Hadoop

Guide d'utilisation d'Hadoop pour le stockage des données

Introduction

Hadoop est un framework open-source utilisé pour le stockage et le traitement des données volumineuses (Big Data). Son système de fichiers distribué, Hadoop Distributed File System (HDFS), permet de stocker de grandes quantités de données sur un cluster de machines.

1. Concepts clés d'Hadoop pour le stockage

1.1. Hadoop Distributed File System (HDFS)

HDFS est un système de fichiers distribué conçu pour stocker de grands volumes de données en les répartissant sur plusieurs nœuds d'un cluster.

Caractéristiques principales :

- **Stockage distribué** : les fichiers sont divisés en blocs et répliqués sur plusieurs nœuds.
- **Tolérance aux pannes** : les données sont répliquées pour éviter toute perte en cas de défaillance d’un nœud.
- **Scalabilité** : la capacité de stockage peut être augmentée en ajoutant de nouveaux nœuds au cluster.

1.2. Blocs et réplication

- Les fichiers sont découpés en blocs de **128 Mo** (par défaut).
- Chaque bloc est répliqué **3 fois** (par défaut) sur différents nœuds pour assurer la redondance.
- Le **Namenode** gère la structure du système de fichiers et les métadonnées des fichiers.
- Les **Datanodes** stockent les blocs de données réels.

2. Gestion des fichiers dans HDFS

2.1. Stockage des données

HDFS suit une approche **write once, read many** (écriture unique, lecture multiple), ce qui signifie que les fichiers ne peuvent pas être modifiés une fois écrits, mais peuvent être lus plusieurs fois.

2.2. Commandes de base pour interagir avec HDFS

Hadoop fournit une interface en ligne de commande (CLI) pour gérer les fichiers et dossiers dans HDFS.

a) Vérification du bon fonctionnement d'HDFS

```
hdfs dfsadmin -report
```

Cette commande affiche l'état du cluster HDFS, y compris les nœuds actifs et la capacité utilisée.

b) Création d'un répertoire dans HDFS

```
hdfs dfs -mkdir /user/mon_utilisateur
```

c) Copier un fichier local vers HDFS

```
hdfs dfs -put fichier_local.txt /user/mon_utilisateur/
```

d) Lister les fichiers et répertoires dans HDFS

```
hdfs dfs -ls /user/mon_utilisateur
```

e) Lire un fichier stocké dans HDFS

```
hdfs dfs -cat /user/mon_utilisateur/fichier_local.txt
```

f) Télécharger un fichier depuis HDFS vers le système local

```
hdfs dfs -get /user/mon_utilisateur/fichier_local.txt ./
```

g) Supprimer un fichier ou répertoire dans HDFS

```
hdfs dfs -rm /user/mon_utilisateur/fichier_local.txt
hdfs dfs -rm -r /user/mon_utilisateur/
```

3. Optimisation du stockage dans HDFS

3.1. Compression des fichiers

Pour optimiser l'utilisation du stockage, vous pouvez compresser vos fichiers avant de les stocker dans HDFS.

Exemple avec Gzip :

```
gzip fichier.txt
hdfs dfs -put fichier.txt.gz /user/mon_utilisateur/
```

3.2. Gestion des répliques

Le facteur de réplication peut être ajusté pour optimiser l'espace de stockage :

```
hdfs dfs -setrep -w 2 /user/mon_utilisateur/fichier_local.txt
```

3.3. Nettoyage des fichiers obsolètes

Supprimer les fichiers inutilisés permet de libérer de l’espace :

```
hdfs dfs -rm /user/mon_utilisateur/fichier_inutile.txt
```

4. Bonnes pratiques

- **Utiliser la compression** pour économiser de l’espace de stockage.
- **Gérer le facteur de réplication** en fonction des besoins de redondance.
- **Surveiller régulièrement l’état du cluster** avec `hdfs dfsadmin -report`.
- **Ne pas stocker de petits fichiers** car HDFS est optimisé pour les gros volumes de données.

Conclusion

HDFS est un système de stockage puissant et fiable pour la gestion des données massives. En suivant ces bonnes pratiques et en utilisant les commandes appropriées, vous pouvez efficacement gérer vos fichiers dans un environnement Hadoop.