



**SAS[®] Integration Technologies
sur OS/390 et z/OS :**

**Object Spawner SAS[®] 9.0
et
IOM Server SAS[®] 8.2**



SOMMAIRE

Introduction	1
1. SAS/IT et OS/390 et z/OS : de la technologie VTAM [®] /APPC aux Unix System Services.....	1
2. Installation de l'Object Spawner SAS/IT 9.....	2
a. Préparation de l'installation.....	2
b. L'environnement d'installation sous USS.....	2
c. Le fichier de configuration de l'Object Spawner	3
d. Le « shell script »	4
e. La CLIST du Serveur IOM SAS	5
f. Sécurité OS/390 – Sécurité Unix	6
g. Le lancement de l'Object Spawner.....	6
3. Test de l'Object Spawner SAS Integration Technologies 9.x.....	7
a. L'interface d'administration de l'Object Spawner	8
b. ITConfig.exe.....	9
c. Module Administrator de SAS [®] Enterprise Guide [®]	12
4. En cas de problème.....	14
5. Ressources – Bibliographie	15

Introduction

Ce document a pour but de présenter l'installation de l'Object Spawner SAS 9.0 sur serveur OS/390 ou z/OS, et son fonctionnement avec des serveurs SAS IOM 8.2.

Nous expliquerons notamment ici l'utilisation de cette installation avec un client SAS® Enterprise Guide®.

L'Object Spawner est un composant du module SAS Integration Technology (SAS/IT). Il s'agit d'un démon TCP, qui permet à des clients non SAS de se connecter à une session SAS IOM.

Les clients en question peuvent être, par exemple :

- SAS Enterprise Guide
- Des applications développées avec AppDev Studio™ (WebAF™, WebEIS™)
- Des applications Visual Basic, Visual C++
- Des applications JAVA
- ...

Un serveur IOM (Integrated Object Model) permet d'accéder aux ressources des logiciels SAS au travers d'interfaces objets distribuées, utilisables en standard par les technologies ci-dessus.

Ce nouvel Object Spawner est le même que celui disponible avec SAS 9. Nous l'appellerons dans ce document Object Spawner, ou Spawner SAS/IT. Il ne faut pas le confondre avec le Spawner utilisé par le module SAS/CONNECT®, qui n'est pas abordé ici.

1. SAS/IT et OS/390 et z/OS : De la technologie VTAM®/APPC aux Unix System Services

La technologie APPC, éprouvée et fiable, peut-être difficile à mettre en œuvre dans certains cas, principalement du fait que les personnes expérimentées dans ce domaine sont de plus en plus rares. Cela nécessite généralement la participation de plusieurs personnes, pour les définitions APPC, VTAM et TCP/IP.

Ce constat, ainsi que la tendance générale du marché et les vœux formulés par les utilisateurs, ont conduit SAS à se tourner vers l'utilisation des Unix System Services comme moyen pour le Spawner SAS/IT de lancer des serveurs SAS IOM.

Fonctionnement de l'Object Spawner SAS 8.2 :

L'Object Spawner est un démon TCP, qui écoute sur un port. Le client se connecte sur ce port, et fournit ses éléments d'autorisation. Une fois ceux-ci validés, l'Object Spawner utilise un TP APPC pour lancer une session SAS IOM dans un espace-adresse APPC. L'Object Spawner donne alors la socket du client au serveur IOM (service TCP *givesocket*), et le serveur la prend (*takesocket*)

Fonctionnement de l'Object Spawner SAS 9 :

L'Object Spawner est toujours un démon TCP, en écoute sur un port. Le client se connecte sur ce port, et est identifié. L'Object Spawner lance (à l'aide de la fonction TCP *spawn*) la commande spécifiée dans son fichier de configuration, qui est en fait un « shell script » Unix, et qui va lancer la session SAS IOM avec les paramètres adéquats. L'Object Spawner utilisant le service TCP *spawn*, il peut facilement passer la socket du client vers le serveur.

2. Installation de l'Object Spawner SAS/IT 9

a. Préparation de l'installation

Il y a 2 composants à installer, sous forme de hotfixes à télécharger (cf. Chap. 5) :

- Le 82IH13, qui contient un PDS avec les load modules de l'Object Spawner proprement dit.
La bibliothèque de l'Object Spawner doit être installée séparément des bibliothèques SAS 8.2. Pour cet exemple, elle sera nommée OBJSPAWN.V9.LOAD.
- Le 82BB92, qui contient une nouvelle version du load module SASWQS
Le nouveau load module remplace celui présent dans la bibliothèque SAS.LIBRARY.
Il est compatible avec l'Object Spawner 8.2 (APPC).
Ce module sera appelé par la CLIST définie plus bas.

b. L'environnement d'installation sous USS

Sous OMVS ou ISHELL, il faut créer un répertoire qui contiendra les fichiers de l'Object Spawner dans le système de fichier USS (HFS). Dans ce document, le répertoire /objspawn9 sera utilisé.

Il faut créer un lien externe vers le module OBJSPAWN sur MVS :

```
ln -e OBJSPAWN objspawn
```

Il faut ensuite indiquer au système l'emplacement du load module.

Le lien externe défini plus haut cherchera le module d'abord à l'aide de la variable d'environnement STEPLIB, puis en suivant l'ordre normal sous MVS (LPA, linklist, ...)

```
export STEPLIB='OBJSPAWN.V9.LOAD'
```

Si la « SAS/C transient library » n'est pas en linklist, il faut également créer une variable d'environnement pour cela :

```
export ddn_TRANS=SAS.SASC.TRANSLIB
```

La concaténation de cette bibliothèque en STEPLIB ne permettra pas à l'Object Spawner de fonctionner.

c. Le fichier de configuration de l'Object Spawner

Nous appellerons ici ce fichier objspawn.conf :

```
dn: sasSpawnercn=MySpawner
objectClass: sasSpawner
sasSpawnercn: MySpawner
sasMachineDNSName: localhost
sasOperatorPort: 9902
sasOperatorPassword: abc
sasVerbose: true
sasLogfile: /objspawn9/objspawn.log

# This Server runs 8.2
dn: sasServercn=MyServer82
objectClass: sasServer
sasPort: 9903
sasProtocol: bridge
sasCommand: /objspawn9/sas82.sh nosasuser --
sasMachineDNSName: localhost
sasServercn: MyServer82
```

Le premier bloc définit l'Object Spawner, le deuxième un serveur IOM 8.2

On peut définir autant de type de serveur que l'on veut, chacun avec un numéro de port associé, et tous gérés par le même Object Spawner.

Les blocs doivent être séparés par une ligne vide.

sasOperatorPort : port de l'interface d'administration de l'Object Spawner

sasOperatorPassword : mot de passe administrateur

sasPort : port de connexion du client

sasCommand : commande lancée dans USS pour démarrer le serveur IOM.

Cette commande est détaillée dans le paragraphe suivant. Le double tiret « -- » sert de balise, pour permettre au script de séparer les arguments statiques et les arguments dynamiques passés au serveur IOM.

Ce fichier de configuration doit être accessible en lecture par tous :

```
chmod 644 objspawn.conf
```

d. Le « shell script »

L'Object Spawner utilise un « shell script » Unix pour lancer la session SAS dans un espace-adresse USS (BPXAS). Pour cet exemple, il sera appelé sas82.sh :

```
#!/bin/sh
#
echo `date` "Demarrage IOM Server. \n"
start=`date`
#
# foundDashDash est un booleen, a TRUE quand la chaine
# "--" est trouvee.
#
foundDashDash=0
#
# Construction des arguments
#
args=''
for arg in "$@" ; do
if [ "$arg" != "--" ]; then
tmp="$arg ";
else
tmp="SRVOPTS('";
foundDashDash=1;
fi
args="$args$tmp"
done
#
# Si on trouve "--", on doit fermer l'option SRVOPTS
#
if [[ $foundDashDash -ne 0 ]]; then
args="$args '"
fi
#
# Construction de la ligne de commande
#
cmd="/bin/tso -t EX 'SAS.CLIST(SAS82IOM)'"
cmd="$cmd '$args'"
#
# Variables d'environnement...
# On utilise account data pour placer SAS
# dans le service class adequat
#
export _BPX_ACCT_DATA=S100
echo `date` "Lancement de SAS par :" $cmd "\n"
#
# Lancement de SAS
#
$cmd > /objspawn9/shell.log
echo `date` "Le Server IOM demarre le " $start "est stoppe. \n"
```

La ligne `cmd="/bin/tso -t EX 'SAS.CLIST(SAS82IOM)'"` doit faire référence à la CLIST qui lancera la session SAS IOM. Celle-ci est définie dans le chapitre suivant.

Ce fichier doit être autorisé en lecture et exécution pour tous :

```
chmod 755 sas82.sh
```

e. La CLIST du Serveur IOM SAS

C'est une CLIST SAS 8.2, légèrement modifiée pour pouvoir lancer un Server IOM. Il est conseillé de partir de la CLIST utilisée par le Spawner SAS/CONNECT, qui contient déjà l'option NOSASUSER. Celle-ci se trouve par défaut dans SAS.CNTL(SPWNSAS8). Pour cet exemple, elle sera nommée SAS.CLIST(SAS82IOM)

Modifiez les variables de l'instruction PROC comme suit (ajout de la variable SRVOPTS) :

```
STAI NOSTAI /* Trap subtask abends? */ +
STAX NOSTAX /* Trap attentions? */ +
STACK NOSTACK /* Create new i/p stack? */ +
SHARE NOSHARE /* Share subpool 78 */ +
SASCLIENTPORT() /* Client socket from spawner*/ +
SASDAEMONPORT() /* Port to connect to spawner*/ +
SRVOPTS() /* for SAS IOM server */ +
NETENCALG() /* Encryption algorithm */ +
INHERIT(0) /* Get socket from spawner */ +
NOSASUSER /* Suppress SASUSER allocate */ +
TCPDEBUG(0) /* TCPIP Debugging */ +
TRACE /* CLIST debugging */
```

A la fin de la CLIST, ajoutez le test suivant :

```
/*
/* put system sort link library in STEPLIB if needed
/*
IF &STR(&SORTLINK) NE THEN +
SET SORTLDSN=&STR()
/*
/*
/* For IOM Server
/*
IF &STR(&SRVOPTS) NE THEN +
SET OPTIONS=&STR(&OPTIONS &SRVOPTS)
/*
/* Invoke SAS
/*
/*
SET TASKLIB = &STR(&LOAD &SASLOAD &SORTLDSN)
SET PARM=&STR(T(&TASKLIB) E(&ENTRY) O('&OPTIONS') &FLUSH &NOFLUSH +
&STAE &NOSTAE &STAI &NOSTAI &STAX &NOSTAX +
&STACK &NOSTACK &SHARE &NOSHARE)
SASCP &PARM
SET RC=&LASTCC
CONTROL NOMSG
IF &AUTOEXEC NE THEN +
FREE F(&DDAUTOEX)
EXIT CODE(&RC)
```

f. Sécurité OS/390 – Sécurité Unix

La tâche de l'Object Spawner doit avoir des autorisations particulières, pour pouvoir lancer un processus avec l'identifiant donné par le client demandant la connexion.

Du point de vue MVS, la bibliothèque OBJSPAWN.V9.LOAD doit être contrôlée (autorisée APF), et un user doit être défini dans la classe BPX.DAEMON.

Du point de vue Unix, pour que le démon TCP puisse utiliser l'identifiant reçu, il doit avoir soit des droits de « superuser » (UID=0), soit lancer une fonction `__passwd()` réussie avant la fonction `spawn()`. Ici, il n'est pas nécessaire que l'Object Spawner tourne avec un UID « superuser ».

g. Le lancement de l'Object Spawner

Il y a 3 méthodes pour lancer l'Object Spawner.

Le script Unix permet à un administrateur SAS de lancer l'Object Spawner sans privilège particulier concernant les STC MVS. Cela peut être particulièrement utile lors des phases de test de l'installation.

Les STC MVS ou BPXBATCH ont la préférence des administrateurs MVS. Dans le cas où l'Object Spawner doit se connecter à un serveur LDAP afin de lire des informations de configuration, il faudra plutôt utiliser le BPXBATCH. En effet, le champ PARM est limité à 100 caractères avec une procédure stockée MVS classique.

Script Unix :

Créez un script Unix qui contient les instructions suivantes :

```
#!/bin/sh
export STEPLIB="OBJSPAWN.V9.LOAD"
export ddn_CTRANS="SAS.SASC.TRANSLIB"
objspawn -cf objspawn.conf -sv -slf sasobj.log > objspawn.log
```

Si ce script est enregistré sous le nom startobj, l'Object Spawner est lancé en tâche de fond par la commande :

```
startobj &
```

BPXBATCH :

BPXBATCH est un utilitaire IBM qui permet d'exécuter des commandes shell Unix depuis un environnement MVS, en batch.

Voici un exemple de STC utilisant BPXBATCH pour lancer l'Object Spawner dans un espace-adresse USS :

```
//OBJSPNV9 PROC
//OBJSPNV9 EXEC PGM=IKJEFT01,REGION=4M
//SYSTSPRT DD SYSOUT=*
//STDENV DD *
STEPLIB=OBJSPAWN.V9.LOAD
_BPX_BATCH_UMASK=0644
_BPXK_JOBLOG=1
/*
//SYSTSIN DD *
BPXBATCH SH /objspawn9/objspawn -sv +
           -cf /objspawn9/objspawn.conf +
           -slf /objspawn9/sasobj.log +
           > /objspawn9/objspawn.log
/*
//
```

Il est recommandé de lancer un batch TSO (sous IKJEFT01), qui permet de ne pas limiter le paramètre PARM à 100 caractères.

Started Task MVS :

Enfin, on peut lancer l'Object Spawner dans un espace adresse MVS, en tant que STC classique :

```
//OBJSPAWN PROC ENTRY=OBJSPAWN,
// OPTIONS='-sv -cf /objspawn9/objspawn.conf -slf LOG '
// OPT2=' '
//*****
/* SAS/Integration Technologies Object Spawner
//*****
//OBJSPAWN EXEC PGM=&ENTRY,REGION=4096K,
// PARM='&OPTIONS &OPT2 '
//STEPLIB DD DISP=SHR,DSN=SASV9.OBJSPAWN.LOAD
//CTTRANS DD DISP=SHR,DSN=SAS.SASC.TRANSLIB
//LOG DD SYSOUT=*
//SYSOUT DD SYSOUT=*
//SYSPRINT DD PATH='/objspawn9/objspawn.stdout',
// PATHMODE=(SIRUSR,SIWUSR,SIRGRP,SIROTH),
// PATHOPTS=(OWRONLY,OCREAT,OTRUNC)
//SYSIN DD DUMMY
```

Avec cette méthode, le champ PARM est limité à 100 caractères, il faut donc être vigilant sur ce point.

Il faut noter que la carte DD SYSPRINT doit obligatoirement pointer vers un fichier HFS, afin que les erreurs éventuelles du serveur IOM puissent être enregistrées.

Le fichier de configuration, lui, peut éventuellement être un fichier MVS classique.

3. Test de l'Object Spawner SAS Integration Technologies 9.x

a. L'interface d'administration de l'Object Spawner

Une fois l'Object Spawner démarré, il doit être en écoute sur les ports définis dans le fichier de configuration (ici 9902 et 9903)

Depuis un poste client, on peut se connecter à l'interface d'administration de l'Object Spawner via telnet :

```
telnet host-name 9902
```

Une fois connecté, il faut taper le mot de passe¹, lui aussi défini dans le fichier de configuration. Le terminal confirme l'établissement de la connexion avec le message suivant :

```
Operator conversation established
```

Les différentes commandes disponibles sont :

help	Lister les commandes disponibles
list	Lister les serveurs disponibles avec cet Object Spawner
btrace	Démarrer un fichier log. Sur OS/390, la commande est de la forme : <code>btrace //dsn:userid.objspawn.log</code>
etrace	Stopper (et libérer) le fichier log
quit	Quitter l'interface d'administration
bye	Arrêter l'Object Spawner

¹ Sous Windows, par défaut, le client telnet ne renvoie pas les caractères saisis. Il faut donc soit taper les commandes en aveugle, soit passer la commande **set local_echo** :

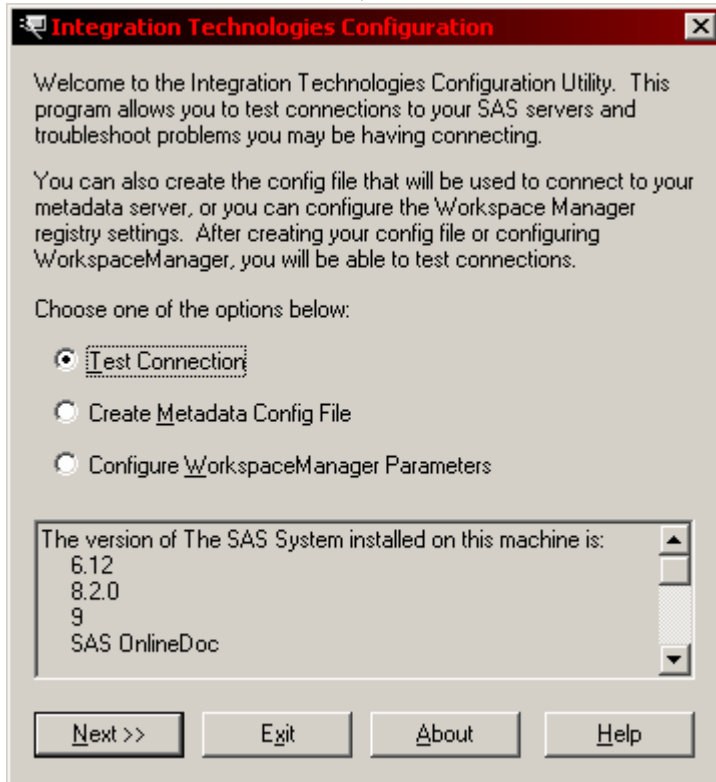
```
C:\> telnet
Microsoft Telnet> set local_echo
Microsoft Telnet> open host-name 9902
```

b. ITConfig.exe

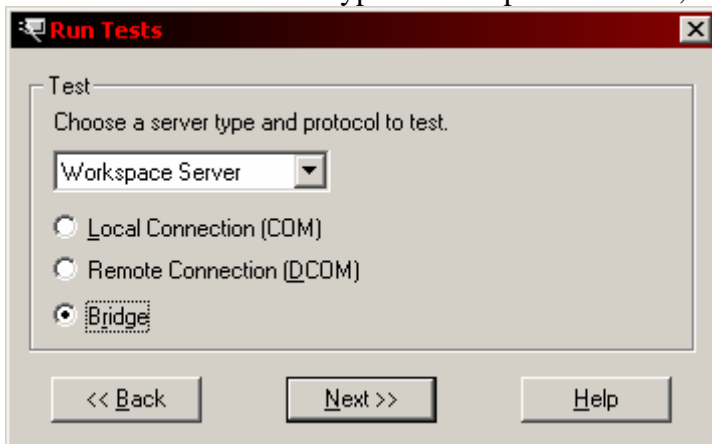
Ce programme est livré avec SAS Enterprise Guide 2.0, et il est également disponible sur le site de SAS (cf. Chap. 5). Par défaut, il est installé dans le répertoire :

C:\Program Files\SAS Institute\Shared Files\Integration Technologies\

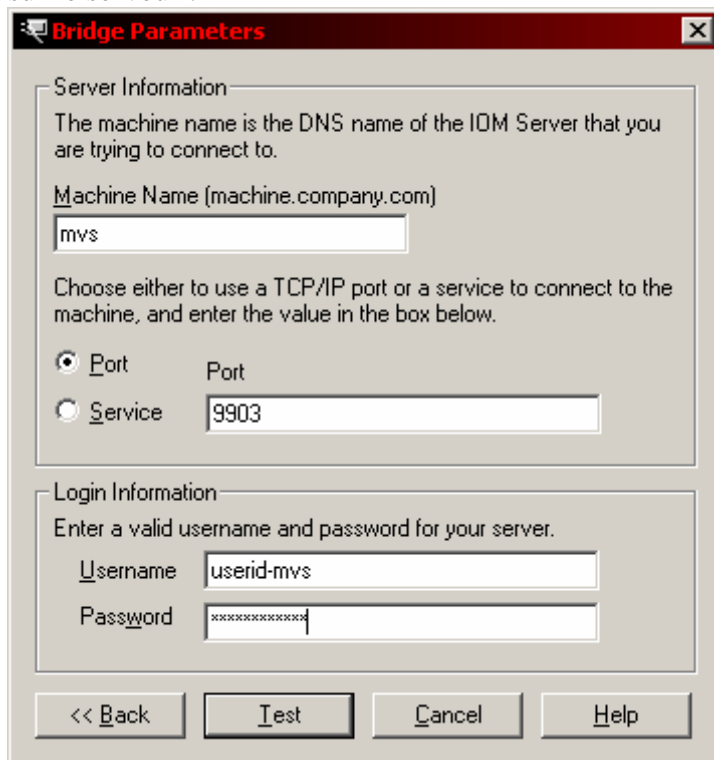
Au lancement de l'utilitaire, choisissez « Test Connection » :



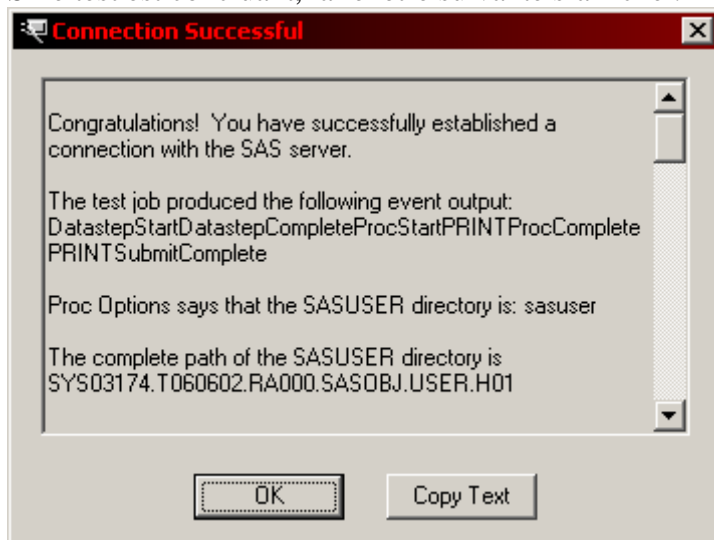
Le serveur à tester est de type « Workspace Server », et le protocole « Bridge » :



Donner ensuite le nom TCP/IP de la machine, le numéro de port du serveur, et on s'identifie sur le serveur :



Si le test est concluant, la fenêtre suivante s'affiche :



Résultat complet du test soumis au serveur IOM par ITConfig :

```
Congratulations! You have successfully established a connection with the SAS server.
```

```
The test job produced the following event output:  
DatastepStartDatastepCompleteProcStartPRINTProcCompletePRINTSubmitComplete
```

```
Proc Options says that the SASUSER directory is: sasuser
```

```
The complete path of the SASUSER directory is  
SYS03174.T060602.RA000.SASOBJ.USER.H01
```

```
MachineDNSName (according to SAS): *****.***.***
```

The version number of the SAS Supervisor. This version number identifies both the release and the particular build of the SAS System that is running on the server. Build version information is helpful for problem determination by SAS Technical Support.

--The SYSVLONG macro is the SAS language equivalent:
8.02.02M0P02222001

A specific operating system name for the server.

--The SYSSCPL macro is the SAS language equivalent: MVS

A generic operating system name for the server.

--The SYSSCP macro is the SAS language equivalent: OS

A version number for the operating system.

--The format of the string depends on the operating system: SP7.0.4

An operating system-specific identifier for the process, job or address-space in which the SAS System is executing.

--The SYSJOBID macro is the SAS language equivalent: SASOBJ

This slot contains the operating system user name under which the SAS server is executing.

--The SYSUSERID macro is the SAS language equivalent: *****

The operating system user name of the client: *****

The hardware model name of each processor in the server computer. If the server has multiple processors, they are very likely to have the same model name. The format of this name will vary by hardware vendor: IBM

The hardware model identifier for each processor in the server computer. If the server has multiple processors, they are very likely to have the same model number. The format of this identifier will vary by hardware vendor: IBM

The hardware serial number for each processor in the server computer. The format of this identifier will vary by hardware vendor: ****

```
1          data work.a; x=5; y=10; run;
```

NOTE: The data set WORK.A has 1 observations and 2 variables.

NOTE: The DATA statement used 0.02 CPU seconds and 8993K.

```
1          !                               proc print data=work.a; run;
```

NOTE: There were 1 observations read from the data set WORK.A.

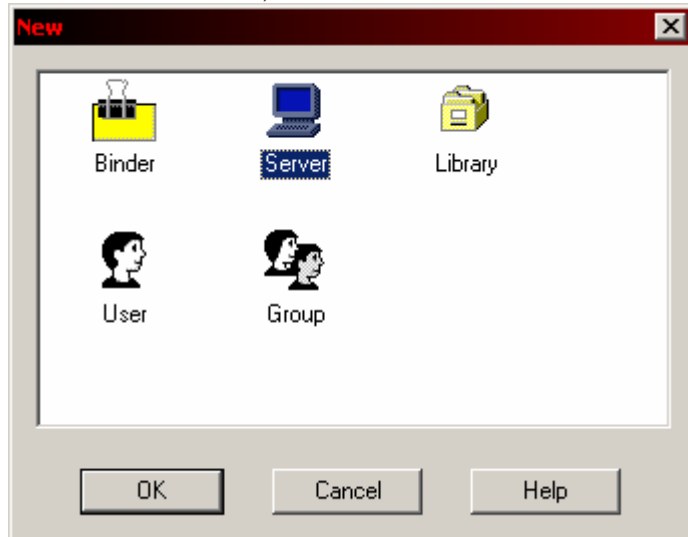
NOTE: The PROCEDURE PRINT printed page 1.

NOTE: The PROCEDURE PRINT used 0.02 CPU seconds and 9978K.

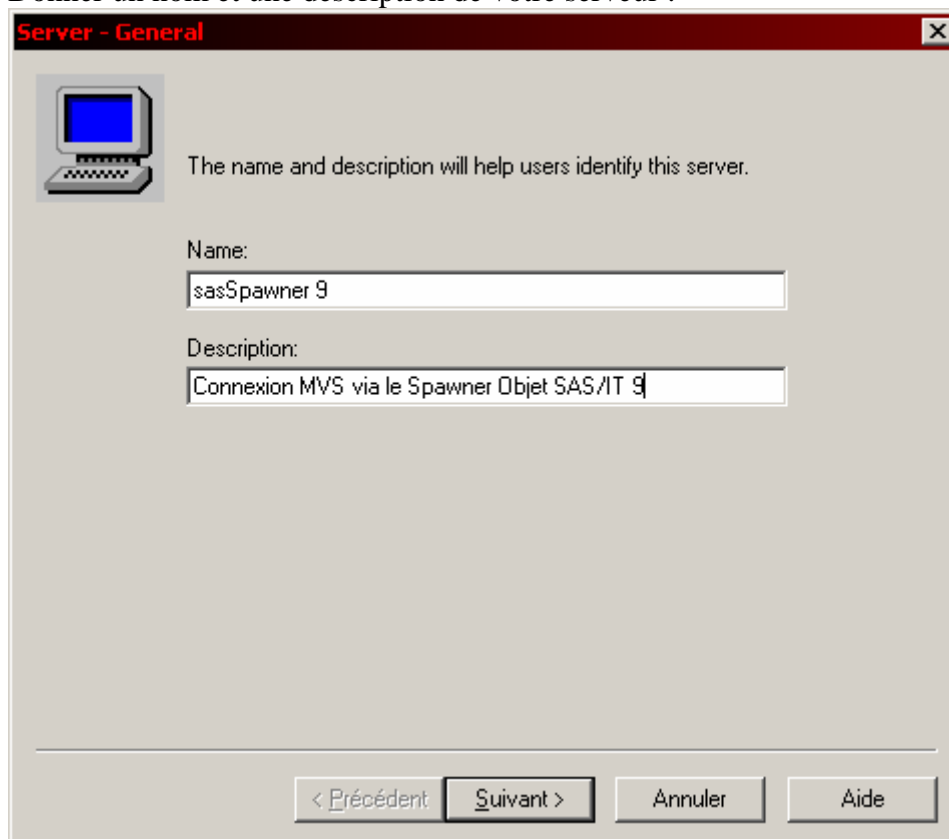
c. Module Administrator de SAS® Enterprise Guide®

Il reste à présent à configurer un nouveau serveur dans le module Administrator, afin de pouvoir accéder à notre serveur MVS depuis SAS Enterprise Guide.

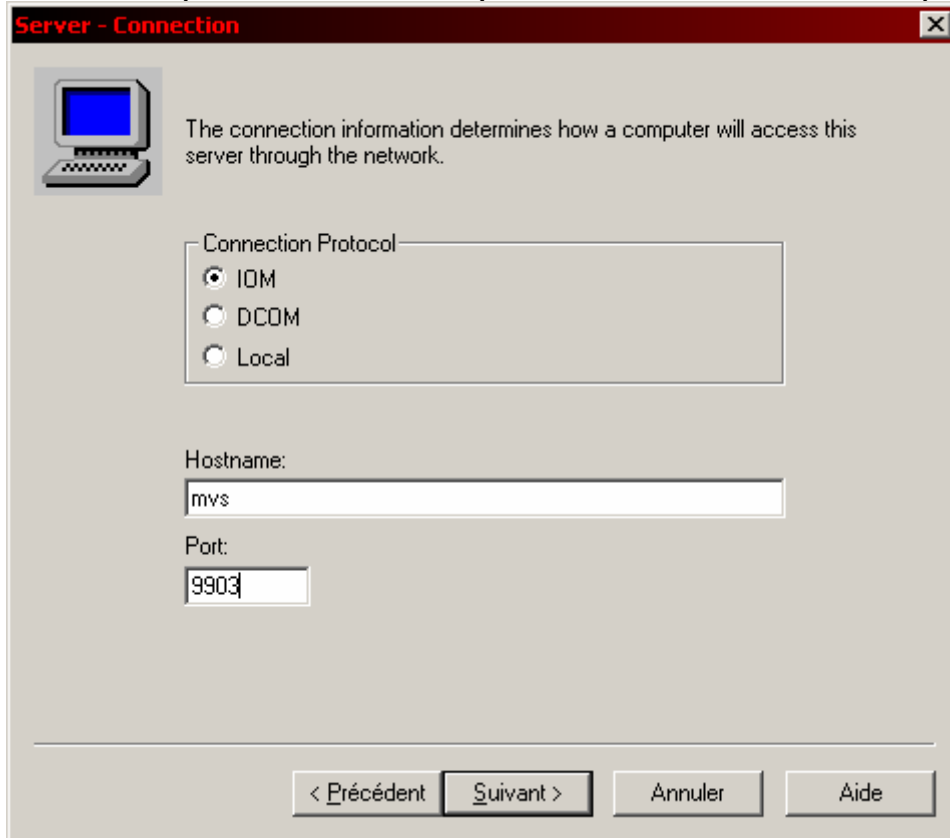
Dans Administrator, faire Fichier > Nouveau...



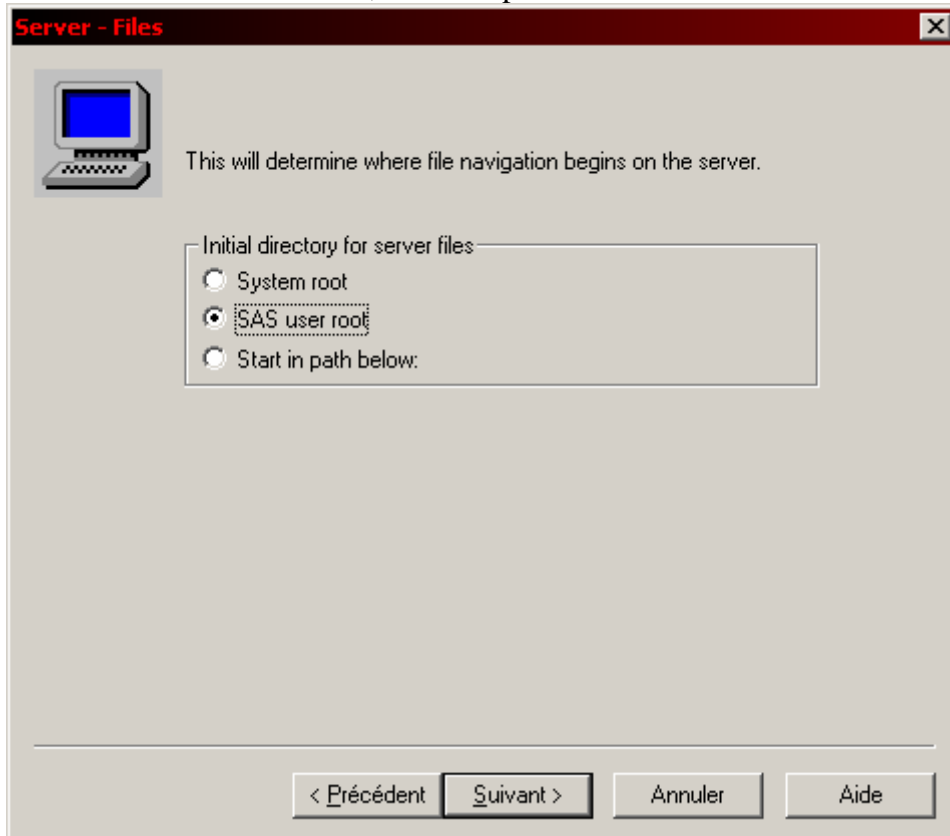
Donner un nom et une description de votre serveur :



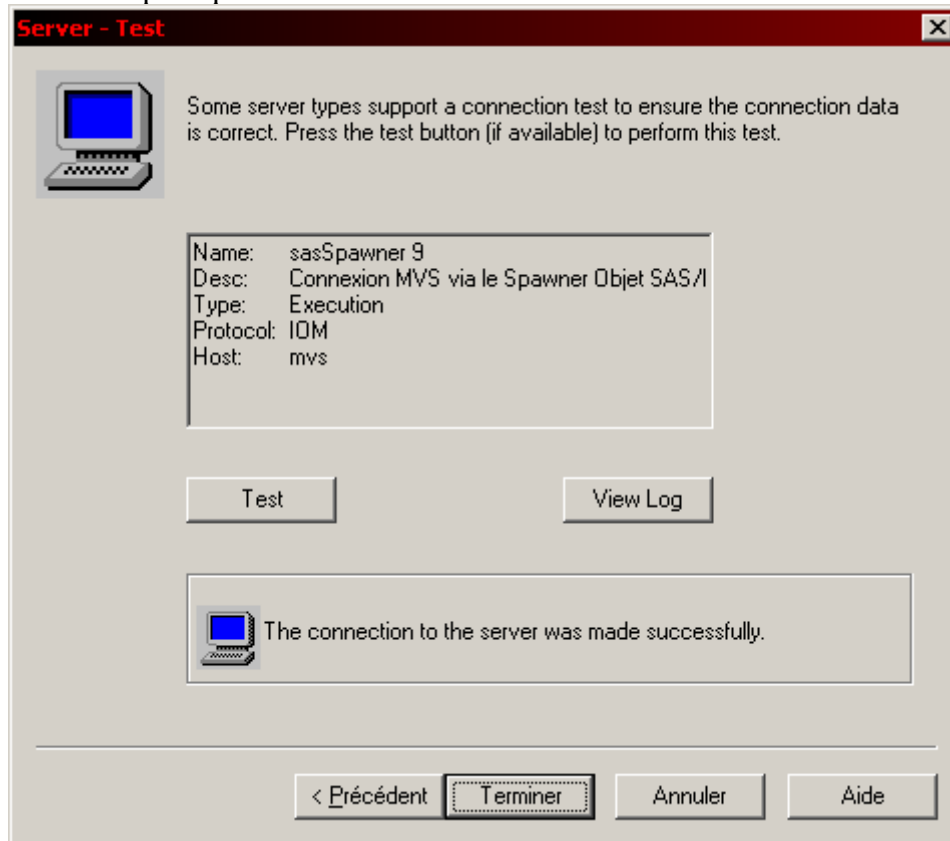
Choisissez le protocole IOM, et indiquer le nom de votre machine et le port de connexion :



Avec un serveur sur OS/390, il faut impérativement sélectionner la racine utilisateur :



Il ne reste plus qu'à tester la connexion :



4. En cas de problème...

En cas de problème lors de l'installation, il faut dans un premier temps essayer d'isoler le composant en cause.

Si l'Object Spawner ne démarre pas, vérifier la log de l'Object Spawner.

Si l'Object Spawner démarre, mais que la connexion échoue :

- Vérifiez les logs du client, de l'Object Spawner et du « shell script ».
- Est-ce que l'Object Spawner est en écoute sur les 2 ports TCP ? Sur OS/390, utilisez la commande : `TSO NETSTAT (PO 9902)`
- Est-ce qu'un espace adresse BPXAS démarre ? Si oui, il y a également des messages dans cet espace adresse.
- S'il ne démarre pas, vérifiez la commande de lancement dans le « shell script ».
- Testez la CLIST sous TSO.

Si vous avez créé le script directement sous OMVS (avec VI par exemple), essayez en créant le fichier avec un éditeur de texte sous Windows, puis en transférant le fichier par FTP (en mode texte).

Enfin, vous pouvez vous adresser au Support Client SAS. Il faudra fournir, lors de l'ouverture du dossier :

- le fichier de configuration
- le « shell script »
- la STC de l'Object Spawner, si elle est utilisée
- le CLIST SAS 8.2
- les logs de l'Object Spawner, du « shell script », éventuellement de l'espace adresse BPXAS
- la log du test ITConfig
- la log du test avec Administrator
- ...

5. Ressources – Bibliographie

SN-008715 : OS/390 Integration Technologies object spawner supports UNIX System Services

<http://support.sas.com/techsup/unotes/SN/008/008715.html>

Hotfix 82IH13 (~1.7 M)

http://www.sas.com/techsup/download/hotfix/82_mv_s_bcs.html#82IH13

Hotfix 82BB10 (~68 K)

http://www.sas.com/techsup/download/hotfix/82_mv_s_bcs.html#82BB92

SN-007626 : SAS IOM Server requires manual intervention to cancel orphaned or "hung" SAS process

<http://support.sas.com/techsup/unotes/SN/007/007626.html>

ITConfig.exe (~2.3 M)

<http://www.sas.com/apps/demosdownloads/setupintro.jsp>

Puis :

- SAS Integration Technologies
- SAS Integration Technologies 8.2 Client Components
- Windows Client Development Component (inttech.exe)

Technical Support Document : TS-678

<http://ftp.sas.com/techsup/download/technote/ts678.html>

SAS Integration Technologies 9 Documentation

<http://support.sas.com/rnd/itech/library/library9.html>



SAS France
Domaine de Grégy - BP 5
77166 Grégy-sur-Yerres
Tél. : 01 60 62 11 11
Fax : 01 60 62 11 99

SAS Europe, Middle East & Africa
P.O. Box 10 53 40
Neuenheimer Landstr. 28-30
D-69043 Heidelberg, Germany
Tel: +49 6221 4160, Fax: +49 6221 474850

SAS, le Système SAS® sont les marques déposées de SAS Institute Inc., Cary NC, USA.
Les autres noms de produits ou concepts sont des marques déposées des sociétés respectives.

www.sas.com/france