

GESTION DE LA MEMOIRE SOUS UNIX

La mémoire est une ressource critique car elle conditionne les temps de traitements. Différentes options SAS sont disponibles pour utiliser au mieux la mémoire de la machine. Cet article vous les présente, et propose une illustration des résultats obtenus.

Les résultats présentés dans cet article ont été réalisés avec SAS 9.3, sur une machine Solaris.

Caractéristiques :

Catégories : Unix
 OS : Unix
 Version : SAS® 9.2, SAS® 9.3
 Vérifié en novembre 2012

Table des matières

Gestion de la mémoire sous UNIX.....	1
Introduction.....	1
MEMSIZE.....	2
SORTSIZE	3
REALMEMSIZE.....	3
Bonnes pratiques.....	4
Où positionner les options ?	4
Comment connaître la consommation mémoire d'une procédure ou d'une étape DATA ?.....	5
Comment connaître les valeurs de la session SAS en cours ?.....	5
Liens utiles	5
Conclusion.....	6

Introduction

D'un point de vue système, ajuster la mémoire est l'une des principales méthodes pour augmenter ou réduire les performances d'un processus.

Lors de l'installation, SAS® Foundation est paramétré par rapport à un fonctionnement standard. Mais vous pouvez être amené à travailler sur des volumes importants de données et, dans ce cas, l'utilisation de certaines procédures très consommatrices en termes de ressources peut devenir problématique, d'autant plus si vous êtes en environnement multi-utilisateurs.

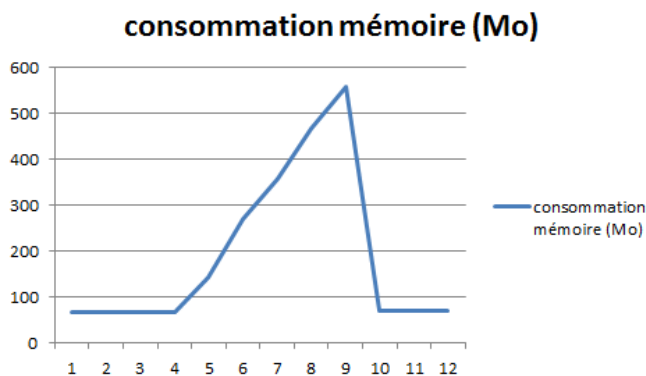
En effet, par défaut, **la valeur de l'option MEMSIZE est positionnée à 512 Mo**. Cela signifie que toute session SAS ne pourra pas allouer plus de **512 Mo** de mémoire sur le serveur.

Le graphique ci-contre présente l'évolution de la consommation mémoire d'un processus SAS durant l'exécution d'une procédure consommatrice (PROC MEANS, PROC SORT).

Si le traitement est trop consommateur en mémoire, on note que la consommation mémoire chute à 512 Mo.

Cela se traduit par le message suivant dans SAS® Enterprise Guide® par ex. :

FATAL: Insufficient memory to execute DATA step program. Aborted during the EXECUTION phase.



Il est donc nécessaire de paramétrer SAS d'une façon appropriée au travail effectué. Les principaux paramètres liés à l'utilisation de la mémoire sur SAS sont les suivants :

- Memsize
- Sortsize
- Realmemsize

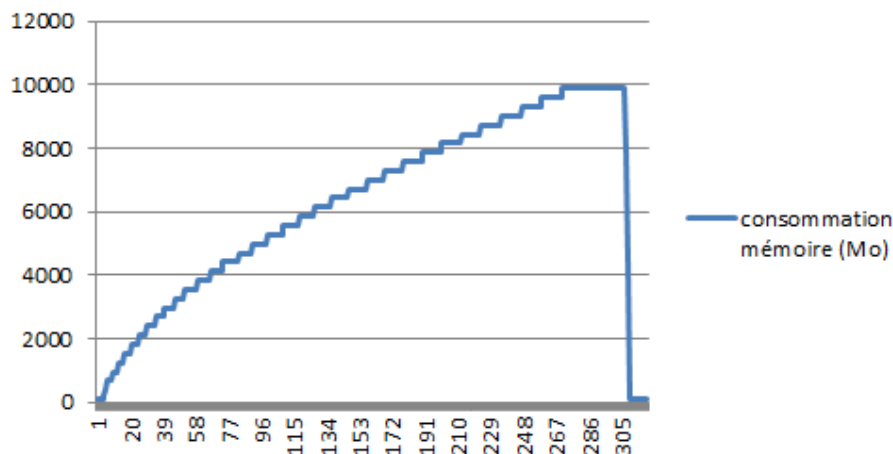
MEMSIZE

MEMSIZE représente la mémoire allouée au processus SAS. Au démarrage, le processus ne consommera que quelques méga-octets (moins de 70 Mo). La valeur de MEMSIZE est donc la valeur maximale que pourra s'allouer le processus.

Il est possible de positionner cette valeur à MAX. Avec cette valeur, **MEMSIZE** est positionné à 80% de la mémoire physique du serveur. Cependant, dans une architecture de type Business Analytics, positionner **MEMSIZE** à **MAX** pour le Workspace Server n'est pas judicieux : si plusieurs sessions sont lancées sur le même serveur (ce qui est un fonctionnement normal lorsque l'on utilise un serveur) et que chaque session peut potentiellement utiliser 80% de la mémoire physique, cela risque de poser des problèmes de ressources systèmes.

Dans cet exemple, le serveur dispose de 16 Go de mémoire. La courbe ci-dessous présente la consommation mémoire lorsque MEMSIZE est positionné à 10 Go :

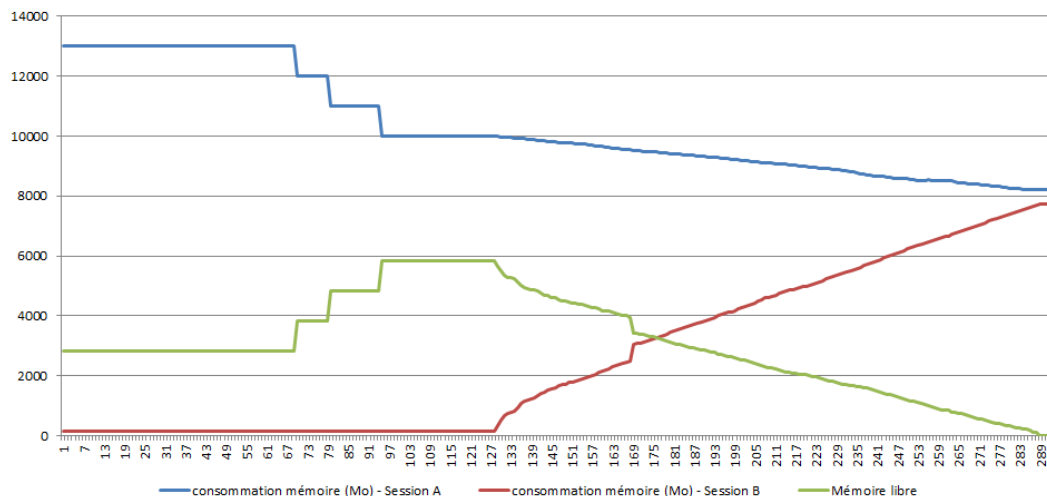
consommation mémoire (Mo)



La valeur de l'option MEMSIZE est à manipuler avec précaution. Une mauvaise valeur en production peut provoquer une consommation trop importante de mémoire. Donc, parfois un message «*Insufficient memory*» est préférable à une défaillance générale du serveur.

Pour illustrer ce problème, nous pouvons lancer deux sessions SAS consommatrices.

Le graphique ci-dessous présente la consommation de deux sessions SAS (consommation mémoire d'un Workspace server). La courbe verte présente la mémoire disponible sur le serveur.



La solution consiste donc, comme évoqué précédemment, à limiter la valeur du MEMSIZE. Cela pourra potentiellement pénaliser certains traitements trop consommateurs, mais préservera l'intégrité du système.

SORTSIZE

L'option **SORTSIZE** représente la mémoire nécessaire à la procédure Sort. Cette valeur est par défaut positionnée à **256 Mo**.

Si la valeur de **SORTSIZE** est insuffisante, le système SAS va créer des fichiers temporaires sur le disque dur (Cf *Graphique 2 - Statistiques I/O*). Du fait d'un nombre de lectures/écritures sur le disque plus important, le temps de traitement augmente également.

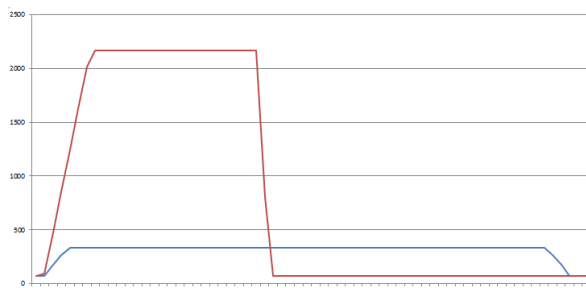
L'exemple ci-dessous illustre ces propos :

```
1043      proc sort data=work.mem_table;
1044      by actual ;
1045      run;
NOTE: There were 16058880 observations read from the data set
WORK.MEM TABLE.
```

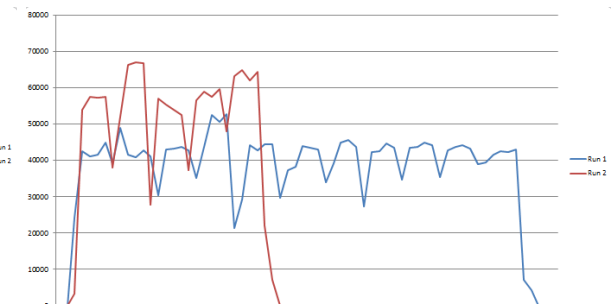
Nous avons effectué deux runs de tests. La valeur de l'option SORTSIZE a été modifiée entre le run 1 et le run 2.

	Sortsize	Real Time	User CPU time	System CPU time	Memory
Run 1	default	51.30 secondes	16.65 secondes	12.51 secondes	278 Mo
Run 2	2 Go	23.24 secondes	14.58 secondes	9.20 secondes	2 Go

Graphique 1 - Utilisation de la mémoire



Graphique 2 - Statistiques I/O



SAS ne va pas réserver ou allouer automatiquement la mémoire indiquée dans l'option MEMSIZE ou SORTSIZE. SAS n'utilise que la mémoire dont il a besoin pour terminer un processus. Par exemple, une étape DATA ne peut exiger que 20 Mo de mémoire. Donc, même si MEMSIZE est fixé à 1 Go, SAS n'utilise que 20 Mo de mémoire.

REALMEMSIZE

L'option REALMEMSIZE permet de spécifier la quantité de mémoire physique que le processus SAS peut s'allouer. De nombreuses procédures SAS utilisent l'option REALMEMSIZE afin de déterminer la quantité de mémoire à allouer et utiliser sans induire une trop grande consommation de Swap.

L'option système REALMEMSIZE fixe une limite supérieure à ne pas dépasser sur la mémoire virtuelle pour les procédures qui peuvent utiliser à la fois la mémoire physique et le Swap, telles que la PROC MEANS ou la PROC SORT.

Cette limite supérieure permet d'éviter de crash de la mémoire virtuelle (et du système d'exploitation).

L'option REALMEMSIZE ne doit jamais être supérieure à la quantité de mémoire réelle.

En revanche, l'option MEMSIZE place une limite de la quantité totale de mémoire virtuelle que le processus peut allouer dynamiquement à tout moment.

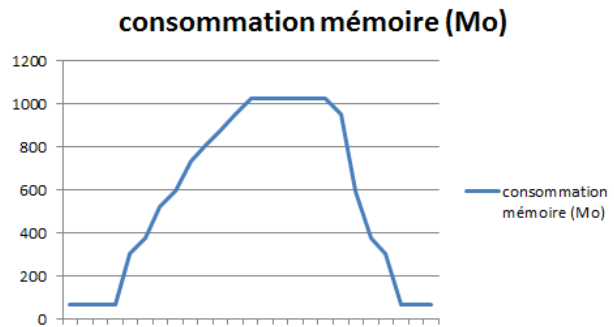
Le système d'exploitation commence la pagination lorsque la quantité de mémoire virtuelle nécessaire est supérieure à la mémoire réelle disponible. Pour éviter la pagination et les problèmes de performance associés, REALMEMSIZE et MEMSIZE doivent être réglés sur un sous-ensemble de la mémoire réelle.

Les deux exemples ci-dessous sont une illustration de ce mécanisme :

Le premier test consiste à positionner les options suivantes :

- memsize 10 Go
- sortsize 2 Go
- realmemsize 1 Go

Bien que la valeur de SORTSIZE soit définie à 2 Go, la consommation maximale du processus est de 1 Go.



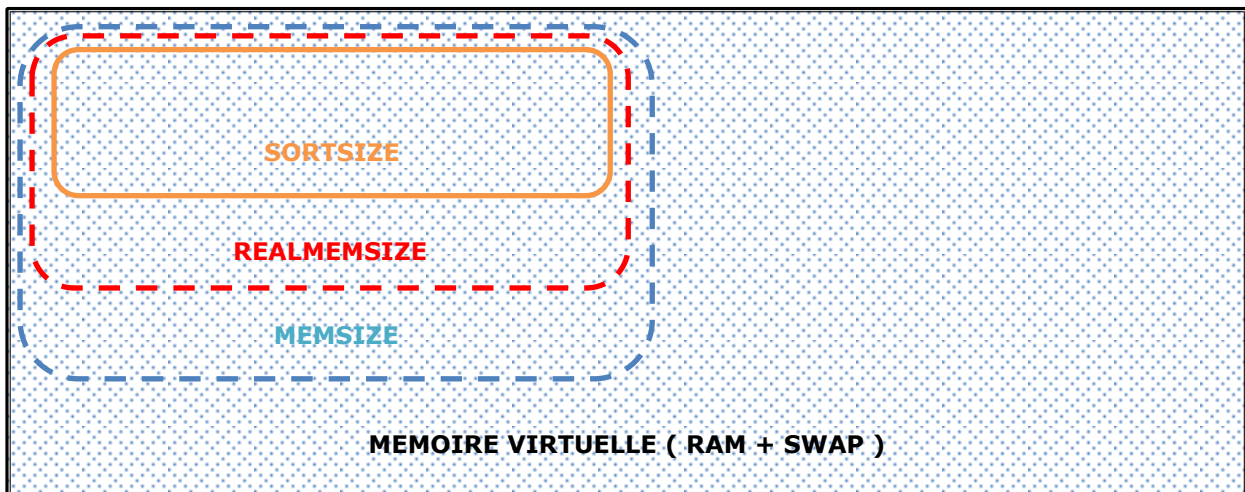
Dans un second test, et avec les options suivantes :

- memsize 512 Mo
- sortsize 1 Go
- realmemsize 1 Go

On constate que la mémoire utilisée ne dépasse pas 420 Mo.

```
NOTE: PROCEDURE SORT used (Total process time):
      real time           1:00.61
      user cpu time       31.36 secondes
      system cpu time    13.81 secondes
      memory              414094.98k
      OS Memory          420320.00k
```

En effet, la valeur de SORTSIZE sera au maximum égale à la valeur de l'option REALMEMSIZE, elle-même inférieure à la valeur de MEMSIZE. Il faut que MEMSIZE soit inférieur à la mémoire virtuelle totale du serveur (RAM + Swap)



Bonnes pratiques

Où positionner les options ?

Tout dépend de l'architecture, et de la nature des options (certaines sont déclarées dans le fichier de configuration, mais peuvent également l'être dans le fichier autoexec, ou dans un programme). Dans le cadre d'une utilisation de SAS via SAS Enterprise Guide, il est nécessaire de modifier le fichier de configuration du Workspace server, c'est-à-dire config/Lev1/SASApp/WorkspaceServer/sasv9_usermods.cfg.

```

* sasv9_usermods.cfg
*
* This config file extends options set in sasv9.cfg. Place your site-specific
* options in this file. Any options included in this file are common across
* all server components in this application server.
*
* Do NOT modify the sasv9.cfg file.
*
*/
-memsize MAX

```

Comment connaître la consommation mémoire d'une procédure ou d'une étape DATA ?

L'option Fullstimer permet d'afficher dans le journal, après chaque procédure, la mémoire consommée par le traitement. En cas d'interruption incorrecte du traitement, elle donne la taille mémoire maximum qui a pu être allouée.

Par défaut, si « `options Fullstimer;` » n'a pas été exécuté dans votre code SAS, l'affichage simple suivant apparaîtra dans le journal :

```

NOTE: L'étape DATA used (Total process time):
      real time          5:04.96
      cpu time           4:58.28

```

En positionnant l'option FULLSTIMER, le système SAS va collecter des statistiques liées aux performances pour chaque étape :

```

NOTE: L'étape DATA used (Total process time):
      real time          5:06.31
      user cpu time      4:16.81
      system cpu time    30.78 secondes
      memory             10051265.76k
      OS Memory          10057496.00k
      Timestamp          11/10/2012 11 h 59
      Page Faults                0
      Page Reclaims              0
      Page Swaps                 0
      Voluntary Context Switches 1800
      Involuntary Context Switches 2052
      Block Input Operations      0
      Block Output Operations     0

```

Comment connaître les valeurs de la session SAS en cours ?

Il faut exécuter le code ci-dessous depuis la session SAS adéquate (depuis un workspace server, un stored process server...) :

```

proc options group=memory ;
proc options group=performance ;run ;

```

Liens utiles

- [Usage Note 14125](#) : REALMEMSIZE vs. MEMSIZE settings on UNIX
- [Problem Note 44256](#) : In SAS® 9.3, setting MEMSIZE system option to values of 0 or MAX results in a value of 2G
- [How to Maintain Happy SAS Users](#)
- [Troubleshoot Your Performance Issues](#)

Conclusion

Comme nous l'avons vu dans cet article, plusieurs paramètres sont à prendre en compte au moment de spécifier les valeurs de chaque option.

Avant de définir les valeurs que vous souhaitez affecter aux différentes options, posez-vous les bonnes questions :

- Combien d'utilisateurs, ou plus précisément combien de processus SAS sont susceptibles de fonctionner simultanément ?
- Pour chaque processus, quelle est la consommation mémoire maximale nécessaire (reportée par l'option FULLSTIMER). En effet, la taille des données en entrée (tables SAS) influe sur la quantité de mémoire nécessaire, mais la nature des traitements et procédures utilisées joue aussi un rôle central.

Cela peut paraître simple mais dans la pratique trouver la bonne valeur est un exercice compliqué. J'ajouterais enfin que l'optimisation des performances systèmes n'est pas uniquement liée à la mémoire. En effet, il est important de prendre également en compte la gestion des I/O et les performances du processeur.

Nicolas Housset

Consultant Support Clients SAS France