

Administration de Bases de Données

Sauvegarde et restauration

SKLAB Youcef

May 20, 2017

Sauvegarde et restauration

- 1 Sauvegarde et restauration
 - La Sauvegarde
 - La Restauration

Sauvegarde et restauration

Sauvegarde : pourquoi, comment ?

- Crash du disque dur
- Une mauvaise manipulation, qui écrase des dizaines de méga-octets de données.
- Plusieurs possibilités d'erreurs de différentes nature....

Pour éviter que tout finisse mal **SAUVEGARDEZ VOS DONNÉES !!!!**

Quoi sauvegarder ?

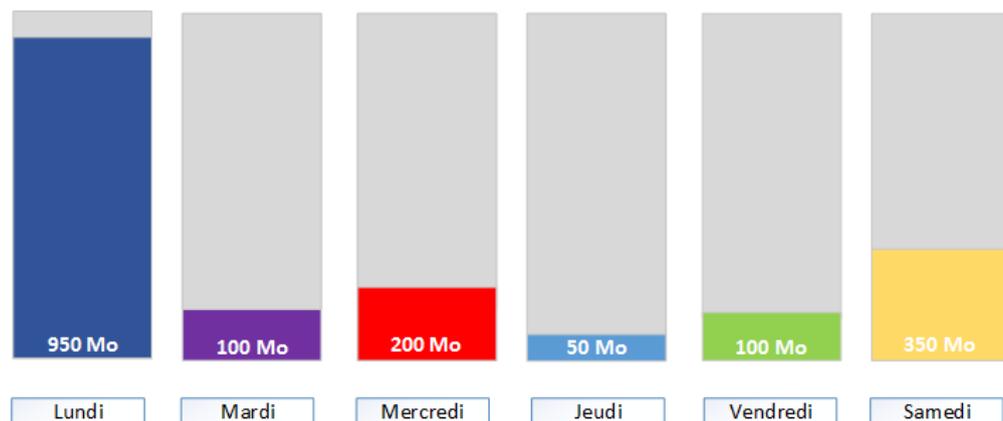
- 1 Les bases de données créées.
- 2 Les bases système *master*, *msdb*, *distribution*
- 3 Les journaux de transactions (pour tenir compte des modifications apportées aux données)

Quand sauvegarder ?

- Tous les jours, ou presque! Tout dépend du type de la base de données.
- Plusieurs critères peuvent être considérés pour établir une stratégie de sauvegarde. En voici quelques-uns :
 - ① L'Indice de volatilité des données, qui indique si les données sont fréquemment mises à jour ou non.
 - **Mise à jour fréquente** ⇒ **sauvegarde fréquente.**
 - ② L'Espace alloué au journal des transactions.
 - **Un journal de faible taille** ⇒ **des sauvegardes fréquentes.**
 - ③ La qualité des supports de sauvegarde.
 - ④ La période où est possible de faire une sauvegarde (Ex: La nuit).

⇒ De manière générale, sauvegardez vos bases de données toutes les semaines !!!

La sauvegarde incrémentielle



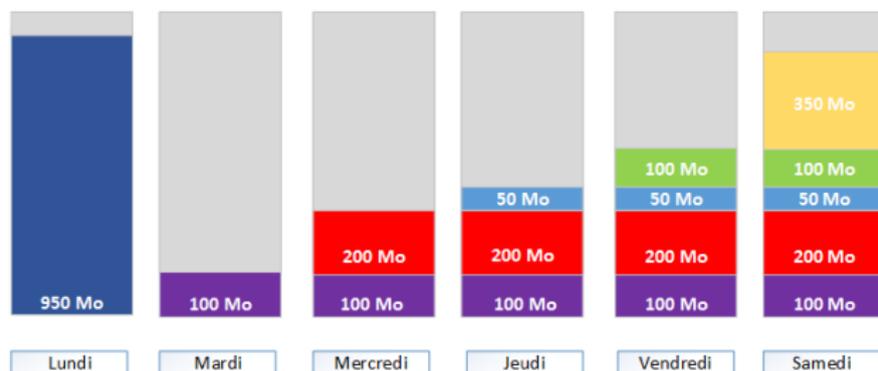
⇒ Pour commencer, réaliser en premier lieu une sauvegarde complète (lundi).

⇒ L'incrémentielle du mardi va se baser sur la précédente qui sera cette fois la complète en sauvegardant uniquement les nouveaux fichiers créés ou modifiés entre-temps.

La sauvegarde différentielle

La sauvegarde différentielle est une sauvegarde de base de données qui capture les pages modifiées depuis la dernière sauvegarde complète. La restauration d'une sauvegarde différentielle est donc plus rapide, puisqu'elle contient les différences survenues depuis la dernière sauvegarde complète.

La sauvegarde différentielle



⇒ Le lundi, on démarre avec une sauvegarde complète. Et c'est à partir de cette sauvegarde que les différentielles se baseront chaque jour.

⇒ Mardi la différentielle va sauvegarder uniquement les nouveaux fichiers créés ou modifiés entre-temps d'après la complète. Soit n .

⇒ Mercredi la différentielle va sauvegarder uniquement les nouveaux fichiers créés ou modifiés entre-temps d'après la complète. Soit $n+1$.

La sauvegarde - Récapitulatif

- 1 L'incrémentielle se base sur la sauvegarde précédente
- 2 La différentielle se base sur sa sauvegarde complète.
- 3 L'incrémentielle nécessite moins d'espace de stockage, mais demande un temps de restauration plus long.
- 4 La différentielle nécessite plus d'espace de stockage, mais demande un temps de restauration plus court.

Les unités de sauvegarde

Il existe plusieurs types d'unités, regroupant les différents média standards de stockage informatique :

- Bande
- Disque
- Disque réseau
- Canal nommé (permettre la connexion de logiciels de sauvegarde)

La création des unités

■ Syntaxe

```
sp_addumpdevice [@devtype =] 'type_d'unité',  
[@logicalname =] 'nom_logique',  
[@physicalname =] 'nom_physique'  
[, { [@cntrltype =] type_de_contrôleur | [@devstatus =] 'état_unité' } ]
```

■ Exemple

```
sp_addumpdevice 'disk', 'SAUVEMABASE',  
'C:\MSSQL\BACKUP\MABASE.BAK'
```

Gérer les Unités

Éditer le contenu d'une unité: Une unité va contenir une ou plusieurs sauvegardes de bases de données, de journaux de transaction et de tables.

■ Syntaxe

```
sp_helpdevice [[@devname=] 'nom']
```

■ Exemple

```
sp_helpdevice 'SAUVEMABASE';
```

Gérer les Unités

Supprimer une unité de sauvegarde: Supprimer une unité de sauvegarde ne supprime pas le fichier physique associé, mais simplement sa référence dans la table *sysdevices*

■ Syntaxe

```
sp_dropdevice [@logicalname =] 'unité' [, [@delfile =] 'suppr_fichier']
```

La sauvegarde

Sauvegarde complète de la base de données:

La première chose à sauvegarder est la base elle-même : cela assure qu'en cas de crash, on pourra revenir à la dernière sauvegarde pour restaurer les structures et les données. Une sauvegarde complète est le point de références des sauvegardes de journaux et différentielles ultérieures.

La sauvegarde

Syntaxe

```
BACKUP DATABASE {nom_base | @var_nom_base}  
TO <unité_sauvegarde> [...n]  
[WITH  
[[.] DIFFERENTIAL]  
[[.] EXPIREDATE = {date | @var_date}  
| RETAIN_DAYS = {jours | @var_jours}]  
[[.] PASSWORD = { mot_de_passe | @var_mot_de_passe } ]  
[[.] FORMAT | NOFORMAT]  
[[.] {INIT | NOINIT}]  
[[.] MEDIADESCRIPTION = {texte | @var_texte}]  
[[.] MEDIA_NAME = {nom_support | @var_nom_support}]  
[[.] MEDIA_PASSWORD = { mot_de_passe_media | @var_mot_de_passe_media } ]  
[[.] [NAME = {nom_jeu_sauve | @var_nom_jeu_sauve}]  
[[.] {NOSKIP | SKIP}]  
[[.] {NOUNLOAD | UNLOAD}]  
[[.] [RESTART]  
[[.] STATS [= pourcentage]]
```

La sauvegarde

Exemple

```
BACKUP DATABASE mabase TO SAUVEMABASE
```

Sauvegarde du journal des transactions

Syntaxe

```
BACKUP LOG {nom_base_de_données | @var_nom_base_de_données}
{
[WITH { NO_LOG | TRUNCATE_ONLY }] }{
TO <unité_sauvegarde> [...n]
[WITH
[.] EXPIREDATE = {date | @var_date}
| RETAIN_DAYS = {jours | @var_jours}]
[.] PASSWORD = { mot_de_passe | @var_mot_de_passe } ]
[.] FORMAT | NOFORMAT]
[.] MEDIADESCRIPTION = {texte | @var_texte}]
[.] MEDIANAME = {nom_support | @var_nom_support}]
[.] MEDIAPASSWORD = { mot_de_passe_media | @var_mot_de_passe_media } ]
[.] [NAME = {nom_jeu_sauve | @var_nom_jeu_sauve}]
NO_TRUNCATE
[.] {NOSKIP | SKIP}]
[.] [RESTART]
[.] STATS [= pourcentage]]
}] }
```

Sauvegarde du journal des transactions

Exemple

```
BACKUP LOG Mabase TO Sauvelog;
```

sauvegarde le journal de la base mabase dans l'unité Sauvelog.

Restauration d'une base de données

Restauration d'une base de données

Restauration d'une base de données

La restauration de la base de données restitue le contenu complet de la base de données. Elle ne doit donc pas être en exploitation lors de la restauration.

Restauration d'une sauvegarde d'une base de données

■ Exemple

```
RESTORE DATABASE mabase FROM Sauvebase;
```

Restauration d'une sauvegarde d'un fichier log

■ Exemple

```
RESTORE LOG mabase FROM Sauvelog
```

Synthèse

Restauration suite à corruption de la base de données

- 1 Faire d'urgence une sauvegarde du journal des transactions :
BACKUP LOG mabase **TO** unité_sauvegarde **WITH NO_TRUNCATE**
- 2 Supprimer la base avec l'instruction **DROP DATABASE** ;
- 3 Restaurer la base de données avec **RESTORE DATABASE** :
 - **RESTORE DATABASE** mabase **FROM** unité_sauvegarde **WITH NORECOVERY**
 - **RESTORE LOG** mabase **FROM** unité_sauvegarde
 - ...
 - **RESTORE LOG** mabase **FROM** unité_sauvegarde

Crash des disques de la base de données

- 1 Faire une sauvegarde d'urgence du journal des transactions :
BACKUP LOG mabase **TO** unité_sauvegarde
- 2 Remplacer et formater les disques incriminés;
- 3 Supprimer la base avec l'instruction **DROP DATABASE**;
- 4 Restaurer la base de données avec **RESTORE DATABASE**:
 - **RESTORE DATABASE** mabase **FROM** unité_sauvegarde
 - **RESTORE LOG** mabase **FROM** unité_sauvegarde
 - ...
 - **RESTORE LOG** mabase **FROM** unité_sauvegarde

...

Crash des disques du journal des transactions

En cas de crash des disques du journal des transactions, le système recrée automatiquement un journal à l'emplacement du fichier .MDF permettant l'exploitation de la base.

- 1 Changer le disque défectueux;
- 2 Détacher la base incriminée: `sp_detach_db mabase`
- 3 Déplacer le fichier journal que le système a créé automatiquement et le renommer avec l'ancien nom;
- 4 Rattacher la base: `sp_attach_db mabase, "fichierMDF", "FichierNDF",..., "FichierLDF"`

Saturation du journal des transactions

- 1 Faire d'urgence un vidage du journal des transactions: **BACKUP LOG** mabase **WITH NO_LOG**
- 2 Exécuter une instruction **CHECKPOINT** sur la base de production;
- 3 Faire immédiatement une sauvegarde de la base de données: **BACKUP DATABASE** mabase **TO** unité_sauvegarde
- 4 Augmenter la taille du journal des transactions pour éviter le phénomène.