sous-total de la transaction
- Paramètre "InReference" : Référence de la facture
- Paramètre "InDescription" : Description de la facture



- Paramètre "TANumPosts" : Nombre de détails de facture, généralement 1 détail pour le compte payable, 1 détail par item plus 1 détail par taxe.  
(Voir exemple de Aco.TransARHeader.New() )

#### Aco.TransAPDetail.Update ( PACHarterNumber ; PAAmount ; PaDescription ; PAIsTax ; TaNum )

- Description : Crée un détail de facture pour une transaction/facture fournisseur
- Paramètre "PACHarterNumber" : Numéro du compte comptable
- Paramètre "PAAmount" : Montant
- Paramètre "PaDescription" : Description ( généralement la description du compte comptable, voir Aco.Charter.ChDescription() )
- Paramètre "PAIsTax" : 1 si ce détail est une taxe, 0 si non
- Paramètre "TaNum" : Numéro séquentiel du détail de facture

#### Aco.Utills.CityProvince ( City ; Province )

- Description : Renvoie une chaîne de caractères composée de la ville et de la province
- Paramètre "City" : Ville
- Paramètre "Province" : Province ( deux lettres, ex. : QC )

#### Aco.Utills.DebitCreditAR ( Amount ; TransactionType ; TaNum )

- Description : Formate le montant selon le type de facture client
- Paramètre "Amount" : Montant
- Paramètre "TransactionType" : Type de transaction
- Paramètre "TaNum" : Numéro séquentiel du détail de facture client

#### Aco.Utills.DebitCreditAP ( Amount ; TransactionType ; TaNum )

- Description : Formate le montant selon le type de facture fournisseur
- Paramètre "Amount" : Montant
- Paramètre "TransactionType" : Type de transaction
- Paramètre "TaNum" : Numéro séquentiel du détail de facture fournisseur

#### Aco.Utills.DecimalSeparator ( Number )

- Description : Convertit le format de nombres utilisés par FileMaker par un format compatible pour la requête SQL ( 100,50 => 100.50 , les nombres décimaux doivent être séparés par un point "." )



- Paramètre "Number" : Montant

#### Aco.Utils.ESQ ( Text )

- Description : Convertit les apostrophes ' en double apostrophe '' pour la requête SQL
- Paramètre "Text" : Texte

#### Aco.Utils.ISOCountryCodeEn ( Country )

- Description : Renvoie le code ISO à deux caractères pour le nom d'un pays en anglais
- Paramètre "Country" : Nom du pays ( ex. : "Germany" => "DE" )

#### Aco.Utils.ISOCountryCodeFr ( Country )

- Description : Renvoie le code ISO à deux caractères pour le nom d'un pays en français
- Paramètre "Country" : Nom du pays ( ex. : "Allemagne" => "DE" )

#### Aco.Utils.Phone ( Text )

- Description : Renvoie un numéro de téléphone à 7 ,10 ou 11 caractères
- Paramètre "Text" : numéro de téléphone



## FONCTIONS PERSONNALISÉES – SAGE 50 (Simple Comptable)

### Liste des fonctions personnalisées pour le logiciel Sage 50

- s50.sASUserId
- s50.sASOrgId
- s50.Customer.Exist ( sName )
- s50.Customer.New ( sName ; sCntcName ; sStreet1 ; sStreet2 ; sCity ; sProvState ; sCountry ; sPostalZip ; sPhone1 ; sPhone2 ; sFax ; dCrLimit ; sEmail ; sWebSite ; nLangPref ; dtSince ; sMemo )
- s50.Customer.Update ( IId ; sName ; sCntcName ; sStreet1 ; sStreet2 ; sCity ; sProvState ; sCountry ; sPostalZip ; sPhone1 ; sPhone2 ; sFax ; sEmail ; sWebSite ; sMemo )
- s50.Invoice.New ( sName ; ICustomrId ; dtDate ; sReference ; sTaxCode ; dSubTotal ; dTotal ; dTotalTaxA ; IAccountTaxA ; dTotalTaxB ; IAccountTaxB ; sComment )
- s50.InvoiceDetail.New ( sInvoiceNumber ; sPartCode ; IQuantity ; dPrice ; sTaxCode ; dAmountTaxA ; IAccountTaxA ; dAmountTaxB ; IAccountTaxB )
- s50.InvoiceDetailAccount.New ( sInvoiceNumber ; IAccount ; sDescription ; IQuantity ; dPrice ; sTaxCode ; dAmountTaxA ; IAccountTaxA ; dAmountTaxB ; IAccountTaxB )
- s50.Invoice.Freight ( sInvoiceNumber ; dFreightAmount ; sFreightTaxCode ; dFreightTaxA ; IFreightAccountTaxA ; dFreightTaxB ; IFreightAccountTaxB )
- s50.Invoice.Commit
- s50.Invoice.Rollback
- s50.InvoicePayment.New ( sInvoiceNumber ; ICustomrId ; dtDate ; dAmount ; IType ; IBankAccount ; sReference ; sComment ) Aco.InvoiceAR.Exist ( InCustomerSupplierNumber ; InInvoiceNumber )
- s50.Purchase.New ( sName ; IVendorId ; dtDate ; sReference ; sDescription ; sTaxCode ; dSubTotal ; IChargeAccount ; dTotalTaxA ; IAccountTaxA ; dTotalTaxB ; IAccountTaxB ; sNonTaxableDescription ; dNonTaxableAmount ; INonTaxableAccount ; dTotal ; sComment )
- s50.Vendor.Exist ( sName )
- s50.Vendor.New ( sName ; sCntcName ; sStreet1 ; sStreet2 ; sCity ; sProvState ; sCountry ; sPostalZip ; sPhone1 ; sPhone2 ; sFax ; sEmail ; sWebSite ; sTaxCode ; nLangPref ; dtSince ; sMemo )
- s50.Vendor.Update ( IId ; sName ; sCntcName ; sStreet1 ; sStreet2 ; sCity ; sProvState ; sCountry ; sPostalZip ; sPhone1 ; sPhone2 ; sFax ; sEmail ; sWebSite ; sMemo )
- s50.Utils.CheckSqlSyntax ( Sql )
- s50.Utils.Date ( fmDate )
- s50.Utils.DateFirstOfLastMonth ( fmDate )
- s50.Utils.ESQ ( Text )
- s50.Utils.FormatDecimal ( Number )



- s50.Utills.getNextId ( TableId )
- s50.Utills.getNextInvoiceNumber
- s50.Utills.getNextReceiptNumber
- s50.Utills.getInvoiceBalance ( sInvoiceNumber ; ICustomrId )
- s50.Utills.getPurchaseBalance ( sInvoiceNumber ; IVendorId )
- s50.Utills.getTaxCodeId ( sTaxCode )
- s50.Utills.getTaxInfo ( sTaxCode ; ITaxAccountNumber )
- s50.Utills.setNextId ( TableId )
- s50.Utills.setNextInvoiceNumber ( Number )
- s50.Utills.setNextReceiptNumber ( Number )

### s50.sASUserId

- Description : Fonction à modifier avec le nom de l'utilisateur que vous-voulez utiliser.

### s50.sASOrgId

- Description : Fonction à modifier avec le nom de l'organisation que vous-voulez utiliser.

### s50.Customer.Exist ( sName )

- Description : Renvoie le numéro unique du client dans Sage 50 si il existe sinon renvoie "".
- Paramètre "sName" : Nom de la compagnie du client dans Sage 50.

### s50.Customer.New ( sName ; sCntcName ; sStreet1 ; sStreet2 ; sCity ; sProvState ; sCountry ; sPostalZip ; sPhone1 ; sPhone2 ; sFax ; dCrLimit ; sEmail ; sWebSite ; nLangPref ; dtSince ; sMemo )

- Description : Ajout d'un client. Renvoie le numéro unique du client dans Sage 50.
- Paramètre "sName" : Nom de la compagnie.
- Paramètre "sCntcName" : Nom du contact principale de la compagnie.
- Paramètre "sStreet1" : Adresse de la compagnie.
- Paramètre "sStreet2" : Adresse de la compagnie.
- Paramètre "sCity" : Nom de la ville.
- Paramètre "sProvState" : Nom de la province.
- Paramètre "sCountry" : Nom du pays.



- Paramètre "sPostalZip" : Code postal.
- Paramètre "sPhone1" : Numéro de téléphone #1.
- Paramètre "sPhone2" : Numéro de téléphone #2.
- Paramètre "sFax" : Numéro de fax.
- Paramètre "dCrLimit" : Limite de crédit.
- Paramètre "sEmail" : Adresse courriel.
- Paramètre "sWebSite" : Adresse du site web.
- Paramètre "nLangPref" : Langue préférée. (0=Anglais, 1=Espagnol, 2=Français)
- Paramètre "dtSince" : Date d'ajout de ce client à Sage 50.
- Paramètre "sMemo" : Notes.

#### s50.Customer.Update ( lId ; sName ; sCntcName ; sStreet1 ; sStreet2 ; sCity ; sProvState ; sCountry ; sPostalZip ; sPhone1 ; sPhone2 ; sFax ; sEmail ; sWebSite ; sMemo )

- Description : Mise à jour des informations générale d'un client.
- Paramètre "lId" : Numéro unique du client dans Sage 50.
- Paramètre "sName" : Nom de la compagnie.
- Paramètre "sCntcName" : Nom du contact principale de la compagnie.
- Paramètre "sStreet1" : Adresse de la compagnie.
- Paramètre "sStreet2" : Adresse de la compagnie.
- Paramètre "sCity" : Nom de la ville.
- Paramètre "sProvState" : Nom de la province.
- Paramètre "sCountry" : Nom du pays.
- Paramètre "sPostalZip" : Code postal.
- Paramètre "sPhone1" : Numéro de téléphone #1.
- Paramètre "sPhone2" : Numéro de téléphone #2.
- Paramètre "sFax" : Numéro de fax.
- Paramètre "sEmail" : Adresse courriel.
- Paramètre "sWebSite" : Adresse du site web.
- Paramètre "sMemo" : Notes.



### s50.Invoice.New ( sName ; lCustomrId ; dtDate ; sReference ; sTaxCode ; dSubTotal ; dTotal ; dTotalTaxA ; lAccountTaxA ; dTotalTaxB ; lAccountTaxB ; sComment )

- Description : Crée l'entête d'une transaction/facture client et retourne le numéro de la facture utilisée par Sage 50.
- Paramètre "sName" : Nom de la compagnie du client.
- Paramètre "lCustomrId" : Numéro unique du client dans Sage 50.
- Paramètre "dtDate " : Date de la transaction.
- Paramètre "sReference" : Référence externe.
- Paramètre "sTaxCode" : Code de taxe à utiliser.
- Paramètre "dSubTotal" : Sous-total de la transaction.
- Paramètre "dTotal" : Total de la transaction.
- Paramètre "dTotalTaxA" : Montant de la taxe A.
- Paramètre "lAccountTaxA" : Numéro de compte de la taxe A.
- Paramètre "dTotalTaxB" : Montant de la taxe B.
- Paramètre "lAccountTaxB" : Numéro de compte de la taxe B.
- Paramètre "sComment" : Notes.
- Attention : le 'Commit' n'est pas automatique. Vous devez utiliser les fonctions s50.Invoice.Commit() ou s50.invoice.Rollback().

### s50.InvoiceDetail.New ( sInvoiceNumber ; sPartCode ; lQuantity ; dPrice ; sTaxCode ; dAmountTaxA ; lAccountTaxA ; dAmountTaxB ; lAccountTaxB )

- Description : Ajoute une ligne de détail à une transaction / facture client.
- Paramètre "sInvoiceNumber" : Numéro de la facture utilisée par Sage 50.
- Paramètre "sPartCode" : Code du produit ou service affiché dans la section 'Stocks & Services' de Sage 50
- Paramètre "lQuantity" : Quantité du produit ou service.
- Paramètre "dPrice" : Prix du produit ou service.
- Paramètre "sTaxCode" : Code de taxe à utiliser.
- Paramètre "dAmountTaxA" : Montant de la taxe A.
- Paramètre "lAccountTaxA" : Numéro de compte de la taxe A.
- Paramètre "AmountTaxB" : Montant de la taxe B.
- Paramètre "lAccountTaxB" : Numéro de compte de la taxe B.
- Attention : le 'Commit' n'est pas automatique. Vous devez utiliser les fonctions s50.Invoice.Commit() ou s50.invoice.Rollback().



### `s50.InvoiceDetailAccount.New ( sInvoiceNumber ; IAccount ; sDescription ; IQuantity ; dPrice ; sTaxCode ; dAmountTaxA ; IAccountTaxA ; dAmountTaxB ; IAccountTaxB )`

- Description : Utilise un compte du G/L au lieu d'un item du catalogue.
- Paramètre "sInvoiceNumber" : Numéro de la facture utilisée par Sage 50.
- Paramètre "IAccount" : Numéro du compte du G/L à utiliser.
- Paramètre "sDescription" : Description de la ligne de détails de facture.
- Paramètre "IQuantity" : Quantité du produit ou service.
- Paramètre "dPrice" : Code de taxe à utiliser.
- Paramètre "sTaxCode" : Code de taxe à utiliser.
- Paramètre "dAmountTaxA" : Montant de la taxe A.
- Paramètre "IAccountTaxA" : Numéro de compte de la taxe A.
- Paramètre "AmountTaxB" : Montant de la taxe B.
- Paramètre "IAccountTaxB" : Numéro de compte de la taxe B.
- Attention : le 'Commit' n'est pas automatique. Vous devez utiliser les fonctions `s50.Invoice.Commit()` ou `s50.invoice.Rollback()`.

### `s50.Invoice.Freight ( sInvoiceNumber ; dFreightAmount ; sFreightTaxCode ; dFreightTaxA ; IFreightAccountTaxA ; dFreightTaxB ; IFreightAccountTaxB )`

- Description : Utiliser pour inclure le montant des frais de transport dans une facture de type client.
- Paramètre "sInvoiceNumber" : Numéro de la facture client.
- Paramètre "dFreightAmount" : Montant des frais de transport.
- Paramètre "sFreightTaxCode" : Code de taxe à utiliser.
- Paramètre "dFreightTaxA" : Montant de la taxe A.
- Paramètre "IFreightAccountTaxA" : Numéro du compte de taxe A à utiliser.
- Paramètre "dFreightTaxB" : Montant de la taxe B.
- Paramètre "IFreightAccountTaxB" : Numéro du compte de taxe B à utiliser.

### `s50.Invoice.Commit`

- Description : Enregistre de façon permanente à la base de données Sage 50 les enregistrements générés par les fonctions `s50.Invoice.New()` et `s50.InvoiceDetail.New()`.



## s50.Invoice.Rollback

- Description : Annuler la transaction vers la base de données Sage 50 des enregistrements générés par les fonctions s50.Invoice.New() et s50.InvoiceDetail.New().

## s50.InvoicePayment.New ( sInvoiceNumber ; lCustomrId ; dtDate ; dAmount ; lType ; lBankAccount ; sReference ; sComment )

- Description : Ajoute un paiement à une transaction/facture client.
- Paramètre "sInvoiceNumber" : Numéro de la facture utilisée par Sage 50.
- Paramètre "lCustomrId" : Numéro unique du client dans Sage 50.
- Paramètre "dtDate" : Date du paiement.
- Paramètre "damount" : Montant du paiement.
- Paramètre "lType" : Type du paiement. (0=Comptant, 1=Chèque, >1=Carte de crédit)
- Paramètre "lBankAccount" : Numéro du compte où le paiement sera enregistré.
- Paramètre "sReference" : Numéro du chèque ou nom de la carte de crédit.
- Paramètre "sComment" : Notes.

## s50.Purchase.New ( sName ; lVendorId ; dtDate ; sReference ; sDescription ; sTaxCode ; dSubTotal ; lChargeAccount ; dTotalTaxA ; lAccountTaxA ; dTotalTaxB ; lAccountTaxB ; sNonTaxableDescription ; dNonTaxableAmount ; lNonTaxableAccount ; dTotal ; sComment)

- Description : Crée l'entête et les détails d'une transaction/achat fournisseur.
- Paramètre "sName" : Nom de la compagnie du fournisseur.
- Paramètre "lVendorId" : Numéro unique du fournisseur dans Sage 50.
- Paramètre "dtDate " : Date de la transaction.
- Paramètre "sReference" : Référence externe.
- Paramètre "sDescription" : Description de l'achat.
- Paramètre "sTaxCode" : Code de taxe à utiliser.
- Paramètre "dSubTotal" : Sous-total de la transaction.
- Paramètre "lChargeAccount" : Numéro du compte où la charge sera enregistrée.
- Paramètre "dTotalTaxA" : Montant de la taxe A.
- Paramètre "lAccountTaxA" : Numéro de compte de la taxe A.
- Paramètre "dTotalTaxB" : Montant de la taxe B.
- Paramètre "lAccountTaxB" : Numéro de compte de la taxe B.
- Paramètre "sNonTaxableDescription" : Description de la partie sans taxe.
- Paramètre "dNonTaxableAmount" : Montant de la partie sans taxe.



- Paramètre "INonTaxableAccount" : Numéro de compte où la partie sans taxe sera enregistrée.
- Paramètre "dTotal" : Total de la transaction.
- Paramètre "sComment" : Notes.

### s50 Vendor.Exist ( sName )

- Description : Renvoie le numéro unique du fournisseur dans Sage 50 s'il existe sinon renvoie "" (chaîne de caractère vide).
- Paramètre "sName" : Nom de la compagnie du fournisseur dans Sage 50.

### s50 Vendor.New ( sName ; sCntcName ; sStreet1 ; sStreet2 ; sCity ; sProvState ; sCountry ; sPostalZip ; sPhone1 ; sPhone2 ; sFax ; sEmail ; sWebSite ; nLangPref ; dtSince ; sMemo )

- Description : Ajout d'un fournisseur. Renvoie le numéro unique du fournisseur dans Sage 50.
- Paramètre "sName" : Nom de la compagnie.
- Paramètre "sCntcName" : Nom du contact principale de la compagnie.
- Paramètre "sStreet1" : Adresse de la compagnie.
- Paramètre "sStreet2" : Adresse de la compagnie.
- Paramètre "sCity" : Nom de la ville.
- Paramètre "sProvState" : Nom de la province.
- Paramètre "sCountry" : Nom du pays.
- Paramètre "sPostalZip" : Code postal.
- Paramètre "sPhone1" : Numéro de téléphone #1.
- Paramètre "sPhone2" : Numéro de téléphone #2.
- Paramètre "sFax" : Numéro de fax.
- Paramètre "sEmail" : Adresse courriel.
- Paramètre "sWebSite" : Adresse du site web.
- Paramètre "nLangPref" : Langue préféré. (0=Anglais, 1=Espagnol, 2=Français)
- Paramètre "dtSince" : Date d'ajout de ce fournisseur à Sage 50.
- Paramètre "sMemo" : Notes.

### s50 Vendor.Update ( IId ; sName ; sCntcName ; sStreet1 ; sStreet2 ; sCity ; sProvState ; sCountry ; sPostalZip ; sPhone1 ; sPhone2 ; sFax ; sEmail ; sWebSite ; sMemo )

- Description : Mise à jour des informations générale d'un fournisseur.
- Paramètre "IId" : Numéro unique du fournisseur dans Sage 50.



- Paramètre "sName" : Nom de la compagnie.
- Paramètre "sCntcName" : Nom du contact principale de la compagnie.
- Paramètre "sStreet1" : Adresse de la compagnie.
- Paramètre "sStreet2" : Adresse de la compagnie.
- Paramètre "sCity" : Nom de la ville.
- Paramètre "sProvState" : Nom de la province.
- Paramètre "sCountry" : Nom du pays.
- Paramètre "sPostalZip" : Code postal.
- Paramètre "sPhone1" : Numéro de téléphone #1.
- Paramètre "sPhone2" : Numéro de téléphone #2.
- Paramètre "sFax" : Numéro de fax.
- Paramètre "sEmail" : Adresse courriel.
- Paramètre "sWebSite" : Adresse du site web.
- Paramètre "sMemo" : Notes.

#### s50.Utills.CheckSqlSyntax ( Sql )

- Description : Vérifie la requête SQL pour des valeurs manquantes.
- Paramètre "Sql" : Requête SQL.

#### s50.Utills.Date ( fmDate )

- Description : Convertie une date FileMaker au format requis par Sage 50.
- Paramètre "fmDate" : Date.

#### s50.Utills.DateFirstOfLastMonth ( fmDate )

- Description : Renvoie la date du début du mois précédent fmDate. Ex. : 28/09/2013 renvoie 01/08/2013.
- Paramètre "fmDate" : Date.

#### s50.Utills.ESQ ( Text )

- Description : Convertit les apostrophes ' en double apostrophe '' pour la requête SQL.
- Paramètre "Text" : Texte



### s50.Utills.FormatDecimal ( Number )

- Description : Change le format du nombre avec un point (.) comme séparateur de décimal.
- Paramètre "Number" : Nombre à formater.

### s50.Utills.getNextId ( TableId )

- Description : Renvoie le prochain numéro unique à utiliser pour la table TableId.
- Paramètre "TableId" : Numéro unique de la table Sage 50.
- Table ID :

20 tAccount 24 tBRSum 25 tBRTTr 26 tBRSumJE 30 tProject  
40 tVendor 43 tVenTr 44 tVenTrDt 50 tCustomr 53 tCusTr  
54 tCusTrDt 60 tPurOrdr 62 tSalOrdr 70 tEmpUS 72 tPyDetU  
80 tEmp 82 tPyDet 90 tInvent 97 tPrclList 100 tJourEnt  
103 tJEntLY 106 tJEntCsh 110 tUser 113 tCurrncy 115 tCCard  
116 tShipper 119 tTask 121 tStmtDwn 122 tSDLine 123 tUserWnd  
126 tTaxDtl 150 tITRec 160 tRCGen 163 tRCPur 166 tRCSal  
169 tRCPrI 171 tRCPrIU 173 tRCTrf 176 tRCAdj 179 tRCPay  
191 tTaxAuth 192 tTaxCode 200 tActHRel 201 tActHdat 210 tRCBOM  
212 tRCTSlp 220 tTSRec 222 tPmtHdr 223 tRcpHdr 224 tPriHdr  
225 tActDpt 255 tActRang 256 tJobCat 261 tCnActRI 268 tDeposit  
269 tDSCheq 270 tDSCash 272 tDSDenom 273 tDSMult 304 tRptOpts  
307 tRptGrp 350 tInvLoc 352 tSDSHdr 353 tSDSCHqs 354 tSDSCash  
369 tRQRet 381 tGrpExp 386 tCustShp 388 tInvLot 390 tInvCate  
391 tUsrRole 400 tInvSNum 404 tSBom 407 tJEntPH 408 tJELYPH  
410 tCCardTr 414 tSDNode 415 tSDUUIID 438 tGovForm 440 tSDDgst  
445 tSDTax

### s50.Utills.getNextInvoiceNumber

- Description : Renvoie le prochain numéro de facture client à utiliser.

### s50.Utills.getNextReceiptNumber

- Description : Renvoie le prochain numéro de reçu à utiliser.



### s50.Utills.getInvoiceBalance ( sInvoiceNumber ; ICustomrId )

- Description : Renvoie le solde d'une facture client.
- Paramètre "sInvoiceNumber" : Numéro de la facture utilisée par Sage 50.
- Paramètre "ICustomrId" : Numéro unique du client dans Sage 50.

### s50.Utills.getPurchaseBalance ( sInvoiceNumber ; IVendorId )

- Description : Retourne la balance sur une facture d'achat fournisseur.
- Paramètre "sInvoiceNumber" : Numéro de la facture d'achat.
- Paramètre "IVendorId" : Numéro unique du fournisseur dans Sage 50.

### s50.Utills.getTaxCodeId ( sTaxCode )

- Description : Renvoie le numéro unique d'un groupe de taxe.
- Paramètre "sTaxCode" : Code de taxe utilisé par Sage 50.

### s50.Utills.getTaxInfo ( sTaxCode ; ITaxAccountNumber )

- Description : Renvoie des informations relatives au code de taxe. (réf. : code source)
- Paramètre "sTaxCode" : Code de taxe utilisé par Sage 50.
- Paramètre "ITaxAccountNumber" : Numéro de compte de la taxe de vente ou d'achat.

### s50.Utills.setNextId ( TableId )

- Description : Assigne le prochain numéro unique à utiliser pour la table TableId. (voir s50.Utills.getNextId())
- Paramètre "TableId" : Numéro unique de la table Sage 50.

### s50.Utills.setNextInvoiceNumber ( Number )

- Description : Assigne le prochain numéro de facture client à utiliser.
- Paramètre "Number" : Numéro

### s50.Utills.setNextReceiptNumber ( Number )

- Description : Assigne le prochain numéro de reçu à utiliser.
- Paramètre "Number" : Numéro



## RÉFÉRENCES

### Notes techniques

- L'extension sODBC sous Windows requiert la version XP ou plus récent.
- L'extension sODBC sous Mac OS X requiert la version 10.8 (Mountain Lion) ou plus récent.
- L'extension sODBC sous Windows requiert le standard Microsoft .NET 3.5 qui est installé avec Windows Vista, 7 et 8. Dans le cas de Windows XP, il faut installer les ressources .NET 3.5 disponibles chez Microsoft. Consulter la page Web <http://www.microsoft.com/net/>
- Si votre OS est Windows 7 ou 8 en mode 64 bits (ou XP 64 bits), utilisez le ODBC Manager suivant pour les pilotes 32 bits ODBC : C:\Windows\SysWOW64\odbcad32.exe
- L'extension sODBC est compatible avec les versions 9, 10, 11, 12 et 13 de FileMaker Pro, FileMaker Pro Advanced et FileMaker Server.
- L'extension sODBC peut-être utilisée avec le pilote ODBC d'Acomba avec les versions 9, 10, 11 et 13 de FileMaker Pro et FileMaker Pro Advanced. Cependant pas avec la version 12 de FileMaker Pro et FileMaker Pro Advanced.
- L'extension sODBC peut-être utilisée avec le logiciel Sage 50 avec les versions 9, 10, 11, 12 et 13 de FileMaker Pro et FileMaker Pro Advanced.
- L'extension sODBC est compatible avec la technologie Server-side scripting (SSS) de FileMaker.
- Les codes d'items doivent concorder entre Sage 50 et votre solution FileMaker.
- Le code de taxe pour 'Aucune Taxe' dans Sage 50 doit être vide.
- Il doit y avoir au moins un client et un fournisseur créés dans Sage 50 pour permettre l'exportation avec FileMaker.
- Il n'y a pas de gestion de rabais (remise) client pour Sage 50. Nous vous recommandons de configurer ces valeurs par défaut à zéro (0) dans le menu paramètres/client.
- Le 'Commit' n'est pas automatique pour les fonctions s50.Invoice.New() et s50.InvoiceDetail.New(). N'oubliez pas d'utiliser les fonctions s50.Invoice.Commit() ou s50.invoice.Rollback().
- Vous référer au fichier de démonstration livré avec les fonctions personnalisées pour plus de détails sur le paramétrage du logiciel Sage 50.
- Un problème pour le numéro du reçu est relié au comportement du logiciel Sage 50. Exemple : si on ouvre le module de facturation, le prochain numéro de facture est « loadé » en mémoire et est mise à jour quand on ferme Sage 50. Donc si on fait du travail normal en plus d'exporter des factures dans Sage 50, le numéro qui avait été mis en mémoire n'est plus le bon. Il y a une mécanique en place pour tenter de gérer ce comportement au niveau du numéro de facture. Nous vous recommandons de ne pas utiliser la même session Sage 50 pour travailler et pour exporter. Le mieux serait de fermer Sage 50 et de le réouvrir avant un export et de ne pas aller ce promener dans les différentes section du logiciel. Donc on ouvre Sage 50, on ne touche à rien et on exporte. Si vous aviez un paiement en double, vous pourriez simplement le supprimer et le recréer « à la mitaine ».



## Dépannage (« Troubleshooting »)

- Nous invitons les utilisateurs à communiquer avec nous pour nous aider à bâtir cette section du manuel.

## À propos de FileMaker, Inc.

- FileMaker développe des programmes de base de données primés. Ses produits comprennent la légendaire ligne de produits FileMaker Pro pour Windows, Mac et Internet. FileMaker a gagné plus de 52 prix, plus que ses huit concurrents combinés de 2003 à 2009 aux États-Unis, et un total de 134 prix dans le monde durant ces mêmes années. Des millions d'utilisateurs – des individus aux grandes entreprises – font confiance aux applications FileMaker pour gérer, analyser et échanger de l'information. FileMaker, Inc. est une filiale d'Apple Inc.

