

LES ACCES ODBC AVEC LE SYSTEME SAS®

Par SAS  - fafabzh6

Date de publication : 7 avril 2009

Le but de ce tutoriel est de vous présenter les accès via ODBC sur SAS.

Introduction.....	3
I - SAS/ACCESS to ODBC.....	3
II - Driver ODBC SAS.....	5
III - Driver ODBC SAS Universel.....	6
IV - Version 8.....	6
Références et liens utiles.....	7

Introduction

ODBC, qui signifie Open DataBase Connectivity, est une interface standard fournissant une API commune pour accéder à des bases de données.

Beaucoup de logiciels adhèrent à ce standard, permettant à ses utilisateurs d'accéder aux données créées par d'autres logiciels.

Trois composants sont nécessaires à la mise en oeuvre d'ODBC :

- l'application cliente,
- le gestionnaire de pilotes/drivers ODBC (ODBC driver manager),
- le pilote/driver ODBC.

Ce gestionnaire de drivers ODBC, développé par Microsoft, permet de gérer l'interaction entre l'application cliente et le(s) driver(s) ODBC.

Le Système SAS et ODBC

Deux scénarii vont impliquer le Système SAS et ODBC.

Dans le premier cas, le Système SAS est l'application cliente. L'utilisateur SAS veut accéder à des données non-SAS. Il doit avoir à disposition le driver ODBC du produit qu'il veut accéder. Au niveau du Système SAS, il doit avoir le module SAS/ACCESS® to ODBC.

Exemple : un utilisateur SAS veut pouvoir accéder, depuis le Système SAS, à des données MSACCESS. Il lui faut le "Microsoft Access Driver" et le module SAS/ACCESS to ODBC.

Dans le deuxième cas, le Système SAS est le serveur de données. L'utilisateur non-SAS veut accéder à des données SAS. Il doit avoir à disposition le driver ODBC SAS (fourni par SAS) et bien-sûr une application cliente adhérant au standard ODBC. Au niveau du Système SAS, plusieurs paramètres sont à prendre en compte selon que l'on travaille en local ou en client/serveur. Ces hypothèses seront explicitées plus loin.

Exemple : un utilisateur MS-EXCEL veut pouvoir accéder, depuis MS-QUERY, à des données SAS. Il lui faut le driver ODBC SAS et bien-sûr un Système SAS quelque part sur la même machine ou ailleurs sur un réseau.

Trois types de sources de données existent dans ODBC :

DSN Utilisateur (User DSN) : ces sources de données sont spécifiques à un ordinateur (locales) et ne peuvent être utilisées que par l'utilisateur en cours.

DSN Système (System DSN) : ces sources de données sont spécifiques à un ordinateur (locales), et non associées (dédiées) à un utilisateur. Le système, ou tout utilisateur disposant d'autorisations d'accès, peut utiliser une source de données définie avec un DSN système.

DSN Fichier (File DSN) : il s'agit de sources de données basées sur des fichiers qui peuvent être partagées par tous les utilisateurs disposant des mêmes pilotes installés et qui ont donc accès à la base de données. Il n'est pas nécessaire que ces sources de données soient associées (dédiées) à un utilisateur ou spécifiques à un ordinateur (locales).

I - SAS/ACCESS to ODBC

SAS/ACCESS to ODBC est un module du Système SAS permettant d'accéder à diverses sources de données ODBC.

En version 6.12, ce module comprend, contrairement à beaucoup de modules SAS/ACCESS, une seule fonctionnalité : le pass-through. Cela consiste à envoyer des ordres SQL directement à la source de données. On utilise pour cela la procédure SQL.

Dans cette situation, il faut savoir qu'une source de données ODBC doit être préalablement configurée, à l'aide d'un driver ODBC tiers. Cette source doit pouvoir être validée par une application autre que le Système SAS, c'est-à-dire testée par un autre logiciel comme par exemple MS -QUERY. Cette simple manipulation évite bien des soucis, et notamment recadre le problème si jamais cela ne fonctionne pas.

Exemple : un utilisateur SAS veut pouvoir accéder, depuis le Système SAS, à des données MSACCESS. Il a configuré une source de données ODBC à l'aide du "Microsoft Access Driver". Il veut désormais attaquer cette source de données depuis le Système SAS. Avant cela, il faut valider l'accès à cette source de données depuis MS-QUERY par exemple. Cela fonctionne, on peut alors passer au Système SAS. Cela ne fonctionne pas, l'utilisateur doit voir avec son support informatique pour détecter l'origine du problème.

Attention : la configuration d'une source de données ODBC tierce est à la charge de l'utilisateur.

Une fois la source testée et validée dans un monde non-SAS, on peut tenter d'y accéder depuis le Système SAS.

Comment faire ?

La programmation des ordres SQL n'étant pas d'une simplicité extrême, on bénéficie au sein du Système SAS d'un outil qui va faciliter les choses. Il s'agit du QUERY. Il va permettre de construire en interactif une requête SQL (donc une procédure SQL) vers la source de données ODBC, dont le code servira de modèle pour une utilisation orientée "programmation" (batch).

Pour créer un modèle de programme, il suffit d'exécuter les étapes suivantes (en version 6.12, pour la version 8, voir plus loin) :

- lancer le Système SAS
- taper "query" dans la barre de commandes
- menu "Actions" -> "Switch Access Mode" -> "ODBC"
- une fenêtre "Sélectionner la source de données" apparaît, il faut donc choisir la source de données ODBC que l'on veut accéder qui est soit une source de données fichier (DSN Fichier) ou machine (DSN Système ou DSN Utilisateur)
- une fois la source choisie, dans la fenêtre "SQL Query Tables", la List Box "Available Tables" doit contenir des éléments
- sélectionner une table parmi les tables disponibles, et la faire basculer vers la List Box "Selected Tables"
- cliquer sur "OK"
- la fenêtre "SQL Query Columns" apparaît avec la liste des variables disponibles dans la List Box "Available Columns" ; les 2 premières lignes correspondent à un "select count(*)" et à un "select *"
- sélectionner une colonne parmi les colonnes disponibles, et la faire basculer vers la List Box "Selected Columns"
- menu "Actions" -> "Show Query..."
- la fenêtre "SQL Query" apparaît avec la requête générée
- bouton "Save Query" -> "Save as External File with Create Table."
- une fenêtre apparaît demandant le nom de la bibliothèque de stockage et le nom de la table à créer ; mettre par exemple WORK pour la bibliothèque et TEST pour la table
- cliquer sur "OK"
- la fenêtre "Save As – SQL QUERY" apparaît ; donner un nom de fichier et éventuellement un chemin pour l'enregistrement
- cliquer sur "Enregistrer"
- bouton "Goback"
- menu "File" -> "End..."
- cliquer sur "OK" pour confirmer
- dans le "Program Editor", ouvrir le fichier précédemment créé

- rajouter l'instruction "quit ;" à la fin et exécuter

Voilà, un modèle de procédure SQL est désormais disponible et réutilisable.

II - Driver ODBC SAS

Le driver ODBC SAS est une implémentation du standard ODBC permettant d'accéder, de manipuler ou de mettre à jour des sources de données SAS depuis des applications clientes adhérant au standard ODBC. Ce driver est fourni gratuitement avec le Système SAS et est disponible uniquement sur Windows.

Le but de ce document n'est pas de détailler toutes les étapes à suivre, d'autres documents le faisant parfaitement (Technical Report, références à la fin de ce document), mais est d'expliquer brièvement le fonctionnement de ce driver et les différentes situations que l'on peut rencontrer.

Cas 1 : SAS et l'application cliente sur le même poste client

Produits nécessaires sur le **client** : **SAS (BASE, +) + driver ODBC SAS**

Protocole utilisé : **DDE** (local)

Commentaire : c'est le cas le plus simple et le plus souvent rencontré. L'utilisateur travaille à la fois sur SAS et sur d'autres applications depuis lesquelles il veut accéder à ses données SAS.

Quelques mots sur les paramètres :

onglet Servers

- "Server Name" : nom quelconque
- "Password" : pas de mot de passe
- "Access Method" : DDE
- bouton "Configure..." : indiquer où se trouve SAS en local

onglet Libraries

- "Library Name" : nom de la bibliothèque
- "Host File Name" : répertoire de la bibliothèque

Au niveau du fonctionnement, une demande ODBC effectuée par l'application cliente lance via DDE une session SAS sur le poste. Cette session est utilisée ensuite pour toutes les opérations. A la fin de l'utilisation d'ODBC, cette session SAS n'est pas arrêtée, il faut le faire manuellement.

Cas 2 : SAS sur un serveur et l'application cliente sur un poste client

Produits nécessaires sur le poste **client** : **driver ODBC SAS**

Produits nécessaires sur le **serveur** : **SAS (SAS/BASE®, SAS/SHARE®, SAS/SHARE*Net™, +)**

Protocole utilisé : **TCP/IP**

Commentaire : c'est un cas plus rare et plus complexe à mettre en place, beaucoup d'utilisateurs ne connaissant pas SAS/SHARE.

Préalablement à la création des sources de données, il faut d'abord mettre à disposition les données SAS au sein d'un serveur SAS/SHARE par le biais d'une proc server. Cette procédure écoute les connexions sur un certain service (correspondant à un certain port). Cette combinaison service/port doit être référencée dans le fichier "services" du poste client.

Ensuite, le paramétrage de la source se fait de la façon suivante :

onglet Servers

- "Server Name" : nom_du_serveur.nom_du_service

- "Password" : pas de mot de passe
- "Access Method" : TCP
- bouton "Configure..." : indiquer le nom du serveur s'il n'est pas déjà mentionné par défaut, ainsi qu'un userid/password si le serveur SAS/SHARE est démarré en mode sécurisé

onglet Libraries

- il est préférable de déclarer ses bibliothèques au niveau du serveur SAS/SHARE plutôt qu'au niveau du driver

Au niveau du fonctionnement, une session SAS (serveur SAS/SHARE) est préalablement lancée et écoute en permanence sur un port donné. Chaque requête ODBC ouvre une connexion via TCP/IP vers cette session qui va effectuer toutes les opérations.

Exemple de démarrage d'un serveur SAS/SHARE :

```
libname data1 'chemin1' ;  
libname data2 'chemin2' ;  
proc server serverid=nom_du_service ;  
quit;
```

Ainsi les données de data1 et data 2 sont mises à disposition au sein d'un serveur SAS/SHARE.

III - Driver ODBC SAS Universel

Le driver ODBC SAS Universel est une autre forme de mise à disposition de données SAS par ODBC. Ici, il s'agit de lire des données SAS sans avoir SAS là où se trouvent les données. Ce driver donne accès en lecture uniquement à des fichiers SAS en local ou sur un serveur distant par les protocoles FTP ou HTTP.

Exemple : un utilisateur reçoit des fichiers SAS. Il n'a pas SAS sur son poste ni ailleurs. Il peut quand même utiliser ces données dans une application adhérant au standard ODBC par le biais de ce driver.

Tout comme le driver ODBC SAS classique, il est uniquement disponible sur Windows. Par contre, il est payant.

IV - Version 8

La version 8 apporte de nouvelles fonctionnalités, que ce soit au niveau de SAS/ACCESS to ODBC ou au niveau du driver ODBC SAS.

SAS/ACCESS to ODBC :

L'arrivée des nouveaux moteurs SAS/ACCESS impacte aussi SAS/ACCESS to ODBC. En effet, ce dernier qui se distinguait par sa pseudo-complexité (procédure SQL uniquement), est désormais aussi simple à utiliser que les autres.

Pour tester une source de données ODBC dans SAS (après avoir validé qu'elle fonctionnait dans un monde non-SAS), il suffira désormais d'exécuter le code suivant au sein de SAS :

```
libname test odbc prompt=yes ;
```

Pour les amateurs du query et du pass-through, vous pouvez comme en version 6.12 générer un modèle de pass-through en passant par ce query :

- lancer le Système SAS
- taper "query" dans la barre de commandes
- menu "Tools" -> "Switch Access Mode" -> "ODBC"

- une fenêtre "Sélectionner la source de données " apparaît, il faut donc choisir la source de données ODBC que l'on veut accéder qui est soit une source de données fichier (DSN Fichier) ou machine (DSN Système ou DSN Utilisateur) une fois la source choisie, dans la fenêtre "SQL Query Tables", la List Box "Available Tables" doit contenir des éléments
- sélectionner une table parmi les tables disponibles, et la faire basculer vers la List Box "Selected Tables"
- cliquer sur "OK"
- la fenêtre "SQL Query Columns" apparaît avec la liste des variables disponibles dans la List Box "Available Columns" ; les 2 premières lignes correspondent à un "select count(*)" et à un "select **"
- sélectionner une colonne parmi les colonnes disponibles, et la faire basculer vers la List Box "Selected Columns"
- menu "Tools" -> "Show Query..."
- la fenêtre "SQL Query" apparaît avec la requête générée
- bouton "Save Query" -> "Save as External File with Create Table."
- une fenêtre apparaît demandant le nom de la bibliothèque de stockage et le nom de la table à créer ; mettre par exemple WORK pour la bibliothèque et TEST pour la table
- à cliquer sur "OK"
- la fenêtre "Save As" apparaît ; donner un nom de fichier et éventuellement un chemin pour l'enregistrement
- cliquer sur "Enregistrer"
- bouton "Goback"
- menu "File" -> "Close"
- cliquer sur "OK" pour confirmer
- dans l'"Editor", ouvrir le fichier précédemment créé
- rajouter l'instruction "quit ;" à la fin et exécuter

On retrouve alors un modèle réutilisable de pass-through pour ODBC.

Driver ODBC :

En ce qui concerne le driver ODBC, la version 8 (ou tout du moins le driver fourni avec la version 8) apporte un changement au niveau du protocole utilisé en local. En effet, le protocole DDE n'est plus disponible pour les connexions locales. Le protocole TCP est utilisé désormais à la fois pour les connexions locales et distantes.

Un paramétrage différent est donc nécessaire pour les sources ODBC locales. Dans l'onglet "Servers", le "Server Name" ne sera plus un nom quelconque mais le nom d'un service (combinaison service/port déclarée dans le fichier "services" du poste client).

Références et liens utiles

SAS ODBC Driver Technical Report: User's Guide and Programmer's Reference, Release 6.11 (#55272)
SAS ODBC Driver User's Guide and Programmer's Reference, Version 8 (#57257)

TS-593 Configuring the SAS V6.12 ODBC Driver

TS-626 Configuring the SAS V8 ODBC Driver

 **le site SAS, documentation technique**

 **Le domaine SAS Developpez**