



Administration

SQL Server

2012

SQLcmd



SQLcmd

- Ouverture des ports pour un accès à distance avec déclaration des utilisateurs connectés qui exécutent les commandes
 - Pour se connecter à distance il faut installer un client SQL Management studio sur les poste
 - ouvrir les ports 1433 et 1434 pour les serveurs SQL
 - Déclarer les utilisateurs AD et windows dans la base SQLServer distante



Outil SQLcmd

- SQLCmd pour saisir des commandes en ligne ou exécuter des scripts en ligne de commande
- Exemple
 - set BACKUPDIR_LOG=C:\Backuplog
 - set BACKUPDIR=C:\Backup
 - set SERVERNAME=WIN-J1TUGT2M59F
 - set INSTANCENAME=FORM
 - set DATABASE_NAME=ALL_DATABASES
 - set DDATE=%madate%_%mytime%
 - sqlcmd -E -S %SERVERNAME%\%INSTANCENAME% -d master -Q "EXECUTE dbo.DatabaseIntegrityCheck @Databases = '%DATABASE_NAME%', @CheckCommands = 'CHECKDB' -b -o %BACKUPDIR_LOG%\DatabaseIntegrityCheck_%DATABASE_NAME%_%DDATE%.log 2>&1 ;"



Outil SQLcmd

- Options de SQLcmd (suite)
 - -A : Se connecte à SQL Server avec une connexion administrateur dédiée (Dedicated Administrator Connection). Ce type de connexion est utilisé pour dépanner un serveur.
 - Elle ne fonctionne qu'avec les serveurs prenant en charge DAC ([Connexion de diagnostic pour les administrateurs de base de donnée](#))
 - Sinon SQLcmd setermine
 - **ddb_name** : Exécute une instruction USE *db_name* lors du démarrage de **sqlcmd**. Cette option définit la variable de script **sqlcmdSQLCMDDBNAME**. Celle-ci spécifie la base de données initiale.



Outil SQLcmd

- Options de SQLcmd (suite)
 - -E : Utilise une connexion approuvée au lieu d'un nom d'utilisateur et un mot de passe pour vous connecter à SQL
 - -H*workstation_name* : Cette option définit la variable de script **sqlcmd** SLCMDWORKSTATION. Le nom de la station de travail est indiqué dans la colonne **hostname** de l'affichage catalogue **sys.processes** et peut être retourné à l'aide de la procédure stockée **sp_who**.
Si cette option n'est pas spécifiée, le nom de l'ordinateur actif est utilisé par défaut



Outil SQLcmd

- Options de SQLcmd
 - -i : permet d'exécuter un fichier
 - -S : permet de préciser le nom du serveur et de l'instance
 - -S %SERVERNAME% \ %INSTANCENAME%
 - -o*output_file* : permet de gérer un fichier de log
 - -Q : permet d'exécuter une requête
 - Q " select * from avion "
 - -Q "DBCC SHRINKDATABASE (N' %DATABASE_NAME%', 10, TRUNCATEONLY) "



Outil SQLcmd

- Exemple «

- Réorganisation des index

- `sqlcmd -E -S %SERVERNAME%\%INSTANCENAME% -d master -Q "EXECUTE dbo.IndexOptimize @Databases = '%DATABASE_NAME%', @FragmentationLow = NULL, @FragmentationMedium = 'INDEX_REORGANIZE,INDEX_REBUILD_ONLINE,INDEX_REBUILD_OFFLINE', @FragmentationHigh = 'INDEX_REBUILD_ONLINE,INDEX_REBUILD_OFFLINE', @FragmentationLevel1 = 5, @FragmentationLevel2 = 30, @TimeLimit = 3600" -b -o %BACKUPDIR_LOG%\Reorg_Rebuild_Index_%DATABASE_NAME%_%DDATE%.log 2>&1`



Outil SQLcmd

- Exemple 2

- Shrink Database

- sqlcmd -E -S %SERVERNAME%\%INSTANCENAME% -d
master -Q "DBCC
SHRINKDATABASE(N'%DATABASE_NAME%', 10,
TRUNCATEONLY)" -b -o
%BACKUPDIR_LOG%\shrink_database_%DATABASE_NAME%
_%DDATE%.log 2>&1

- Backup des journaux

- sqlcmd -E -S %SERVERNAME%\%INSTANCENAME% -d
master -Q "EXECUTE dbo.DatabaseBackup
@Databases = '%DATABASE_NAME%', @Directory =
'%BACKUPDIR%', @BackupType = 'LOG', @Compress
= 'Y', @CheckSum = 'Y', @Verify = 'Y',
@ChangeBackupType = 'Y'" -b -o
%BACKUPDIR_LOG%\backup_log_%DATABASE_NAME%_%DDA
TE%.log 2>&1