

QU'APPORTE LA VERSION 9.4 DE SAS POUR ECRIRE/LIRE DES FICHIERS EXCEL ET CRÉER DES RAPPORTS EXCEL ?

Au Support Clients, nous recevons très régulièrement des questions sur les interactions entre SAS® et Microsoft Excel. Elles sont globalement de deux types : la lecture et l'écriture de fichiers Excel depuis SAS, ou la production de rapports au format Excel. Nous allons ici faire un état des lieux sur ces deux sujets en version 9.4 de SAS.

Caractéristiques :

Catégories : SAS Access® to PC Files, Base SAS®
OS : Windows
Version : SAS® 9.4
Vérifié en mai 2016

Sommaire

1.	Importer et exporter au format Excel, avec le module dédié	1
1.1.	Pré-requis dans SAS.....	1
1.2.	Vérification de l'Adressage (32 ou 64 bit) de SAS et d'Excel	2
1.3.	Et en pratique ?.....	3
1.3.1.	Moteur xlsx	4
1.3.2.	Les assistants d'importation / exportation	5
1.4.	Les messages d'erreur les plus fréquents, en version 9.4	6
1.5.	Quelques nouveautés en version 9.4	7
2.	Importer et exporter, les autres solutions	8
2.1.	SAS Enterprise Guide	8
2.2.	Le format CSV	9
2.3.	Les liens DDE	9
2.4.	SAS Add-In for Microsoft Office	10
3.	Exporter, en passant par des rapports	10
3.1.	ODS HTML	10
3.2.	ODS TAGSETS.EXCELXP	11
3.3.	ODS EXCEL	11
4.	En cas de problème.....	13
4.1.	Eléments à transmettre au Support Clients	13
5.	Liens utiles.....	13
6.	Conclusion	13

1. IMPORTER ET EXPORTER AU FORMAT EXCEL, AVEC LE MODULE DÉDIÉ

Le module dédié s'appelle **SAS/Access® Interface to PC Files**. Il permet de créer et de lire des fichiers Excel natifs.

Un article, très complet, sur ce sujet a déjà été proposé dans une [précédente édition de notre newsletter](#). Le but ici est de s'attacher à décrire les situations qui sont le plus fréquemment rencontrées en environnement Windows avec SAS et Excel sur le même PC, et à illustrer comment réaliser les importations et exportations de fichiers Excel le plus simplement et rapidement possible.

1.1. Pré-requis dans SAS

Tout d'abord, il convient de vérifier si vous disposez de la licence pour le module SAS/Access Interface to PC Files. Pour le vérifier, il suffit d'exécuter :

```
proc setinit;run;
```

dans SAS et de vérifier le contenu du journal. A minima, le journal doit présenter ce type d'information dans la section dates d'expiration :

Dates d'expiration du produit :

```
---Logiciel Base SAS
                                01MAY2015
--- SAS/ACCESS Interface to PC Files
                                01MAY2015
```

Un module présent dans la licence est automatiquement installé dans SAS. Mais il est néanmoins bon de le vérifier en exécutant par exemple :

```
proc product_status;run;
```

```
For Logiciel Base SAS ...
  Custom version information: 9.4_M2
  Image version information: 9.04.01M2P072314
For SAS/ACCESS Interface to PC Files ...
  Custom version information: 9.4_M2
```

Ces informations étant validées, l'étape suivante est de vérifier l'adressage (32 ou 64 bit de SAS et d'Excel).

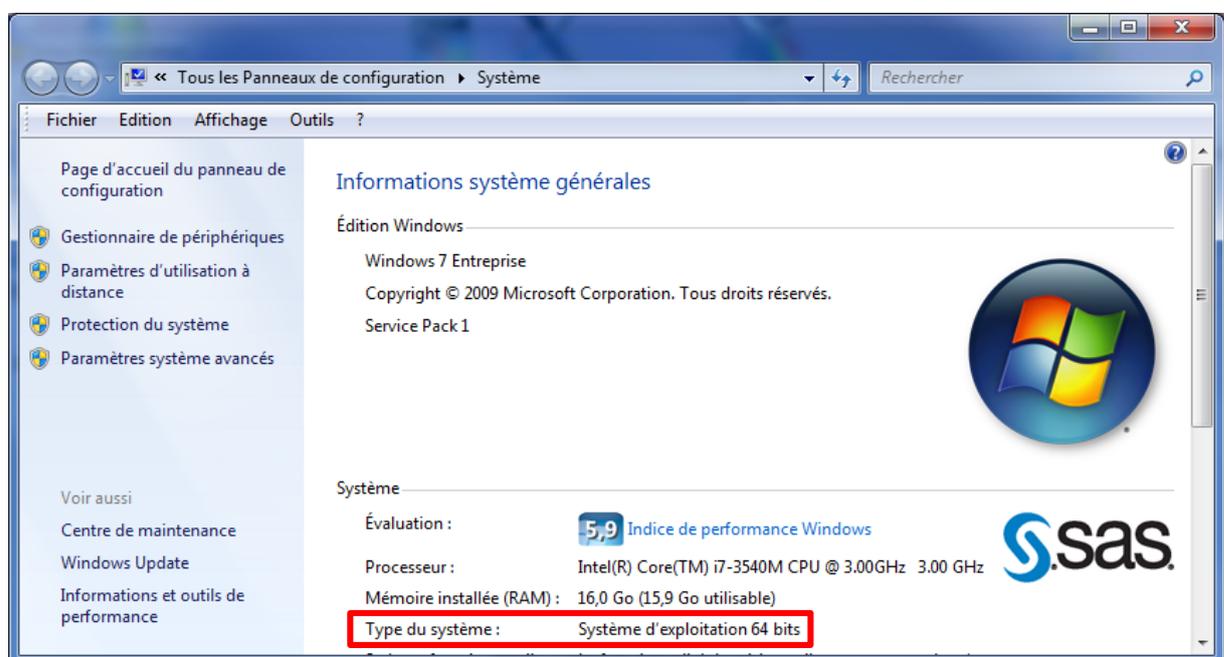
1.2. Vérification de l'Adressage (32 ou 64 bit) de SAS et d'Excel

Sur un PC en 32-bit, SAS et Excel seront forcément disponibles en 32-bit, tandis que sur un PC en 64-bit, toutes les combinaisons sont possibles, à savoir :

- SAS en 64-bit et MS Office en 32-bit
- SAS en 64-bit et MS Office en 64-bit
- SAS en 32-bit et MS Office en 32-bit
- SAS en 32-bit et MS Office en 64-bit

Comment vérifier ces informations ?

- Vérification de l'adressage du PC :
 - dans le panneau de configuration, sous le groupe Système :



- Ou encore depuis SAS, en consultant une macro variable dans laquelle nous récupérerons cette information. L'exécution de :

```
%put Version de Windows : &SYSHOSTINFOLONG ;
```

renvoie :

```
2 %put Version de Windows : &SYSHOSTINFOLONG ;  
Version de Windows : X64_7PRO WIN 6.1.7601 Service Pack 1 Workstation
```

L'information importante ici est X64 qui indique qu'il s'agit d'une machine 64-bit.

➤ Vérification de l'adressage de SAS 9.4 :

Une macro variable contient l'information. Il s'agit de `sysaddrbits`.
Il suffit donc de la consulter, en exécutant :

```
%put Adressage de SAS : &SYSADDRBITS bit;
```

et de vérifier l'affichage dans le journal :

```
3 %put Adressage de SAS : &SYSADDRBITS bit;  
Adressage de SAS : 64 bit
```

➤ Vérification de l'adressage d'Excel :

Les versions 2003 et 2007 sont disponibles uniquement en 32-bit. A partir de la version 2010, une version 64-bit est également disponible. L'information est présentée dans le menu :

- Fichier > Aide, dans Excel 2010
- Fichier > Compte > A propos d'Excel, dans les versions d'Excel supérieures.



1.3. Et en pratique ?

Le module SAS/Access Interface to PC File propose trois méthodes distinctes pour lire des fichiers Excel d'une part et pour en créer d'autre part :

- L'assistant d'importation et l'assistant d'exportation, accessibles depuis les menus *Fichier > Importer données...* et *Fichier > Exporter données...*
- Les procédures Import et Export
- L'instruction libname, avec des moteurs spécifiques

Notre première recommandation est de vous assurer que vous disposez au moins de la maintenance 2 de SAS 9.4 (sachant qu'aujourd'hui, une maintenance 3 est disponible). Nous avons vu comment afficher cette information au point 1.1 (où la version était 9.4_M2). Si vous disposez de la version 9.4_M0 ou 9.4_M1, une mise à jour est nécessaire (la demande est à faire par votre société, auprès de votre gestionnaire contrats SAS). Vous pourrez ainsi utiliser le moteur XLSX dans une instruction libname (nouveau de la maintenance 2) et bénéficier des nouveautés dans les procédures import et export.

Nous préconisons de préférer XLSX à XLS. En effet, le moteur XLSX supporte les derniers formats d'Excel (Excel 2007 et supérieur, ainsi que .xlsx, .xlsb et .xlsm). Par ailleurs, au niveau de la technologie utilisée, SAS accède directement au fichier Excel, sans aucune étape intermédiaire (ce qui est par contre le cas de nos autres moteurs - EXCEL et PCFILES - qui s'appuient sur des API Microsoft).

Un autre élément peut être déterminant pour le choix entre l'une ou l'autre de ces méthodes, si vous avez une configuration où l'adressage de SAS et d'Excel sont différents. En effet, si l'adressage (i.e. 32 ou 64 bits) est identique, les trois méthodes peuvent être utilisées directement, sans configuration supplémentaire. Par contre, si l'adressage est différent, la nouveauté (avec SAS 9.4 Maintenance 2) est que les méthodes utilisant le moteur XLSX fonctionneront également directement,

que ce soit avec les procédures import et export, ou dans une instruction libname. Avec d'autres moteurs (xls par exemple), que ce soit dans les versions précédentes ou encore dans cette version, il est nécessaire d'installer un composant qui fera le pont entre SAS et Excel (64/32 bit). Il s'agit du composant PC File Server.

En résumé, avec SAS 9.4 Maintenance 2, vous pourrez utiliser nativement et directement :

- Proc import et proc export, avec dbms=xlsx.
- libname, avec le moteur xlsx.
- Les assistants d'importation et d'exportation dans tous les cas de figure à cette exception près : si SAS et Excel sont dans des adressages différents, ces assistants ne fonctionneront qu'après l'installation du composant « PC File Server ».

Nous allons maintenant voir la syntaxe de façon plus détaillée pour ces deux méthodes (procédures import/export et libname) et ensuite, la manière d'utiliser les assistants d'importation/exportation.

1.3.1. Moteur xlsx

Dans les procédures Import et Export, ce moteur est spécifié au niveau de l'option DBMS=XLSX. Voici un exemple :

```
proc export data=sashelp.class outfile='c:\temp\class.xlsx' dbms=xlsx;
run;

proc import datafile='c:\temp\class.xlsx' out=class dbms=xlsx;
run;
```

Voyons maintenant la syntaxe dans une instruction libname et une illustration de quelques actions possibles par la suite avec la bibliothèque ainsi définie :

```
libname classexc XLSX 'c:\temp\class.xlsx';

/* Exemples d'utilisation de la bibliothèque nommée classexc */
data class_M;
  set classexc.class;
  where sex='M';
run;

proc means data=classexc.class;
var age;
run;

data classexc.test;
  do i=1 to 10000;
    v=round(ranuni(120)*100);
    output;
  end;
run;
```

Ici, une bibliothèque classexc a été déclarée. Elle pointe sur le fichier Excel class.xlsx. Via une étape data ensuite, la feuille (nommée class), peut être lue. L'avantage de cette approche est que du code SAS peut immédiatement être appliqué, que ce soit pour :

- ne rapatrier dans SAS qu'une sélection du fichier Excel (cf la clause where) via une étape data, comme présenté ici,
- exécuter directement une procédure, sans créer de table SAS intermédiaire,
- créer une nouvelle feuille dans le classeur Excel, de nouveau via une étape data.

Les avantages :

- Dans la procédure Import, l'option getnames= permet de lire des fichiers Excel qui n'ont pas de noms de variables en première ligne en la valorisant à NO (=YES par défaut).
- Les données de types différents dans une même colonne Excel (données à la fois numériques et alphanumériques) sont automatiquement gérées. Pour les utilisateurs qui avaient déjà été confrontés à ce type de souci par le passé et qui l'avaient résolu avec l'option mixed dans la

proc import ou la modification de la valeur de `typeguessrows`, il faut savoir qu'ici aucune option du type `MIXED=` ni aucun changement dans la base de registre ne sont nécessaires.

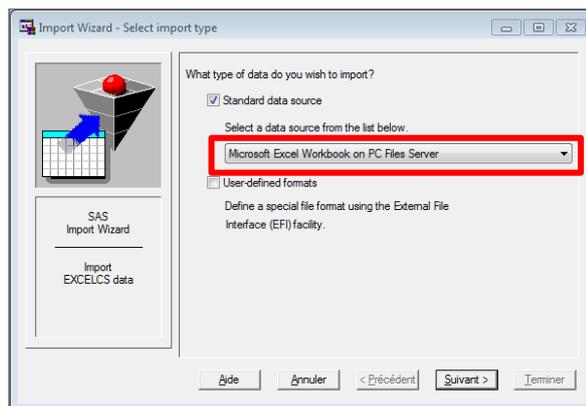
Les inconvénients :

- Quand on utilise une instruction `libname`, on a l'habitude d'aller en consulter son contenu et de visualiser ses données en passant par l'explorateur SAS. Or ouvrir (avec le Viewtable) des feuilles d'un classeur Excel via une bibliothèque de moteur XLSX ne fonctionnera pas. La raison est que le moteur lit les données de façon séquentielle, ligne par ligne.
- Il n'existe pas d'équivalent au niveau des assistants d'importation/exportation.
- Les fichiers `xlsb` ne sont pas supportés.
- Une bibliothèque de moteur `xlsx` ne peut pas être utilisée dans SAS Management Console : on ne peut pas enregistrer de tables dans les métadonnées.

1.3.2. Les assistants d'importation / exportation

Dans le cas où l'adressage de SAS et d'Excel ne correspondent pas, il est nécessaire d'installer un composant pour faire la communication entre SAS et Excel : SAS PC File Server. Ce composant est disponible sur notre site Internet, dans la rubrique « [Download](#) », ou encore dans votre dépôt SAS 9.4, en choisissant l'option « Installer SAS Foundation et logiciels associés » ou « Installer un autre logiciel ». Il est proposé de l'installer en 32 ou 64 bit. Il faut ici respecter l'adressage d'Excel.

- Dans l'assistant d'importation, il faut choisir « Microsoft Excel Workbook on PC Files Server »



Sur l'écran suivant, seul le champ `Workbook` est à renseigner dans cette configuration. Il suffit d'utiliser le bouton « Browse » pour aller sélectionner son fichier Excel. Tous les autres champs dans la section `PC Files Server` sont à laisser vides. Ces champs sont nécessaires dans une configuration où SAS et le `PC File Server` seraient sur deux machines différentes.



Sur le dernier écran, il est proposé d'enregistrer dans un programme le code généré par l'assistant. Dans cet exemple, voici la syntaxe qui aura été utilisée :

```
PROC IMPORT OUT= WORK.donnees
```

```
DATAFILE= "c:\donnees.xlsx"  
DBMS=EXCELCS REPLACE;  
    RANGE="Feuill1$";  
    SCANTEXT=YES;  
    USEDATE=YES;  
    SCANTIME=YES;
```

RUN;

On remarque le moteur EXCELCS, qui correspond bien à l'utilisation du PC File Server.

- Dans une instruction libname lorsque SAS et Excel sont installés dans un adressage différent, le moteur à utiliser est PCFILES.

```
LIBNAME monfic PCFILES PATH='c:\donnees.xlsx';  
proc print data=monfic.'Feuill1$'n;run;
```

1.4. Les messages d'erreur les plus fréquents en version 9.4

➤ 1^{er} cas

Le contexte : vous migrez d'un environnement où SAS et Excel étaient installés dans le même adressage (c'est-à-dire étaient tous les deux en 32-bit, ou tous les deux en 64-bit) vers un environnement où SAS et Excel sont dans des adressages différents (en général, vous disposez maintenant de SAS en 64-bit).

Vous utilisiez auparavant la procédure export avec l'option DBMS=XLS ou DBMS=EXCEL et/ou la procédure import avec ces mêmes options, pour créer et/ou lire des fichiers d'extension.xlsx. Elle(s) ne fonctionne(nt) plus.

L'erreur :

```
ERROR: Connect: Classe non enregistrée  
ERROR: Error in the LIBNAME statement.  
Connection Failed. [Ou Echec de la connexion.]
```

Le code :

```
PROC EXPORT DATA=libref.base  
    OUTFILE= "C:\mon_fichier.xls"  
    DBMS=EXCEL REPLACE;  
    SHEET="Feuill1";  
RUN;
```

La solution :

Utiliser DBMS=XLSX
(ceci est valable également pour la proc import)

➤ 2^{ème} cas

Le contexte : une procédure export utilisée pour créer un fichier Excel.

L'erreur :

```
Error creating XLSX file -> c:\temp\MesDonnees.xlsx . It is either not an Excel  
spreadsheet or it is damaged. Error code=80001019  
Requested Output File is Invalid  
ERROR: Export unsuccessful. See SAS Log for details.
```

Ou bien

```
ERROR: Could name the temporary file (c:\temp\MesDonnees.$$1) to the Excel name  
(c:\temp\MesDonnees.xlsx)
```

Le code :

```
proc export data=sashelp.prdsale outfile='c:\temp\MesDonnees.xlsx'  
dbms=xlsx ;  
sheet='Prdsale';  
run;
```

L'explication et la solution :

Le fichier Excel était déjà ouvert. Et il suffit de le fermer.

1.5. Quelques nouveautés en version 9.4

La principale nouveauté est l'ajout du moteur XLSX au niveau de l'instruction libname. Ce moteur est à privilégier à compter de la maintenance 2 dans vos importations/exportations de fichiers Excel, que ce soit pour utiliser les procédures import/export ou pour affecter une bibliothèque.

La procédure export, avec le moteur xlsx (option dbms=xlsx), permet désormais d'ajouter de nouvelles feuilles dans un fichier Excel. Donc si vous référencez un fichier Excel existant dans l'option outfile, une nouvelle feuille y sera insérée (option sheet=) si aucune feuille ne porte déjà ce nom. Si une feuille existe déjà, l'option échouera, à moins de n'ajouter l'option replace. L'option replace a par ailleurs pour effet de créer un backup du fichier Excel (extension .bak).

Un petit détail : l'ajout de feuilles était possible en version 9.3, à la condition d'installer des hotfixes.

Exemple de code qui crée d'abord un fichier Excel, avec une feuille (Prdsale)

```
proc export data=sashelp.prdsale  
outfile='c:\temp\MesDonnees.xlsx'  
dbms=xlsx;  
sheet='Prdsale';  
run;
```

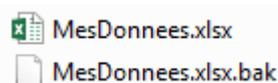
Puis ce fichier est mis à jour avec une seconde feuille (Air), grâce à l'utilisation de l'option replace

```
proc export data=sashelp.air  
outfile='c:\temp\MesDonnees.xlsx'  
dbms=xlsx  
replace;  
sheet='Air';  
run;
```

Vérification du résultat dans Excel

	A	B	C	D	E
1	ACTUAL	PREDICT	COUNTRY	REGION	DIVISION
2	925,00	850,00	CANADA	EAST	EDUCATION
3	999,00	297,00	CANADA	EAST	EDUCATION
4	608,00	846,00	CANADA	EAST	EDUCATION
5	642,00	533,00	CANADA	EAST	EDUCATION
6	656,00	646,00	CANADA	EAST	EDUCATION
7	948,00	486,00	CANADA	EAST	EDUCATION
8	612,00	717,00	CANADA	EAST	EDUCATION
9	114,00	564,00	CANADA	EAST	EDUCATION
10	685,00	230,00	CANADA	EAST	EDUCATION

Dans l'explorateur Windows, 2 fichiers existent



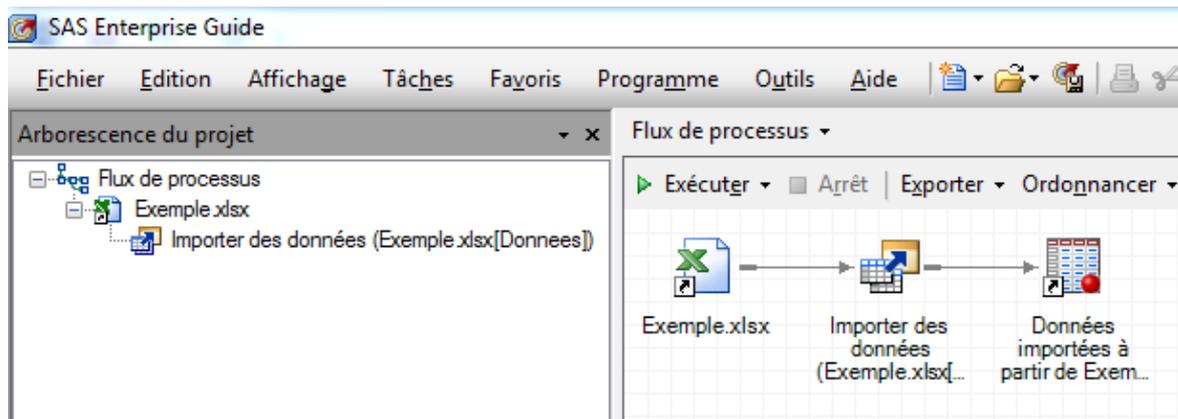
2. IMPORTER ET EXPORTER, LES AUTRES SOLUTIONS

Pour les utilisateurs qui ne disposeraient pas du module SAS/Access to PC File Format, nous allons maintenant voir quelles sont les autres approches possibles pour exporter des données SAS dans un format exploitable par Excel et lire des fichiers Excel.

2.1. SAS® Enterprise Guide®

SAS Enterprise Guide présente l'avantage de proposer des tâches et des fonctionnalités pour lire et créer des fichiers natifs Excel, sans nécessiter la présence du module SAS/Access to PC Files dans la licence.

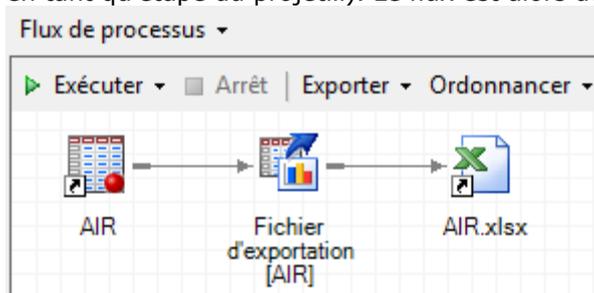
Pour l'importation, le menu se trouve sous *Fichier > Importer des données*. L'assistant permet de spécifier le fichier à lire, la feuille, éventuellement la plage de données, mais aussi, d'indiquer si la première ligne contient des noms de champs et de personnaliser les propriétés des variables (noms, libellés, type, informat, formats). Le flux de processus est de ce type :



Il peut être ré-exécuté quand le fichier Excel est mis à jour pour ré-importer les données en format SAS et enchaîner ensuite des traitements.

Une tâche similaire existe pour l'exportation. On y accède par clic droit sur la table SAS à exporter. Deux options sont possibles :

- une exportation ponctuelle (clic droit > Exporter > Exporter <nom table> ...) : le fichier sera créé, mais aucune trace ne sera présente dans le flux de processus
- ou une exportation qui pourra être rejouée (clic droit > Exporter > Exporter <nom table> en tant qu'étape du projet...). Le flux est alors de ce type :



2.2. Le format CSV

Le format csv a pour avantage d'être directement compris par Excel. L'idée est donc ici d'exporter vos données SAS dans ce format. Au niveau de la lecture, une étape est supplémentaire dans Excel : le fichier Excel devra être enregistré au format csv pour être ensuite lu dans SAS.

Pour importer un fichier csv, la méthode la plus simple est d'utiliser une proc import avec l'option DBMS=CSV.

En France, le séparateur de champs proposé dans un fichier csv est le point virgule, il faudra donc indiquer cette information dans la procédure import, au niveau de l'instruction delimiter :

```
PROC IMPORT DATAFILE="c:\mon_fichier.csv" OUT=importCSV REPLACE dbms=CSV;
delimiter=';';
RUN;
```

Le fonctionnement est identique au niveau de la procédure export. Si l'option delimiter est omise, SAS utilisera par défaut la virgule.

Techniquement, une étape data est générée à la volée par SAS, avec des instructions infile/input pour la proc import, et file/put pour la procédure export. Ce code peut être récupéré si vous utilisez une session SAS classique, avec la touche F4 (ou par le menu Exécuter > Rappeler dernière exécution).

2.3. Les liens DDE

Les liens DDE étaient, voici une vingtaine d'années, la solution sûrement la plus utilisée pour faire communiquer SAS et Excel. Mais elle atteint aujourd'hui ses limites. Ecrire de nouveaux programmes en utilisant cette technologie est plutôt à proscrire, même si nous incluons toujours ce sujet dans notre documentation [en version 9.4](#) et qu'elle fonctionne toujours.

Certes, elle fonctionne toujours, mais elle a des restrictions, en particulier celle-ci : pour fonctionner, SAS et Excel doivent être installés sur la même machine Windows. Cela rend donc impossible d'exécuter du code utilisant les liens DDE depuis Enterprise Guide, connecté à un serveur distant.

2.4. SAS® Add-In for Microsoft Office

Cet outil, si votre société en dispose dans sa licence, quand il est installé dans Excel, permet l'ajout d'un menu complet :



Ainsi, vous avez accès à toutes les bibliothèques SAS définies au niveau du serveur sur lequel vous êtes connecté et donc, toutes les tables qui y sont référencées.

En ouvrant une table (par l'icône « Données SAS »), les données sont automatiquement chargées dans Excel.

Nous vous recommandons de prendre connaissance de cette Installation Note :

“[Installation Note 38120](#): Support for SAS® Add-In for Microsoft Office in Microsoft Office 2016, 2013, and 2010”

pour bien vérifier les versions de SAS Add-In supportées par la version d'Excel dont vous disposez ou vers laquelle vous avez l'intention de migrer. Ainsi, si vous comptez déployer MS Office 2016, il vous faudra également mettre à jour SAS Add-In for Microsoft Office, avec la dernière version aujourd'hui disponible : la version 7.12.

3. EXPORTER, EN PASSANT PAR DES RAPPORTS

Une simple `proc print`, exécutée sur la table dont on souhaite exporter les données en utilisant une destination ODS, est également un autre type d'approche possible.

En termes de destinations, nous les avons étoffées au fil des versions.

3.1. ODS HTML

L'ODS HTML a fait partie des premières destinations possibles. En spécifiant une extension xls, le fichier créé peut être ouvert directement par Excel.

Exemple :

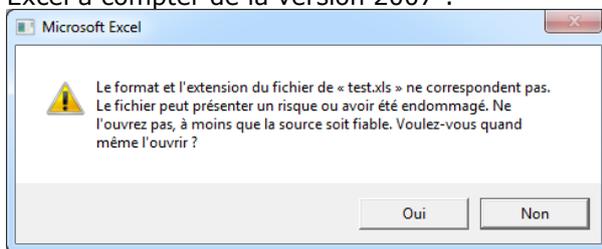
```

title;
ods html file='c:\test.xls'
style=minimal;
proc print data=sashelp.class
(obs=10);
run;
ods html close;

```

	A	B	C	D	E	F
1	Obs.	Name	Sex	Age	Height	Weight
2	1	Alfred	M	14	69.0	112.5
3	2	Alice	F	13	56.5	84.0
4	3	Barbara	F	13	65.3	98.0
5	4	Carol	F	14	62.8	102.5
6	5	Henry	M	14	63.5	102.5
7	6	James	M	12	57.3	83.0
8	7	Jane	F	12	59.8	84.5
9	8	Janet	F	15	62.5	112.5
10	9	Jeffrey	M	13	62.5	84.0
11	10	John	M	12	59.0	99.5
12						

Un inconvénient, non négligeable, est l’affichage systématique d’un message d’avertissement dans Excel à compter de la version 2007 :



3.2. ODS TAGSETS.EXCELXP

Le tagset Excelxp permet de générer un fichier XML, qui sera lisible par Excel. Il présente l’avantage de pouvoir contenir plusieurs feuilles. Mais le même message d’avertissement est affiché à l’ouverture du fichier dans Excel.

Exemple :

```

title;
ods tagsets.excelxp
file='c:\test.xls'
style=minimal;

proc print
data=sashelp.class (obs=10);
run;
proc print
data=sashelp.prdsale
(obs=10);
run;

ods tagsets.excelxp close;

```

	A	B	C	D	E
1	Obs.	ACTUAL	PREDICT	COUNTRY	REGION
2	1	925,00 €	850,00 €	CANADA	EAST
3	2	999,00 €	297,00 €	CANADA	EAST
4	3	608,00 €	846,00 €	CANADA	EAST
5	4	642,00 €	533,00 €	CANADA	EAST
6	5	656,00 €	646,00 €	CANADA	EAST
7	6	948,00 €	486,00 €	CANADA	EAST
8	7	612,00 €	717,00 €	CANADA	EAST
9	8	114,00 €	564,00 €	CANADA	EAST
10	9	685,00 €	230,00 €	CANADA	EAST
11	10	657,00 €	494,00 €	CANADA	EAST
12					

3.3. ODS EXCEL

L’ODS EXCEL est expérimental à compter de SAS 9.4 Maintenance 1 et est production dans la maintenance 3.

Il réunit les avantages des destinations ODS suivantes : CSV, EXCELXP et MSOffice2K. Nous pouvons mettre en avant les bénéfices suivants :

- La taille du fichier Excel créé est inférieure, jusqu'à 75%, à celle obtenue avec les autres destinations ODS.
- Il est compatible avec la version Excel 2010 et versions supérieures (pas de message à l'ouverture dans Excel).
- En plus des tableaux, des graphiques peuvent être intégrés.
- La génération de multiples feuilles au sein d'un même fichier Excel.
- Une meilleure intégration avec l'ODS. Par exemple, l'instruction ODS Text est acceptée.

Quelques exemples :

```
ods excel file="c:\temp2.xlsx"
options(start_at="C4" tab_color="purple" absolute_row_height="20"
embedded_titles="yes" frozen_headers='ON');
```

```
title "Exemple de titre";
ods text="Exemple de texte ajouté avec ODS TEXT";
proc print data=sashelp.class(obs=10);
run;
ods excel close;
```

L'option start_at permet de spécifier la cellule où l'écriture commencera.

L'option tab_color permet d'indiquer une couleur à l'onglet correspondant à la feuille.

L'option absolute_row_height permet de définir la hauteur des lignes.

L'option embedded_title permet d'écrire le titre (instruction title) dans le fichier Excel (sans l'option, l'instruction title est ignorée).

L'option frozen_headers permet de figer un volet sous la ligne contenant le nom des variables.

Obs	Name	Sex	Age	Height	Weight
1	Alfred	M	14	69,0	112,5
2	Alice	F	13	56,5	84,0
3	Barbara	F	13	65,3	98,0
4	Carol	F	14	62,8	102,5
5	Henry	M	14	63,5	102,5
6	James	M	12	57,3	83,0
7	Jane	F	12	59,8	84,5
8	Janet	F	15	62,5	112,5
9	Jeffrey	M	13	62,5	84,0
10	John	M	12	59,0	99,5

L'option sheet_interval est également intéressante car elle permet d'indiquer sous quelle condition de nouvelles feuilles sont créées. Par défaut, chaque tableau demandé est inséré par feuille. La valeur 'none' permet de créer une feuille unique, la valeur 'proc' d'avoir une feuille par procédure, etc.

4. EN CAS DE PROBLÈME

4.1. Éléments à transmettre au Support Clients

Si vous rencontrez des problèmes lors de l'importation et l'exportation de fichiers Excel, vous pouvez nous écrire à support@sas.com, en attachant à votre message l'erreur reçue.

5. LIENS UTILES

Dans la documentation SAS 9.4 :

- [SAS/ACCESS 9.4 Interface to PC Files: Reference, Third Edition](#)
- [ODS EXCEL](#)

La [page de référence](#) sur l'ODS en relation avec les produits Microsoft Office.

Dans notre base de connaissances :

[Installation Note 38120](#): Support for SAS® Add-In for Microsoft Office in Microsoft Office 2016, 2013, and 2010

6. CONCLUSION

En résumé, nous vous encourageons :

- à vérifier que vous disposez du module SAS/Access to PC Files pour les importations/exportations de fichiers Excel, auquel cas à contrôler que vous disposez au moins de la maintenance 2 de SAS 9.4 et à privilégier alors l'utilisation du moteur XLSX (option `dbms=xlsx` dans `proc import` et `proc export`)
- à découvrir l'ODS EXCEL, à partir de la maintenance 1 de SAS 9.4, mais de préférence avec la maintenance 3 (dans laquelle cette destination ODS est officiellement production) pour la production de vos rapports en format Excel.

Karine CHRILLESEN
Consultante Support Clients SAS France