

## Hadoop : stockage avec HBase

Durée: 2 jours

1160 €

1er et 2 mars  
21 et 22 juin

4 et 5 octobre  
13 au 14 décembre

### Public:

Chefs de projet, administrateurs et toute personne souhaitant stocker des données avec Hbase.

### Objectifs:

Comprendre le fonctionnement de HBase, savoir mettre en place une configuration distribuée.

### Connaissances préalables nécessaires:

Connaissance des principes de base Hadoop et des bases de données.

### Programme:

- Introduction** : Rappels rapides sur l'écosystème Hadoop.  
Les fonctionnalités du framework Hadoop  
Le projet et les modules : Hadoop Common, HDFS, YARN, Spark, MapReduce  
Présentation HBase. Historique. Lien avec HDFS.  
Format des données dans HBase  
Définitions : table, région, ligne, famille de colonnes, cellules, espace de nommage, ...  
Fonctionnalités : failover automatique, sharding, interface avec des jobs MapReduce.
- Architecture** : HBase master node, Region Master, liens avec les clients HBase.  
Présentation du rôle de Zookeeper.
- Installation** : Choix des packages.  
Installation et configuration dans le fichier conf/hbase-site.xml  
Démarrage en mode standalone start-hbase.  
Test de connexion avec hbase shell.  
Installation en mode distribué.  
Travaux pratiques :  
Interrogations depuis le serveur http intégré.

## Hadoop : stockage avec HBase

- HBase** : Présentation des différentes interfaces disponibles.
- utilisation : shell** Travaux pratiques avec hbase shell.  
Commandes de base, syntaxe, variables,  
manipulation des données : create, list, put, scan, get  
désactiver une table ou l'effacer : disable (enable), drop, ...  
Programmation de scripts.  
Gestion des tables : principe des filtres.  
Mise en oeuvre de filtres de recherche, paramètres des tables.  
Présentation des espaces de nommage.
- Cluster HBase** : Fonctionnement en mode distribué  
Première étape : fonctionnement indépendant des démons (HMaster, HRegionServer, Zookeeper)  
Passage au mode distribué :  
mise en oeuvre avec HDFS dans un environnement distribué.  
Travaux pratiques :  
sur un exemple de tables réparties : mise en oeuvre des splits.
- Programmation** : Introduction, les APIs (REST, Avro, Thrift, Java, Ruby, ...)  
Utilisation d'un client Java.  
Gestion des tables. Lien avec MapReduce.  
Principe des accès JMX.  
Travaux pratiques :  
création d'un client JMX