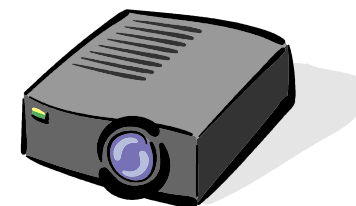


Modernisation et développement d'applications IBM i *Stratégies, technologies et outils*

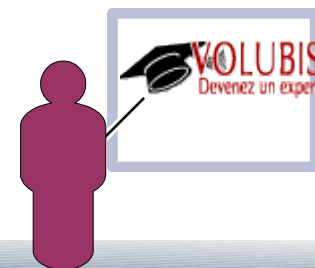
5 et 6 Avril 2012 – IBM Forum de Bois-Colombes



Volubis.fr

Conseil et formation sur OS/400, I5/OS puis IBM *i*
depuis 1994 !

Christian Massé - cmasse@volubis.fr



QZDASOINIT le serveur odbc/jdbc

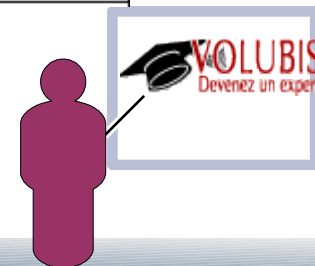
- QZDASOINIT est un job de type serveur, démarré par STRHOSTSVR
*DATABASE et répondant aux requêtes SQL entrantes (ODBC/JDBC))

Voyez la liste des travaux de type serveur ici

<http://publib.boulder.ibm.com/infocenter/iserics/v7r1m0/topic/rzaku/rzakuservertable.htm>

Server name	Job description	Subsystem	Job name	Shipped default value for autostart parameter
Database Server To Start: 1) Starts when the subsystem starts 2) If subsystem is active and the jobs are not active, issue STRPJ SBS (<i>subsystem name</i>) PGM(QSYS/QZDASOINIT), where <i>subsystem name</i> is QUSRWRK or the user-configured subsystem To End: Ends when the subsystem ends Product: 5761-SS1 Server Type: QIBM_OS400_QZBS_SVR_DATABASE	QGPL/QDFTSVR	QUSRWRK or configurable	QZDASOINIT	*YES

- Pour gagner du temps lors de la connexion, des travaux sont démarrés à l'avance, sur le principe du travail à démarrage anticipé.



QZDASOINIT le serveur odbc/jdbc

Sous-Systèmes TRAVAUX A DEMARRAGE ANTICIPE. (ADDPJE)

-Fonctionnent avec les entrées Télécom.
(Historiquement ADDCMNE, maintenant services IP)

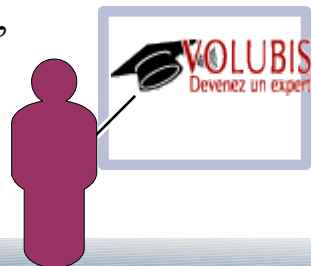
Permettent d'initialiser un travail sur le système cible (IBM i) avant l'activation du programme demandé par le système source (le client ou le serveur d'applications).

-Quand le système reçoit une demande de connexion réseau, il regarde s'il existe une entrée anticipée pour un pgm de MÊME NOM que celui demandé.

1/ si oui il utilise ce job

2/ si non un nouveau job est initialisé de manière traditionnelle.

-L'intérêt est le gain quant à la phase d'initialisation du travail et sa réutilisation, nous allons le voir.



QZDASOINIT le serveur odbc/jdbc

Ce travail va s'exécuter sous un profil donné jusqu'à la connexion.

A partir de cette demande, le profil de référence va devenir celui de la connexion (voir dans la JOBLOG, le profil en cours),

Cas de QZDASOINIT (serveur Database) :

Complément d'informations sur message

ID message : CPIAD02

Date d'envoi : 23/05/08 Heure d'envoi : 06:20:00

Message : User FORMATION1 from client 10.3.1.1 connected to server.

Cause : User profile FORMATION1 from client 10.3.1.1 is currently connected to this server job. The client name is either a TCP/IP remote system name, a dotted decimal IP address, or the local host name



Ajouter poste trav anticipés (ADDPJE)

Indiquez vos choix, puis appuyez sur ENTREE.

```
Description de sous-système . . SBSDB          #####
    Bibliothèque . . . . .          *LIBL
Programme . . . . . PGM             #####
    Bibliothèque . . . . .          *LIBL
Profil utilisateur . . . . . USER      QUSER
Démarrer les travaux . . . . . STRJOBS  *YES      (1)
Nombre initial de travaux . . . INLJOBS  3          (2)
Seuil . . . . . THRESHOLD           2          (3)
Nombre additionnel de travaux . ADLJOBS  2          (4)
Nombre maximal de travaux . . . MAXJOBS *NOMAX   (5)
```

F10

```
Travail . . . . . JOB                *PGM
Description de travail . . . . . JOBDB *USRPRF
Nombre maximal d'utilisations . MAXUSE  200      (6)
Attente de travail . . . . . WAIT     *YES      (7)
Identificateur du pool . . . . . POOLID  1
Classe:                               CLS
```



Ajouter poste trav anticipés (ADDPJE)

1/ les travaux doivent-ils démarrer en même temps que le sous-système ?

Il est possible de démarrer un travail à tout moment avec la commande **STRPJ** et de forcer l'arrêt par **ENDPJ**.

Arrêter travaux anticipés (ENDPJ)

Indiquez vos choix, puis appuyez sur ENTREE.

Sous-système	SBS	QUSRWRK
Programme	PGM	QZDASOINIT
Biblio		*LIBL
Type d'arrêt	OPTION	*CNTRLD

2/ si oui en (1), nombre de travaux à démarrer initialement

3/ seuil mini, qui va enclencher le démarrage de (4)

4/ nombre de travaux à démarrer à l'avance, quand le seuil est atteint.

5/ nombre maximum de travaux actifs en même temps.



Ajouter poste trav anticipés (ADDPJE)

6/ nombre maximal d'utilisations

un travail anticipé en attente d'un besoin passe à l'état PSRW

(vous ne les verrez sur le WRKACTJOB que si vous faites F14)

lors d'une demande entrante il devient actif, la connexion est notée dans la LOG (CPIAD02, comme vu plus haut)

lors de la demande de déconnexion :

-soit le job repasse à l'état PSRW

(il est complétement ré-initialisé, la JOBLOG est mise à blanc)

-soit le nombre de fois où il a été "recyclé" est atteint et il s'arrête

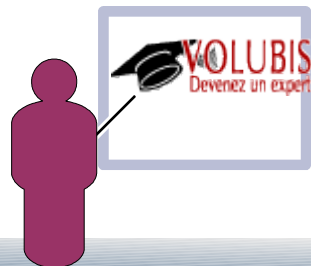
(c'est ce paramètre qui fixe le nombre de ré-utilisation possibles)

7/ si le nombre de jobs pouvant être actifs est atteint (MAXJOBS en 5/)

que se passe-t-il pour les demandes entrantes

*YES elles attendent qu'un job se libère

*NO la demande est rejetée



Vous pouvez avoir des statistiques d'utilisation par DSPACTPJ

```
Travaux anticipés actifs          AS400

Sous-système . . . . . : QUSRWRK      Date de réinit . . . : 17/11/10
Programme . . . . . : QZDASOINIT     Heure de réinit . . : 16:29:24
Bibliothèque . . . . : QSYS          Intervalle . . . . . : 0001:19:57
```

Travaux anticipés :

```
Nombre en cours . . . . . : 3
Moyenne . . . . . : 21,7
Maximum . . . . . : 52
```

Travaux anticipés en cours d'utilisation :

```
Nombre en cours . . . . . : 1
Moyenne . . . . . : 1,6
Maximum . . . . . : 50
```

A suivre...

F3=Exit F5=Réafficher F12=Annuler **F13=Réinitialiser**



Vous pouvez avoir des statistiques d'utilisation par DSPACTPJ

(écran suivant) Travaux anticipés actifs AS400

Sous-système : QUSRWRK Date de réinit : 17/11/10
Programme : QZDASOINIT Heure de réinit : 16:29:24
Bibliothèque : QSYS Intervalle : 0001:19:57

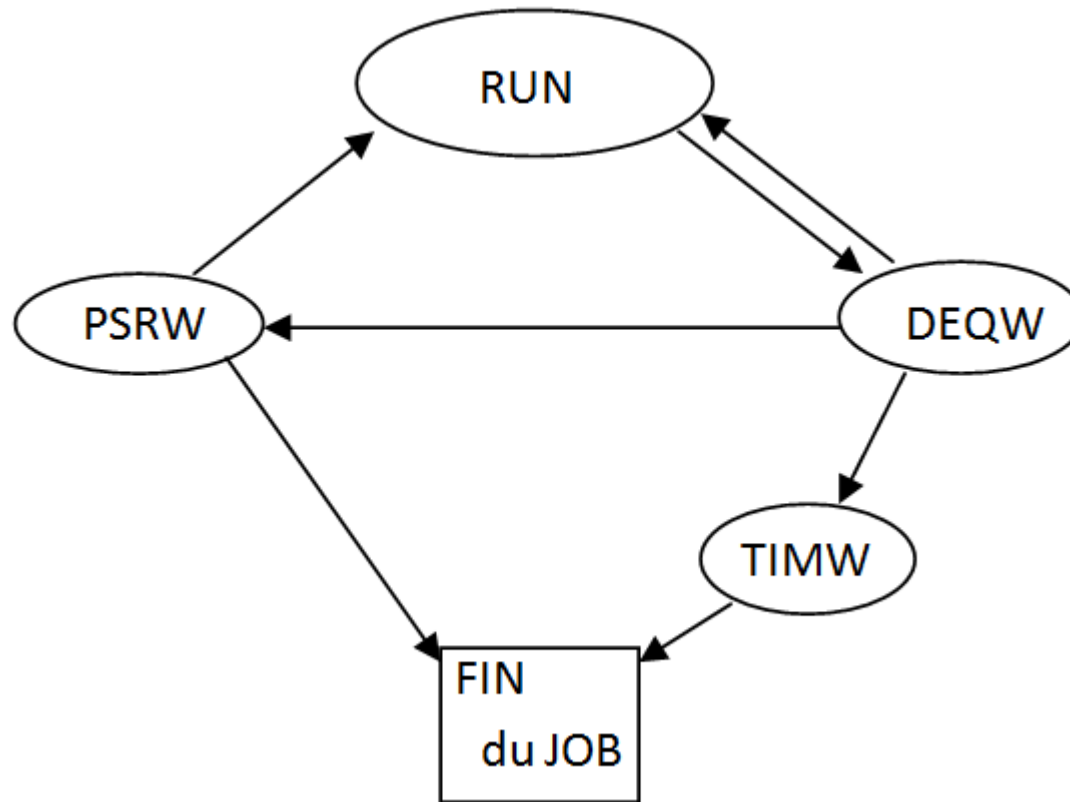
Demands de démarrage de programmes :

Nombre en cours d'attente : 0
Moyenne en attente : 0,0
Maximum en attente : 0
Temps d'attente moyen : 00:00:00,0
Nombre accepté : 5
Nombre refusé : 0

ici, vous avez des infos sur les éventuelles attentes et refus en fonction des paramètres MAXJOBS et WAIT.



cycle de vie d'un job QZDASOINIT



Cycle de vie d'un job QZDASOINIT

Lors d'un démarrage anticipé le job doit passer à l'état PSRW
(on ne les voit qu'avec F14 sur le WRKACTJOB)

à la connexion il passe à l'état RUN, puis exécute la première requête
entre deux requêtes, il est à l'état DEQW, puis repasse RUN à la requête suivante, etc...

Lors d'une demande de déconnexion :

- >si le nombre d'utilisations (200 par défaut pour QZDASOINIT) est atteint il se termine
- >sinon, il repasse à l'état PSRW (on ne le voit plus sur WRKACTJOB), en attente d'un prochain besoin.

En cas d'inactivité longue du client (fin du pgm sans demande de déconnexion, par exemple)
le serveur envoi une demande de réponse et démarre un "timer" (TIMW)

*Cette situation n'est pas normale, sans doute une déconnexion oubliée par le développeur
ou un bug du connecteur (.NET de client access 5.3 avait un tel bug).*

s'il ne reçoit pas de réponse au bout du temps prévu par le timer
(dépend du paramètre TCP/IP TCPKEEPALV, + 3 minutes), il se termine.



Pour travailler dans un Pool mémoire ou un sous-système dédié :

*facultatif, allouez de la mémoire à un Pool partageable : CHGSHRPOOL *SHRPOOLn SIZE(xxx)*

créez un sous système utilisant ce pool ou *BASE:

```
CRTSBSD MONSBS POOLS((1 *SHRPOOLn)) TEXT('Sous système dédié')
```

puis

```
ADDRTGE MONSBS SEQNBR(10) CMPVAL(*ANY) PGM(QCMD) CLS(QBATCH)
```

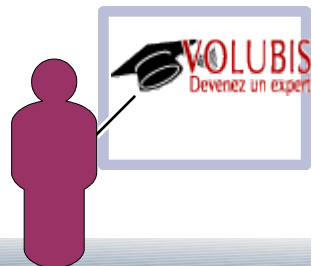
créez et ajoutez un JOBQ (ADDJOBQE) , si vous faites des tests en BATCH

ajoutez une entrée WorkStation (ADDWSE), si vous pensez travailler en 5250

Pour ODBC/JDBC (dont System i Navigator), suivez la procédure suivante :

Ajoutez un travail à démarrage anticipé à votre sous système, par :

```
ADDPJE SBSD(MONSBS) PGM(QSYS/QZDASOINIT) INLJOBS(?)  
JOB(Qgpl/QDFTSVR) CLS(QSYS/QPWFSEVER)
```



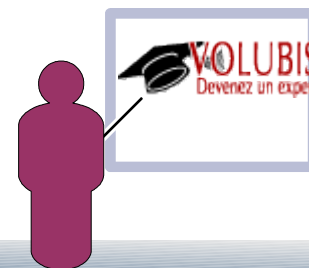
Pour travailler dans un Pool mémoire ou un sous-système dédié :

modifiez les propriétés du serveur Database via Iseries Navigator

Réseau
 Serveur
 System i Access

The screenshot shows the iSeries Navigator interface. On the left, a tree view shows the navigation path: Gestion centralisée (As400) > Mes connexions AS/400 > As400 > Réseau > Configuration TCP/IP > IPv4. The main pane displays a table of servers in the 'As400: iSeries Access' environment. The 'Base de données' server is selected, and a context menu is open over it, with the 'Propriétés' option highlighted.

Nom de serveur	Etat	Description
Central	Démarré	Central
Base de données	Démarré	Base de données
File d'attente de d		File d'attente de données
Fichier		Fichier
Impression réseau		Impression réseau
Commande à dista		Commande à distance
Ouverture de sess		Commande à distance
Programme de mappage de s...	Démarré	Programme de mappage de serveurs



Pour travailler dans un Pool mémoire ou un sous-système dédié :

Allez sur l'onglet « sous système »

Propriétés de Base de données - As400

Général Sous-systèmes

Indiquez les sous-systèmes par défaut des travaux de serveur qui permettent la configuration de sous-systèmes.

Utiliser les valeurs par défaut de serveur

Tous les clients

Sous-système : QUSRWRK

Autre action : Démarrage dans sous-syst en cours

Clients spécifiques

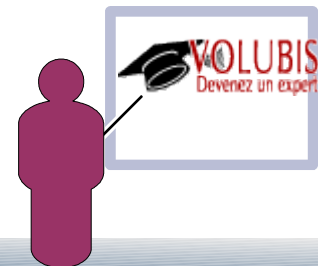
Client	Description	Masque d...	Sous-système	Autre action
10.3.1.15	Test de perf. dans ...	255.255....	FORMATION	Démarrage dans sous-s
<Public>	Tous les autres clients		QUSRWRK	Démarrage dans sous-s

Ajout

Edition

Retrait

OK Annuler Aide



Pour travailler dans un Pool mémoire ou un sous-système dédié :

Le bouton Ajout, permet d'indiquer votre sous système pour un ou plusieurs clients (adresse IP)

Edition de client

Informations sur le client

Description : Test de perf. dans FORMATION/FORMATION

Client :

Adresse IP : 10 . 3 . 1 . 15

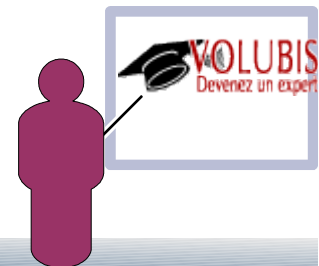
Plage d'adresses IP : . . . -- . . .

Masque de sous-réseau : 255 . 255 . 255 . 0

Sous-système : FORMATION

Autre action : Démarrage dans sous-syst en cours

OK Annulation Aide



QSQSRVR

Certaines requêtes utilisent un autre serveur : QSQSRVR

- Les applications utilisant CLI avec SQL_ATTR_SERVER_MODE
- Les accès JDBC natifs, dont
 - WAS
 - IBM Directory Server
 - Gestion centralisée
- Les accès PHP utilisant le connecteur DB2_connect()

L'application se connecte alors à un travail de type serveur

Ces travaux sont aussi des travaux à démarrage anticipés liés à QSYSWRK.

Par défaut, le système n'en démarre que 5.

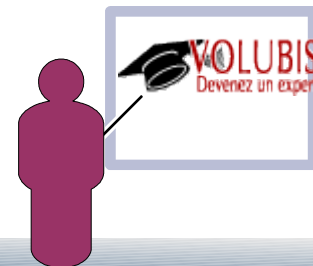


QSQSRVR

Certaines requêtes utilisent un autre serveur : QSQSRVR

- Les PTF SI33949 (V6) et SI33298 (V5R40), ainsi que la version 7, proposent une variable d'environnement QIBM_SRVRMODE_SBS qui fixé à '*SAME' demande à ce que le travail QSQSRVR soit démarré dans le **même** sous-système que le travail « client »
- Ex : `ADDENVVAR ENVVAR(QIBM_SRVRMODE_SBS) VALUE('*SAME') LEVEL(*SYS)`
- Si le sous-système ne possède pas de poste de travail à démarrage anticipé pour QSQSRVR, un travail QSQSRVR est démarré de type BCI et arrêté en fin de traitement
- si le sous-système possède un travail à démarrage anticipé pour QSQSRVR (défini par `ADDPJE PGM(QSYS/QSQSRVR)`) c'est ce dernier qui est (ré) utilisé.

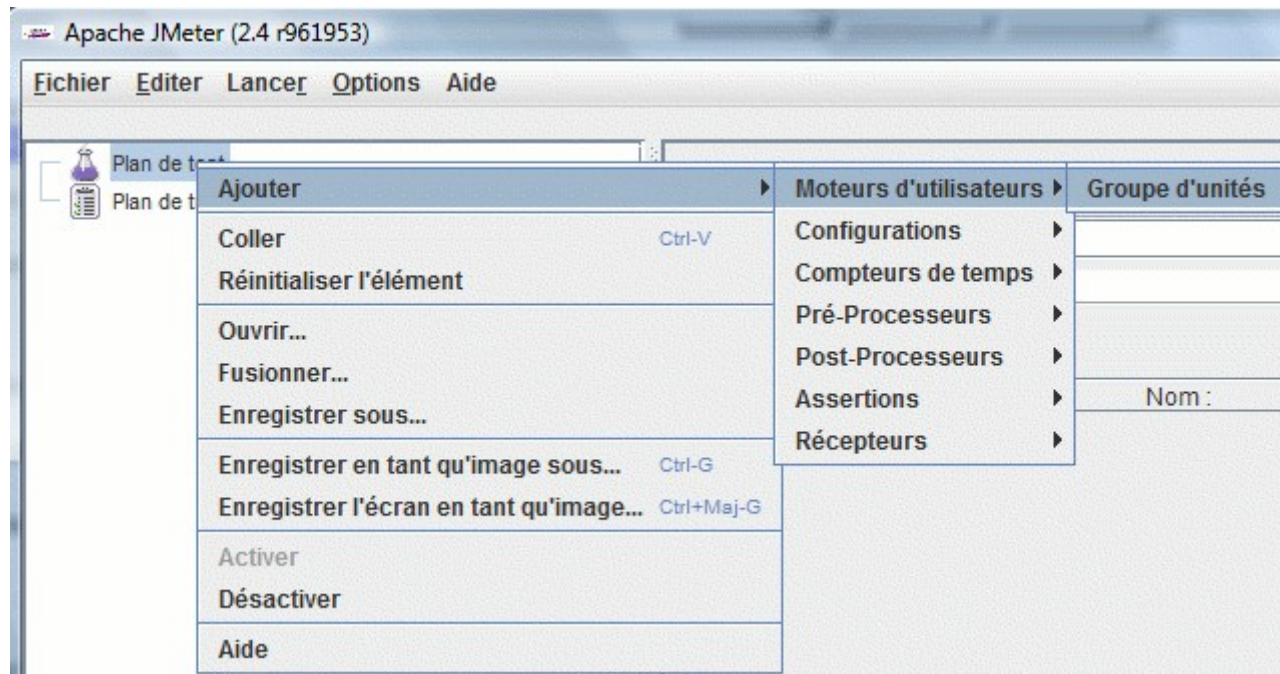
NB : QIBM_SRVRMODE_SBS n'est pas utilisée par QHTTPSVR et ZENDSVR



Pour tester

Pour tester tout cela (et faire des tests de montée en charge), voyez le projet Apache Jmeter

dézippez et lancez jmeter.bat



Ajoutez un groupe d'unité (de thread)



Pour tester

- Indiquez un nombre d'utilisateurs,
- un objectif de durée (peut ne pas être respecté suivant la puissance votre poste)
- un nombre de boucles

Groupe d'unités

Nom :

Commentaires :

Action à suivre après une erreur d'échantillon

Continuer Arrêter l'unité Arrêter le test Arrêter le test immédiatement

Propriétés du groupe d'unités

Nombre d'unités (utilisateurs) :

Durée de montée en charge (en secondes) :

Nombre d'itérations : Infini

Programmeur de démarrage



Pour tester

Ensuite, ajoutez dans Configurations, une « Configuration de connexion JDBC »

indiquez :

Un nom de liaison à utiliser plus tard

une requête SQL de validation
de la connexion

(values 1, retourne 1, et est très rapide)

L'url d'accès :

jdbc:as400://VOTRE-AS400

les coordonnées de la classe JDBC :

com.ibm.as400.access.AS400JDBCdriver

un identifiant et mot de passe

Configuration de connexion JDBC

Nom :

Commentaires :

Nom de liaison du pool

Nom de liaison :

Configuration du pool de connexions

Nombre maximum de connexions :

Expiration du pool (ms) :

Intervalle de nettoyage des connexions disponibles (ms) :

Validation automatique (auto commit) :

Validation des connexions par le pool

Connexions persistantes (keep-alive) :

Durée de vie maximum d'une connexion (ms) :

Requête de validation :

Configuration de connexion à la base de données

URL de la base de données :

Classe du pilote JDBC :

Identifiant :

Mot de passe :



Pour tester

- Il faut ensuite retourner sur le plan de test et ajouter le chemin de jt400.jar au CLASSPATH de Jmeter.

Lancer les groupes d'unités en série (c'est-à-dire : lance un groupe à la fois)

Mode de test fonctionnel

Sélectionner le mode de test fonctionnel uniquement si vous avez besoin d'enregistrer les données reçues du serveur dans un fichier à chaque requête.

Sélectionner cette option affecte considérablement les performances.

Ajouter un répertoire ou un fichier 'jar' au 'classpath'

Librairie
C:\Program Files\jtopen\lib\jt400.jar

Téléchargez le, s'il faut depuis <http://jt400.sourceforge.net/>



Pour tester

·Ajoutez ensuite une ou plusieurs requêtes SQL (Ajouter/Échantillons/requête JDBC)

Requête JDBC

Nom :

Commentaires :

Nom de liaison avec le pool

Nom de liaison :

Requête SQL

Type de requête :

Requête :

Remarquez le nom de liaison, correspondant à l'ID donné lors de la configuration de la connexion JDBC



Pour tester

Ajoutez enfin un récepteur pour voir les résultats de type *Graphique de résultats* ou *Tableau de résultats* (c'est le cas ci-dessous) puis lancez (Ctrl + R)

test JDBC 400.jmx (G:\jakarta-jmeter-2.4\bin\test JDBC 400.jmx) - Apache JMeter (2.4 r961953)

Fichier Editer Lancer Options Aide

50 / 50

Plan de test

- test JDBC
 - JDBC AS400
 - appel_13
 - appel_114
 - Tableau de résultats
- Plan de travail

Tableau de résultats

Nom : Tableau de résultats

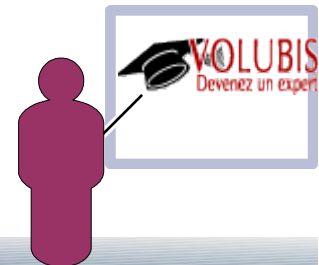
Commentaires :

Ecrire les données dans un fichier

Nom du fichier : Parcourir... Uniquement : Erreurs Succès Configurer

Echantillon #	Heure début	Nom d'unité	Libellé	Temps (ms)	Statut	Ocets
1	10:22:33.835	test JDBC 1-2	appel_13	13423	▲	97925
2	10:22:33.936	test JDBC 1-3	appel_13	13404	▲	97925
3	10:22:33.751	test JDBC 1-1	appel_13	13616	▲	97925
4	10:22:35.001	test JDBC 1-13	appel_13	12336	▲	97925
5	10:22:34.039	test JDBC 1-4	appel_13	13734	▲	97925
6	10:22:38.824	test JDBC 1-49	appel_13	30487	▲	97925
7	10:22:37.596	test JDBC 1-39	appel_13	53310	▲	97925
8	10:22:37.969	test JDBC 1-40	appel_13	52939	▲	97925
9	10:22:37.595	test JDBC 1-37	appel_13	53313	▲	97925
10	10:22:38.175	test JDBC 1-44	appel_13	52732	▲	97925
11	10:22:38.065	test JDBC 1-43	appel_13	52843	▲	97925
12	10:22:37.739	test JDBC 1-38	appel_13	54425	▲	97925
13	10:22:38.066	test JDBC 1-42	appel_13	54100	▲	97925
14	10:22:38.254	test JDBC 1-45	appel_13	53949	▲	97925
15	10:22:38.742	test JDBC 1-50	appel_13	53462	▲	97925
16	10:22:38.064	test JDBC 1-41	appel_13	54155	▲	97925
17	10:22:38.312	test JDBC 1-46	appel_13	53907	▲	97925
18	10:22:38.491	test JDBC 1-48	appel_13	53728	▲	97925
19	10:22:34.340	test JDBC 1-7	appel_13	57880	▲	97925
20	10:22:34.136	test JDBC 1-5	appel_13	58084	▲	97925
21	10:22:38.412	test JDBC 1-47	appel_13	53809	▲	97925
22	10:22:34.237	test JDBC 1-6	appel_13	57985	▲	97925
23	10:22:34.638	test JDBC 1-10	appel_13	57584	▲	97925
24	10:22:34.438	test JDBC 1-8	appel_13	57784	▲	97925
25	10:22:34.541	test JDBC 1-9	appel_13	57684	▲	97925
26	10:22:34.739	test JDBC 1-11	appel_13	57486	▲	97925
27	10:22:34.881	test JDBC 1-12	appel_13	57345	▲	97925

Nombre d'échantillons 33 Dernier échantillon 5610 Moyenne 44805 Ecart type 18961



Pour tester

Les requêtes s'exécutent, voyons le résultat du coté IBM i, par

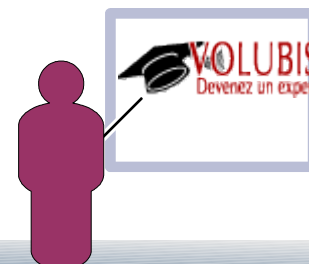
DSPACTPJ QUSRWRK QZDASOINIT :

```
AS400.ws - PCCM
Fichier Edition Vue Communication Actions Fenêtre Aide
Hôte: as400.volubis.intra Port: 23 ID poste de travail: PCCM Déconnexion
Travaux anticipés actifs AS400
18/11/10 10:27:45
Sous-système . . . . . : QUSRWRK Date de réinit . . . . . : 17/11/10
Programme . . . . . : QZDASOINIT Heure de réinit . . . . . : 18:08:57
Bibliothèque . . . . . : QSYS Intervalle . . . . . : 0016:18:48

Travaux anticipés :
Nombre en cours . . . . . : 52
Moyenne . . . . . : 2,3
Maximum . . . . . : 52

Travaux anticipés en cours d'utilisation :
Nombre en cours . . . . . : 0
Moyenne . . . . . : 0,2
Maximum . . . . . : 50

Appuyez sur ENTREE pour continuer.
A suivre...
F3=Exit F5=Réafficher F12=Annuler F13=Réinitialiser
MÂ A MW 01/001
```



Quelles sont nos marges de manœuvres.

Nous ne parlerons pas de stratégies d'indexation, la session 12 le fera.

Voyez si vous devez mettre en place l'expert cache
(sur un pool mémoire dédié c'est largement souhaitable) ==> CHGSHRPOOL PAGING(*CALC)

ce paramètre permet de garder en mémoire les tables les plus fréquemment utilisées

Voyez l'utilisation des MQT (Materialize Table Query)

Il s'agit de créer un table contenant physiquement le résultat d'une requête :

```
CREATE SUMMARY TABLE -(nom)---AS (select SQL complet [ORDER BY admis])->
```

```
!--DATA INITIALLY IMMEDIATE---!  
>---!                               !--REFRESH DEFERRED-->  
!--DATA INITIALLY DEFERRED ---!
```

```
!---ENABLED QUERY OPTIMIZATION--!  
>--- MAINTAINED BY USER--!         !---.   
!---DISABLE QUERY OPTIMIZATION--!
```



Quelles sont nos marges de manœuvres.

MQT, Attention, seul SQE sait utiliser les tables matérialisées,

il y a des restrictions (SOUNDEX, DIFFERENCE, DECRYPT_xx, ...)

L'utilisation des tables matérialisée est une option, ce n'est pas le standard, il faut le demander en plaçant dans QAQQINI (voir ci-dessous) :

```
MATERIALIZED_QUERY_TABLE_USAGE = *ALL  
MATERIALIZED_QUERY_TABLE_REFRESH_AGE = *ANY  
(ou une durée sous la forme d'un timestamp AAAAMMJJHHMMSS)
```

il faut d'autre part que la TABLE a été crée avec ENABLED QUERY OPTIMIZATION et que le paramètre ALWCPYDTA soit à *OPTIMIZE (par défaut sous ODBC/JDBC)

Toute requête utilisant un sous ensembles des données contenues dans la MQT et ceci, dans un laps de temps inférieur à celui indiqué dans QAQQINI, lira directement les données dans celle-ci.

Attention, le rafraîchissement de la table matérialisée est à votre charge (ordre REFRESH)



Quelles sont nos marges de manœuvres.

Vérifiez l'équilibrage disque

WRKDSKSTS (% d'occupation ET % d'utilisation)

Si certains disques sont plus remplis que d'autres (suite à achat de disques)

STRASPBAL *CAPACITY

Si certains disques sont plus utilisés que d'autres (achat de disque *suivi* de l'installation de logiciel)

TRCASPBAL *ON

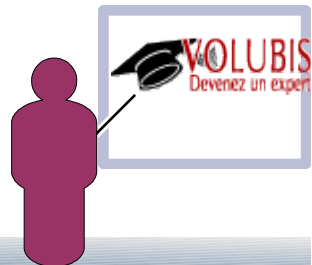
(analyse de l'activité base de données sur une plage horaire significative)

TRCASPBAL *OFF

Puis

STRASPBAL *USAGE

Le système réparti sur l'ensemble de l'ASP un « panachage » de fichiers très utilisés et de fichiers peu utilisés



Quelles sont nos marges de manœuvres.

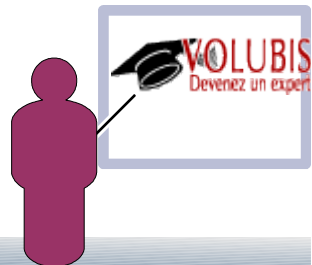
- Installez la V7 : en fonction des statistiques d'utilisation des fichiers (GCS) et de deux nouvelles zones RANDOM_READS et SEQUENTIAL_READS dans les fichiers statistiques SYSPARTITIONSTAT et SYSINDEXSTAT de QSYS2, les fichiers les plus fréquemment utilisés seront placés automatiquement sur disque SSD (si vous en possédez)

ce paramètre peut être forcé lors de la création (paramètre UNIT)

- Un fichier pouvait déjà être chargé "manuellement" en mémoire par la commande SETOBJACC.

vous pouvez maintenant le déclarer comme devant l'être automatiquement à l'aide du paramètre KEEPINMEM(*YES) des commandes CHGPF / CHGLF

le pool mémoire dans lequel garder ce fichier est précisé par l'option de QAQQINI : MEMORY_POOL_PREFERENCE



Rapports sur QAQQINI

il est possible de modifier les attributs des requêtes à venir dans un job par la commande CHGQRYA et l'utilisation du fichier QAQQINI

ce fichier est fourni dans la bibliothèque QSYS, mais vous devez dupliquer ce fichier dans votre bibliothèque.

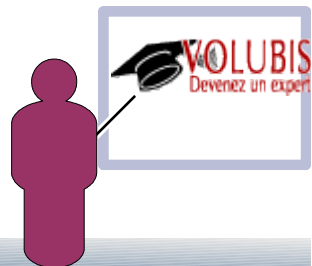
```
CRTDUPOBJ OBJ(QAQQINI) FROMLIB(QSYS) OBJTYPE(*FILE) TOLIB(ma-bib)
DATA(*YES) [il faut dupliquer (pas de CPYF) car il y a un trigger d'associé]
```

puis vous indiquez la version du fichier à utiliser par CHGQRYA

```
CHGQRYA QRYOPLIB(ma-bib) QUSRSYS par défaut.
```

ce fichier est structuré comme suit :

QQPARM CHAR(256)	le paramètre à modifier
QQVAL CHAR(256)	la valeur à attribuer
QQTEXT CHAR(1000)	commentaire



QAQQINI

exemple d'utilisation

```
Update QGPL/QAQQINI set QQVAL = '*YES'
```

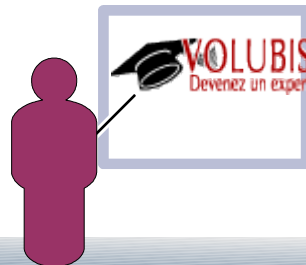
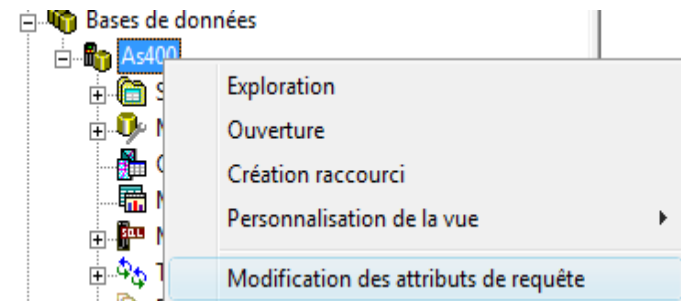
```
where QQPARM = 'MESSAGES_DEBUG'
```

System i navigator

vous offre une interface graphique de gestion de ces paramètres

1/ choisissez l'option base de données, votre « rdb »

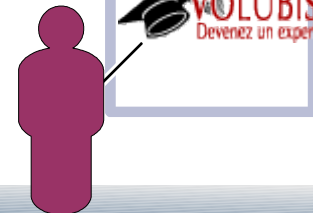
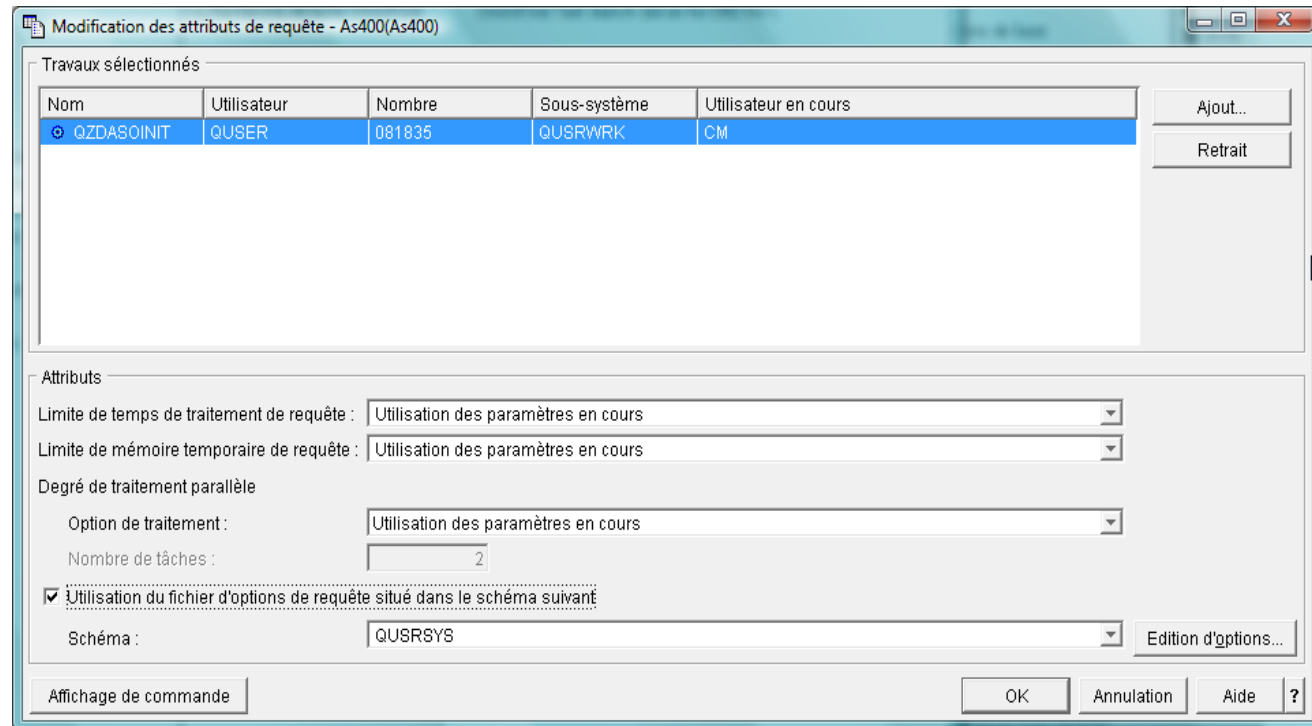
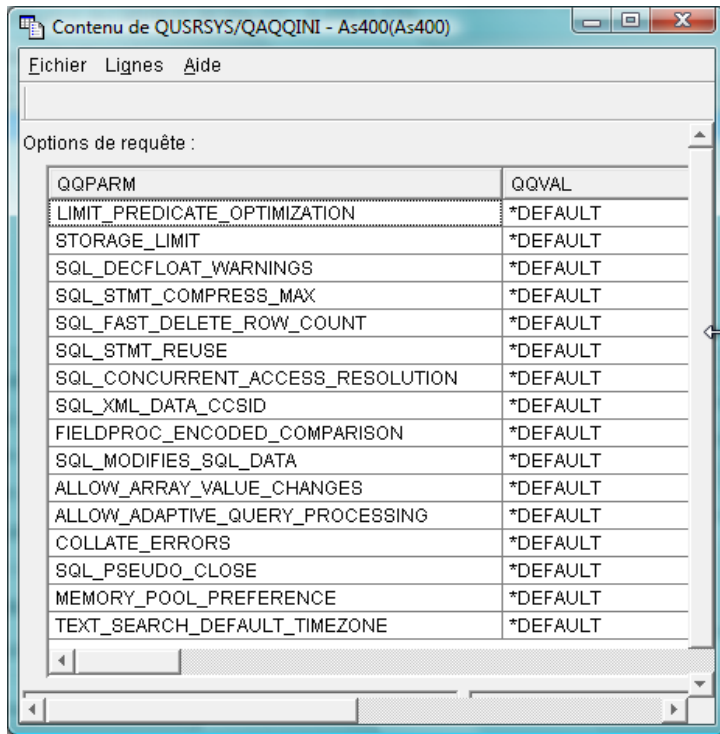
2/ cliquez avec le bouton droit, puis "modification attributs de requêtes"



QAQQINI

3/ indiquez

- le/les job(s) concerné(s) (Ajout, permet d'avoir la liste complète)
- la bibliothèque où se trouve le fichier QAQQINI
- « Edition d'options » vous affiche le contenu du fichier



QAQQINI

Vous remarquerez deux paramètres de la commande CHGQRYA bien pratiques

QRYTIMLMT, permettant de rejeter toute requête dépassant cette durée

Modification des attributs de requête - As400(As400)

Travaux sélectionnés

Nom	Utilisateur	Nombre	Sous-système	Utilisateur en cours
QZDASOINIT	QUSER	081835	QUSRWRK	CM

Attributs

Limite de temps de traitement de requête : Utilisation des paramètres en cours

Limite de mémoire temporaire de requête : Utilisation des paramètres en cours

Degré de traitement parallèle

Option de traitement : Utilisation des paramètres en cours

Nombre de tâches : 2

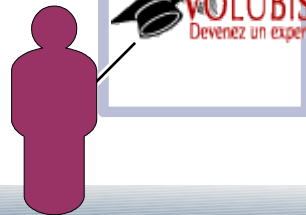
Utilisation du fichier d'options de requête situé dans le schéma suivant

Schéma : QUSRSYS

Affichage de commande

OK Annulation Aide ?

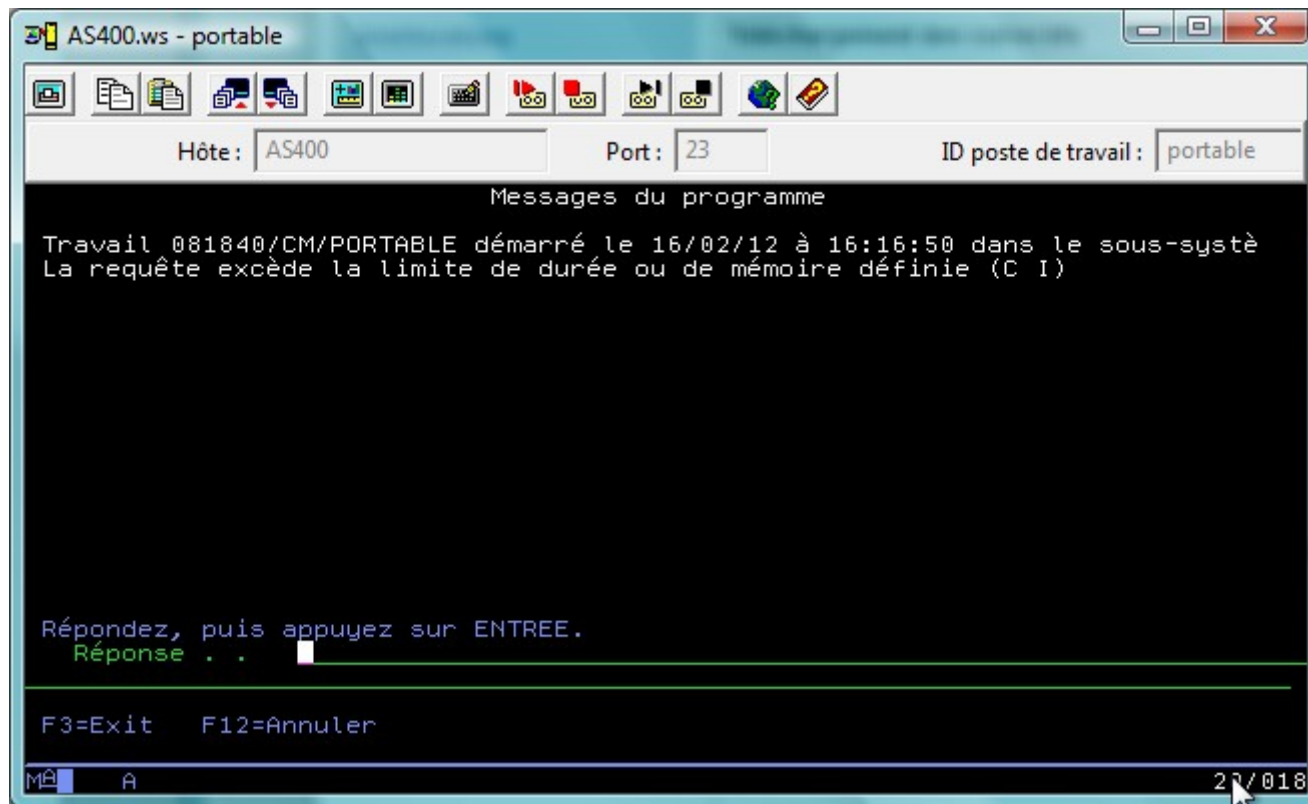
QRYSTGLMT, permettant de rejeter toute requête utilisant (en Mo) plus d'espace mémoire qu'indiqué.



QAQQINI

Vous remarquerez deux paramètres de la commande CHGQRYA bien pratiques

Ces paramètres provoquent l'apparition d'un message d'interrogation (CPA4259)



The screenshot shows a terminal window titled "AS400.ws - portable". The window has a menu bar with various icons. Below the menu bar, there are three input fields: "Hôte: AS400", "Port: 23", and "ID poste de travail: portable". The main area of the terminal is black with white text. It displays the following message:

```
Messages du programme  
Travail 081840/CM/PORTABLE démarré le 16/02/12 à 16:16:50 dans le sous-système  
La requête excède la limite de durée ou de mémoire définie (C I)  
  
Répondez, puis appuyez sur ENTREE.  
Réponse . . .  
  
F3=Exit F12=Annuler
```

At the bottom right of the terminal window, the text "27/018" is visible.

Ici, sur une session 5250



QAQQINI

Vous remarquerez deux paramètres de la commande CHGQRYA bien pratiques

Ce message peut maintenant être intercepté par le point d'exit QIBM_QQQ_QUERY_GOVR Permettant une réponse plus fine, fournie par programme

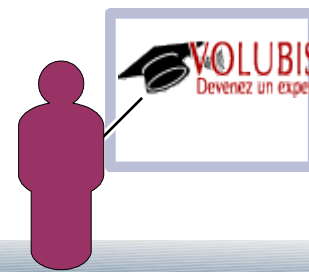
```
AS400.ws - portable
Hôte: AS400      Port: 23      ID poste de travail : portable

Work with Registration Information
Type options, press Enter.
5=Display exit point      8=Work with exit programs

  Opt  Exit Point      Exit Point Format  Registered  Text
  ---  -
  --  QIBM_QOK_NOTIFY  VRFY0100          *YES        System Directory Notify Exit
  --  QIBM_QOK_SUPPLIER  SUPL0100          *YES        System Directory Supplier Exi
  --  QIBM_QOK_VERIFY    VRFY0100          *YES        System Directory Verify Exit
  --  QIBM_QPA_DEVSEL    PADS0100          *YES        Sélection d'unité virtuelle
  --  QIBM_QPMW_ARM4_ADAP  ADAP0100          *YES        Programme d'exit d'adaptateur
  --  QIBM_QPQ_TRANSFORM  XPDF0100          *YES        Point d'exit de conversion IP
  --  QIBM_QPWFS_FILE_SERV  PWFS0100          *YES        Serveur de fichiers
  --  QIBM_QP0L_SCAN_CLOSE  SCCL0100          *YES        Scannage système fichiers int
  --  QIBM_QP0L_SCAN_OPEN  SCOP0100          *YES        Scannage système fichiers int
  --  QIBM_QQQ_QUERY_GOVR  QRYG0100          *YES        QUERY GOVERNOR
  --  QIBM_QRQ_SQL        RSQL0100          *YES        Original Remote SQL Server
                                     A suivre...

Command
===>
F3=Exit  F4=Prompt  F9=Retrieve  F12=Cancel

MA  A  18/003
```



Résultat de WRKREGINF

QAQQINI

Vous remarquerez deux paramètres de la commande CHGQRYA bien pratiques

Enfin le fait que la requête soit refusée peut constituer un critère pour STRDBMON

Assistant de création de moniteur de performances SQL - As400(As400)

Pour limiter le volume de données collectées, indiquez les filtres à utiliser. Lorsque des filtres sont fournis, seules les correspondant aux valeurs de filtre indiquées sont capturées.

Si vous voulez limiter le volume de données collectées, indiquez les filtres à utiliser :

- Nombre initial d'enregistrements : 0
- Durée d'exécution de requête minimale estimée : 0
- Mémoire temporaire minimale estimée : 0
- Nom du travail : QZDASOINIT
- Utilisateur du travail : QUSER
- Utilisateur en cours : CM
- Emplacement du client : 10.3.1.91
- Port local :
- Limites de la fonction de régulation de requêtes : Collecte systématique des informations en cas de dép...

Aide ? < Précédent Suivant > Terminer Annuler

QAQQINI

Une alternative à la création d'un fichier QAQQINI se trouve proposée par la procédure **override_qaqqini** ; nouvelle en V6R10.

Attention, l'utilisation de cette procédure implique les droits *JOBCTL pour l'utilisateur

-- créé un fichier QAQQINI dans QTEMP

```
Call override_qaqqini('1', ' ', ' ')
```

-- modifie la valeur pour le job

```
Call override_qaqqini('2', 'SQL_PSEUDO_CLOSE ', '05')
```

-- détruit le fichier de QTEMP, retour à la normale

```
Call override_qaqqini('3', ' ', ' ')
```



QAQQINI

Pour finir, parlons de deux paramètres.

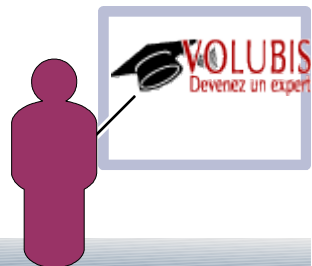
Qu'est-ce qui fait que mes fichiers **restent ouverts** ?

Quand vous passez une requête SQL, le moteur ouvre le fichier, exécute la requête puis ferme le fichier.

Si vous lancez la même requête une *n^{ème}* fois, le moteur se dit que vous aller sans doute l'exécuter de nombreuses fois et décide de ne plus fermer les fichiers concernés (jusqu'à fin du job). On parle de « pseudo close ».

N était jusqu'à maintenant égal à 2 et non négociable.

La V6R1 amène ce nouveau paramètre **SQL_PSEUDO_CLOSE** dans QAQQINI pour rendre paramétrable la valeur de *n*, donc le nombre de fois avant que les fichiers restent ouverts.



QAQQINI

La version 6 avait apporté à l'ordre SELECT l'option SKIP LOCKED DATA

cela permettait lors d'une lecture avec verrouillage d'enregistrement, c.a.d COMMIT à CS(*CS) ou RS(*ALL), d'ignorer les lignes verrouillées

Cette option a été étendue aux instructions UPDATE et DELETE

vous avez, en plus, le choix entre trois comportements :

WAIT FOR OUTCOME

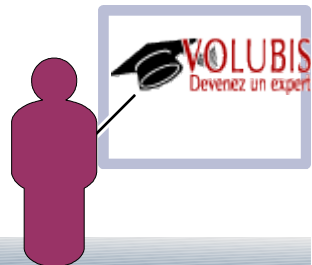
Attendre que les lignes soient libérées (COMMIT par exemple)
cela n'a pas d'effet sur les niveaux de COMMIT inférieurs à CS

USE CURRENTLY COMMITTED

Utiliser les valeurs déjà validées
cela ne peut s'appliquer qu'aux COMMIT niveau CS (sans KEEP LOCKS)

SKIP LOCKED DATA

les lignes verrouillées sont ignorées. Comme cette clause est maintenant valide avec UPDATE/DELETE, elle peut être utilisée avec tous les niveaux de COMMIT, sauf RR (repeatable read)



QAQQINI

Vous pouvez en plus préciser cette option :

1/ en fin des ordres, SELECT, UPDATE, DELETE, PREPARE

2/ sur le nouveau paramètre CONACC des commandes de compilation (CRTSQLRPGI par ex.) et RUNSQLSTM (pas STRSQL)

3/ sur l'ordre CREATE PROCEDURE | CREATE FUNCTION (ou ALTER)

4/ avec SET OPTION CONACC=*CURCMT | *WAIT | *DFT dans la procédure SQL

5/ dans QAQQINI avec l'option SQL_CONCURRENT_ACCESS_RESOLUTION

