

## Formation Programmer Hadoop en Java

<b>Durée :</b>	5 jours
<b>Public :</b>	Développeurs Java, Administrateurs (DBA ou systèmes)
<b>Pré-requis :</b>	Maîtrise de la programmation orientée objets en Java - Développer des algorithmes parallèles efficaces - Analyser des fichiers non structurés et développer des tâches Java MapReduce - Charger et récupérer des données de HBase et du système de fichiers distribué Hadoop (HDFS) - User Defined Functions de Hive et Pig
<b>Objectifs :</b>	
<b>Sanction :</b>	Attestation de fin de stage mentionnant le résultat des acquis
<b>Taux de retour à l'emploi:</b>	Aucune donnée disponible
<b>Référence:</b>	BUS101601-F
<b>Note de satisfaction des participants:</b>	Pas de données disponibles

### Comprendre le contexte d'utilisation d'Hadoop

- Évaluer la valeur que peut apporter Hadoop à l'entreprise
- Examiner l'écosystème d'Hadoop
- Choisir un modèle de distribution adapté

### Défier la complexité de la programmation parallèle

- Examiner les difficultés liées à l'exécution de programmes parallèles : algorithmes, échange des données
- Évaluer le mode de stockage et la complexité du Big Data

### Programmation parallèle avec MapReduce

- Fragmenter et résoudre les problèmes à grande échelle
- Découvrir les tâches compatibles avec MapReduce
- Résoudre des problèmes métier courants

### Appliquer le paradigme Hadoop MapReduce

- Configurer l'environnement de développement
- Examiner la distribution Hadoop
- Étudier les démons Hadoop
- Créer les différents composants des tâches MapReduce
- Analyser les différentes étapes de traitement MapReduce : fractionnement, mappage, lecture aléatoire et réduction

## **Créer des tâches MapReduce complexes**

Choisir et utiliser plusieurs outils de mappage et de réduction, exploiter les partitionneurs et les fonctions map et reduce intégrées, analyser les données en séries temporelles avec un second tri, rationaliser les tâches dans différents langages de programmation

## **Résoudre les problèmes de manipulation des données**

Exécuter les algorithmes : tris, jointures et recherches parallèles, analyser les fichiers journaux, les données des média sociaux et les courriels

## **Mise en œuvre des partitionneurs et des comparateurs**

Identifier les algorithmes parallèles liés au réseau, au processeur et aux E/S de disque  
Répartir la charge de travail avec les partitionneurs  
Contrôler l'ordre de groupement et de tri avec les comparateurs  
Mesurer les performances avec les compteurs

## **Bien-fondé des données distribuées**

Optimiser les performances du débit des données  
Utiliser la redondance pour récupérer les données

## **Interfacer avec le système de fichiers distribué Hadoop**

Analyser la structure et l'organisation du HDFS  
Charger des données brutes et récupérer le résultat  
Lire et écrire des données avec un programme  
Manipuler les types SequenceFile d'Hadoop  
Partager des données de référence avec DistributedCache

## **Structurer les données avec HBase**

Passer du stockage structuré au stockage non structuré  
Appliquer les principes NoSQL avec une application de modèle à la lecture, se connecter à HBase à partir des tâches MapReduce, comparer HBase avec d'autres types de magasins de données NoSQL

## **Exploiter la puissance de SQL avec Hive**

Structurer bases de données, les tables, les vues et les partitions  
Intégrer des travaux MapReduce avec des requêtes Hive  
Lancer des requêtes avec HiveQL  
Accéder aux servers Hive via JDBC, ajouter des fonctionnalités à HiveQL avec les fonctions définies par l'utilisateur

## **Tester et déboguer le code Hadoop**

Enregistrer des événements importants à auditer et à déboguer  
Valider les spécifications avec MRUnit  
Déboguer en mode local

## **Déployer, surveiller et affiner les performances**

Déployer la solution sur un cluster de production  
Utiliser des outils d'administration pour optimiser les performances  
Surveiller l'exécution des tâches via les interfaces utilisateur web