

Formation Pig, Hive et Impala

Durée :	4 jours
Public :	techniciens et spécialistes des bases de données, responsables, analystes métier et professionnels BI, souhaitant utiliser les technologies Big Data dans leur entreprise
Pré-requis :	Connaissances fondamentales des bases de données et de SQL sont un atout majeur
Objectifs :	- Manipuler des ensembles de données complexes stockés dans Hadoop sans avoir à écrire de code complexe avec Java - Automatiser le transfert des données dans le stockage Hadoop avec Flume et Sqoop - Filtrer les données avec les opérations Extract-Transform-Load (ETL) avec Pig - Interroger plusieurs ensembles de données pour une analyse avec Pig et Hive
Sanction :	Attestation de fin de stage mentionnant le résultat des acquis
Taux de retour à l'emploi:	Aucune donnée disponible
Référence:	BUS100295-F
Note de satisfaction des participants:	Pas de données disponibles

Introduction

- Vue d'ensemble de Hadoop
- Analyser les composants Hadoop
- Définir l'architecture Hadoop

Stocker les données dans HDFS

- Réaliser un stockage fiable et sécurisé
- Surveiller les mesures du stockage
- Contrôler HDFS à partir de la ligne de commande

Traitement parallèle avec MapReduce

- Détailler l'approche MapReduce
- Transférer les algorithmes et non les données
- Décomposer les étapes clés d'une tâche MapReduce

Automatiser le transfert des données

- Faciliter l'entrée et la sortie des données
- Agréger les données avec Flume

Configurer le fan in et le fan out des données
Déplacer les données relationnelles avec Sqoop

Décrire les caractéristiques d'Apache Pig

Exposer les différences entre Pig et MapReduce
Identifier les cas d'utilisation de Pig
Identifier les configurations clés de Pig

Structurer les données non structurées

Représenter les données dans le modèle de données de Pig
Exécuter les commandes Pig Latin au Grunt Shell
Exprimer les transformations dans la syntaxe Pig Latin
Appeler les fonctions de chargement et de stockage

Transformer les données avec les opérateurs relationnels

Créer des nouvelles relations avec des jointures
Réduire la taille des données par échantillonnage
Exploiter Pig et les fonctions définies par l'utilisateur

Filtrer les données avec Pig

Consolider les ensembles de données avec les unions
Partitionner les ensembles de données avec les splits
Ajouter des paramètres dans des scripts Pig

Exploiter les avantages métier de Hive

Factoriser Hive en composants
Imposer la structure sur les données avec Hive

Organiser les données dans Hive

Créer des bases de données et des tables Hive
Exposer les différences entre les types de données dans Hive
Charger et stocker les données efficacement avec SerDes

Concevoir la disposition des données pour la performance

Remplir les tables à partir de requêtes
Partitionner les tables de Hive pour des requêtes optimales
Composer des requêtes HiveQL

Réaliser des jointures sur des données non structurées

Distinguer les jointures disponibles dans Hive
Optimiser la structure des jointures pour les performances

Repousser les limites de HiveQL

Trier, répartir et regrouper des données
Réduire la complexité des requêtes avec les vues
Améliorer la performance des requêtes avec les index

Déployer Hive en production

Concevoir les schémas de Hive
Établir la compression des données
Déboguer les scripts de Hive

Rationaliser la gestion du stockage avec HCatalog

Unifier la vue des données avec HCatalog
Exploiter HCatalog pour accéder au metastore Hive
Communiquer via les interfaces HCatalog
Remplir une table Hive à partir de Pig

Traitement parallèle avec Impala

Décomposer les composants fondamentaux d'Impala
Soumettre des requêtes à Impala
Accéder aux données Hive à partir d'Impala

Lancer le framework Spark

Réduire le temps d'accès aux données avec Spark-SQL
Interroger les données Hive avec Spark-SQL