

# UTILISER SQL SERVER BCP<sup>1</sup>

POUR EXPORTER LES DONNEES DES GRANDES TABLES  
(100 MILLIONS DE LIGNES)

## LEVIERS DE PERFORMANCE

**LA TAILLE DES PAQUETS RESEAU:** Le paramètre -a spécifie combien d'octets de données sont envoyés par le Serveur SQL.

**DISQUES:** Sys de disques rapides pour l'emplacement des fichiers exportés

**NIC:** Utilisation d'une carte NIC (réseau) rapide

## EXEMPLE

```
bcp <table> out <file> -S <server> -T -w -a 32768
```

## BENCHMARK

± 10 mn soit ± 170k lignes/seconde

POUR IMPORTER UNE ENORME QUANTITE DE DONNEES DANS DES TABLES  
(100 MILLIONS DE LIGNES)

## LEVIERS DE PERFORMANCE

**MINIMUM LOGGING OPTION:** La table cible ne doit pas être répliquée ou memory-optimized

**TABLOCK:** Utiliser l'indice TABLOCK dans la commande BCP, et aussi la table doit être vide si elle inclut un index clusterisé.

**NATIVE DATA FORMAT:** Microsoft recommande d'utiliser les formats de données natifs lorsque les données sont transférées entre deux instances de SQL Server pour des tables identiques si elles ne contiennent pas un jeu de caractères (DBCS), évitant ainsi la conversion inutile des types de données vers et depuis le format de caractères.

## EXEMPLE

```
bcp <table> in <file> -S <server> -T -n -h"TABLOCK"
```

## BENCHMARK

± 5 minutes soit ± 400k lignes/seconde et Log Size ± 200 Mb

---

<sup>1</sup> Utilitaire de ligne de commande très simple qui exporte les données des tables dans des fichiers plats depuis SQL Server, ou qui peut importer des données dans les tables de la base de données. \*\*\* Si vous exportez/importez des données de l'ordre de dizaines de millions, il est recommandé de les diviser en plus petites quantités. Et cela peut se faire en utilisant l'instruction SELECT avec la clause WHERE. Nous devons aussi faire attention aux contraintes définies. Parfois, nous devons supprimer la contrainte et charger les données, puis remettre les contraintes supprimées. \*\*\*