Réponses aux guides d’étude

Chapitre Introductif

Qu’est-ce qu’un paradigme ?

Un paradigme est une perception des choses, un schéma de pensée, une idéologie, qui produit automatiquement une réponse, un comportement, une façon de faire précise.

Pourquoi dit-on que le MapReduce est un paradigme ?

On dit qu’il s’agit d’un paradigme parce que programmer en MapReduce, c’est penser autrement la programmation des tâches parallèles : penser au découpage d’un problème en tâches indépendantes.

Qu’est-ce qu’un modèle algorithme ?

C’est une façon de programmer, une manière de penser le découpage d’un problème en tâches.

En quoi consiste le MapReduce ?

Il s’agit de découper un problème en tâches indépendantes qui s’exécutent en trois phases : Map, Shuffle et Reduce.

Quel est le but du MapReduce ?

Son but est de masquer la complexité des opérations liées à la parallélisation, à la distribution des traitements de données, à la gestion de leur exécution dans le cluster et à la gestion des défaillances qui peuvent survenir pendant le traitement.

Quelle définition donneriez-vous d’Hadoop ?

C’est un ensemble de classes Java servant à faire du calcul massivement parallèle sur un cluster de façon complètement tolérante aux pannes.

Quelle différence faites-vous entre le MapReduce et Hadoop ?

Le MapReduce est un paradigme et Hadoop est une implémentation logicielle de ce paradigme.

Quelle est la différence entre le jobtracker, le nœud de référence et le processus Master ?

Le nœud de référence est la machine centrale du cluster ; c’est à son niveau qu’est déclenché le processus Master. Ce dernier est le processus central du cluster. Dans le cadre de Hadoop, il s’appelle le jobtracker et, en plus de coordonner l’exécution des tâches MapReduce, il s’assure également de leur planification sur le cluster.

Qu’est-ce que le tasktracker ?

C’est un processus qui s’exécute sur les nœuds de données du cluster. Il est chargé d’exécuter soit une tâche Map, soit une tâche Reduce.

Quelle est la différence entre le tasktracker, le nœud de données et le processus Worker ?

Le nœud de données est la machine esclave du cluster. C’est à son niveau qu’est déclenché le Worker, un processus chargé de l’exécution des tâches Map ou Reduce. Dans la terminologie Hadoop, le Worker porte le nom de tasktracker.

Pourquoi est-il nécessaire de diviser le fichier d’entrée en plusieurs blocs de taille égale (de 64 Mo par défaut) pour le traitement ?

Diviser le fichier d’entrée en blocs de taille égale répartit mieux le traitement parallèle entre les nœuds du cluster, c’est-à-dire que chaque nœud reçoit exactement la même charge de travail. Comme tous les nœuds de données ont la même configuration, le traitement devrait normalement s’achever à peu près au même moment.

Qu’est-ce que le *facteur de réplication* dans un cluster Hadoop ?

C’est le nombre de fois qu’un bloc de fichier est répliqué dans le cluster par le HDFS. Il est défini par l’utilisateur et vaut 3 par défaut.

Par défaut, combien de nœuds de données sont utilisés pour un job MapReduce complet ? Justifiez votre réponse.

M nœuds pour le nombre de tâches Map, avec le nombre de tâches Map égal au nombre de blocs du fichier à traiter + R nœuds pour le nombre de tâches Reduce + le nœud de référence N qui supervise l’exécution. Donc M + R + N nœuds sont nécessaires pour l’exécution complète d’un job MapReduce.

Par défaut, combien de tasktrackers sont démarrés lors d’un job MapReduce ? Justifiez votre réponse.

M + R tasktrackers sont démarrés lors d’un job MapReduce. M est automatique et est égal au nombre de partitions du fichier à traiter et R est fixé par l’utilisateur et correspond au nombre de tâches Reduce à exécuter.

Y a-t-il autant de jobtrackers déclenchés que de jobs MapReduce exécutés ?

Non.

Déterminez l’ordre d’exécution d’un job MapReduce dans la liste suivante (attribuez un numéro entre 1 et 7) :

1. Configuration du job MapReduce

2. Découpe du fichier d’entrée en blocs de taille fixe par le HDFS

3. Création de M tasktrackers correspondant aux M blocs de fichiers

4. Application de la fonction Map et sérialisation sur le disque dur local du nœud

5. Tri des paires clé-valeur et passage à la fonction Reduce pour traitement

6. Application de la fonction Reduce et sérialisation dans le fichier ri

7. Sérialisation des r fichiers Reduce dans un SGBD

Pour développer des jobs MapReduce, faut-il forcément être développeur Java ?

Non.

Qu’est-ce que l’écosystème Hadoop ?

C’est l’ensemble des outils qui facilitent le développement d’applications sous Hadoop et son adoption.

Pourquoi a-t-on besoin d’un écosystème Hadoop ?

Sans ces outils, il reviendrait à chaque entreprise, en fonction de son besoin, de développer elle-même des outils compatibles avec Hadoop afin de déployer ses solutions sur le cluster, alors que d’autres ont peut-être déjà résolu le même problème.

Peut-on utiliser Hadoop sans passer par un écosystème d’outils ?

Oui.

Citez trois outils de votre choix de l’écosystème Hadoop et donnez leur rôle.

1. ZooKeeper : service de coordination distribué. Il fournit des fonctions de registre et de consensus, entre autres, pour permettre le partage de la donnée et la synchronisation des nœuds d’un cluster.

2. YARN : application générique de gestion des ressources, qui permet d’exécuter plusieurs modèles de calcul sur un cluster Hadoop.

3. Lucene : ensemble de classes Java destinées au développement d’applications de recherche et d’indexation de contenu.

Chapitre 1

Qu’est-ce qu’un modèle de calcul ?

Le modèle de calcul définit la façon dont l’utilisateur va écrire ses algorithmes et la façon dont ils vont s’exécuter.

Citez les quatre principes de base du traitement informatique de données.

Tout traitement informatique des données implique toujours la participation de trois composants de l’ordinateur : le disque dur, la mémoire centrale (RAM) et le micro-processeur.

Le micro-processeur traite toujours les données à partir de la RAM.

La mémoire RAM garde son contenu le temps d’une session.

Le disque dur a été conçu pour la persistance des données.

Qu’est-ce que le *batch processing* ou traitement sur disque ?

C’est un traitement qui consiste à traiter les données par lots.

Quel est l’avantage du batch processing ?

Il permet de traiter de gros volumes de données.

Quel est l’inconvénient majeur du batch processing ?

Son inconvénient est la latence dans les calculs, due aux innombrables allers-retours entre le disque dur et la mémoire.

En quoi le batch processing facilite-t-il le calcul parallèle ?

L’intérêt du batch processing pour le calcul parallèle est que chaque nœud du cluster possède son propre disque dur ; par conséquent, les données à traiter peuvent être répliquées ou être découpées en blocs, dont le stockage sera distribué entre les nœuds du cluster.

Expliquez l’expression suivante : « le calcul des index inversés est un problème *embarrasingly parallel* »

Le calcul des index inversés est un problème facile à paralléliser. « embarrasingly parallel » ne se traduit pas en français par « embarrassant » ou « difficile », comme on pourrait le croire.

Quelle est la condition indispensable pour qu’un problème s’exécute sur un cluster ?

Il faut qu’il soit parallélisable.

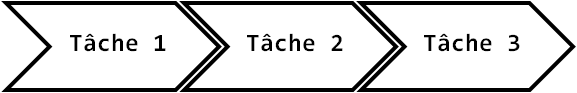
Peut-on dire qu’un traitement batch parallèle est un traitement séquentiel répliqué plusieurs fois sur les nœuds d’un cluster ?

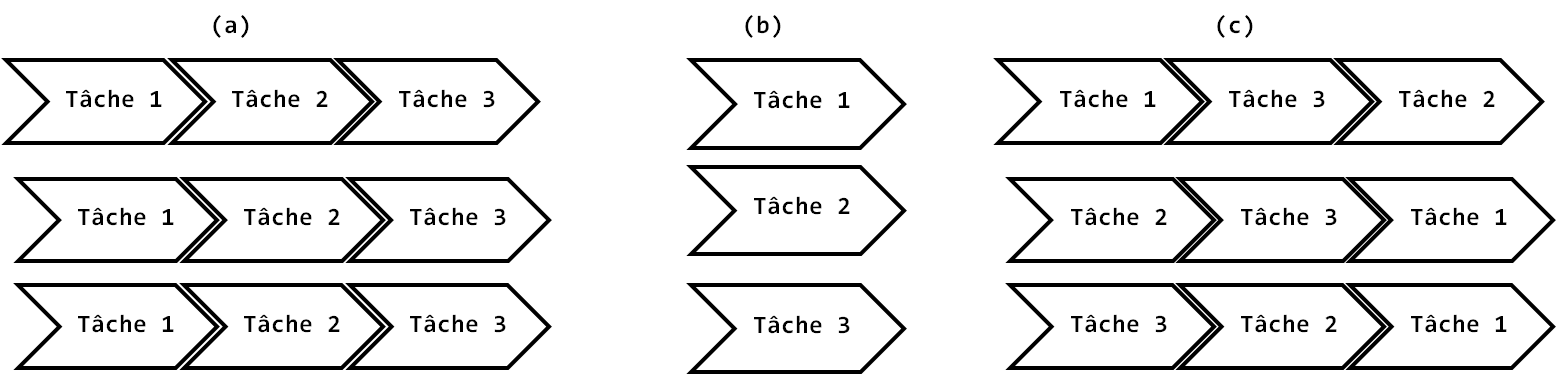
Oui.

Peut-on dire qu’un traitement batch parallèle est un traitement découpé en plusieurs tâches indépendantes dont l’exécution est parallélisée dans un cluster ?

Non.

Des trois figures suivantes, laquelle représente le batch parallèle du traitement séquentiel suivant ? Justifiez votre réponse.





La réponse correcte est la (a) car, dans un traitement batch parallèle, le même traitement séquentiel est répliqué plusieurs fois sur les nœuds du cluster, sur des blocs de données différents.

En quoi le MapReduce consiste-t-il ?

Il consiste à découper un problème en tâches indépendantes qui s’exécutent en trois phases : Map, Shuffle et Reduce.

Le MapReduce est-il un modèle algorithmique asynchrone ?

Non.

Le MapReduce est-il un patron de conception (*design pattern*) ?

Non.

Par défaut, combien de tâches Map sont-elles exécutées lors d’un job MapReduce ? Justifiez votre réponse.

Par défaut, le nombre de tâches Map est égal au nombre de blocs du fichier d’entrée.

À quel moment les r fichiers d’entrée du Reduce sont-ils créés ?

Les r fichiers sont créés lorsque l’utilisateur spécifie le nombre de tâches Reduce et exécute le job MapReduce.

Citez et donnez le rôle des trois processus qui sont déclenchés lors de l’exécution du MapReduce dans un cluster.

1. Le processus Master planifie et coordonne l’ensemble des tâches MapReduce qui s’exécutent sur le cluster.

2. Les processus workers Map exécutent les tâches Map.

3. Les processus workers Reduce exécutent les tâches Reduce.

Pour un traitement précis, un nœud peut-il faire tourner à la fois un worker Map et un worker Reduce ? Justifiez votre réponse.

Non, chaque nœud exécute soit un worker Map, soit un worker Reduce. Parce que le MapReduce est un graphe acyclique direct, des nœuds différents sont utilisés à chaque phase.

Quel est le processus chargé de l’exécution du Shuffle ? Cochez la bonne réponse

C’est le processus Maître.

Supposons que l’analyste de données, dans un traitement MapReduce, spécifie le nombre de processus Reduce r=0. Quelle sera l’incidence de cette valeur ?

Si le nombre de tâches Reduce est nul, alors il n’y aura pas de phase Shuffle ni Reduce dans l’exécution du traitement.

Quelle est l’entrée d’une fonction map() ?

Un bloc du fichier d’entrée.

Quelle est l’entrée d’une fonction reduce() ?

Les paires clé/valeur triées de la phase Shuffle.

Supposons que nous souhaitions obtenir l’index inversé des tweets suivants :

Tweet1 : « je mange la banane »

Tweet2 : « je déteste la banane »

Tweet3 : « la banane est un bon aliment »

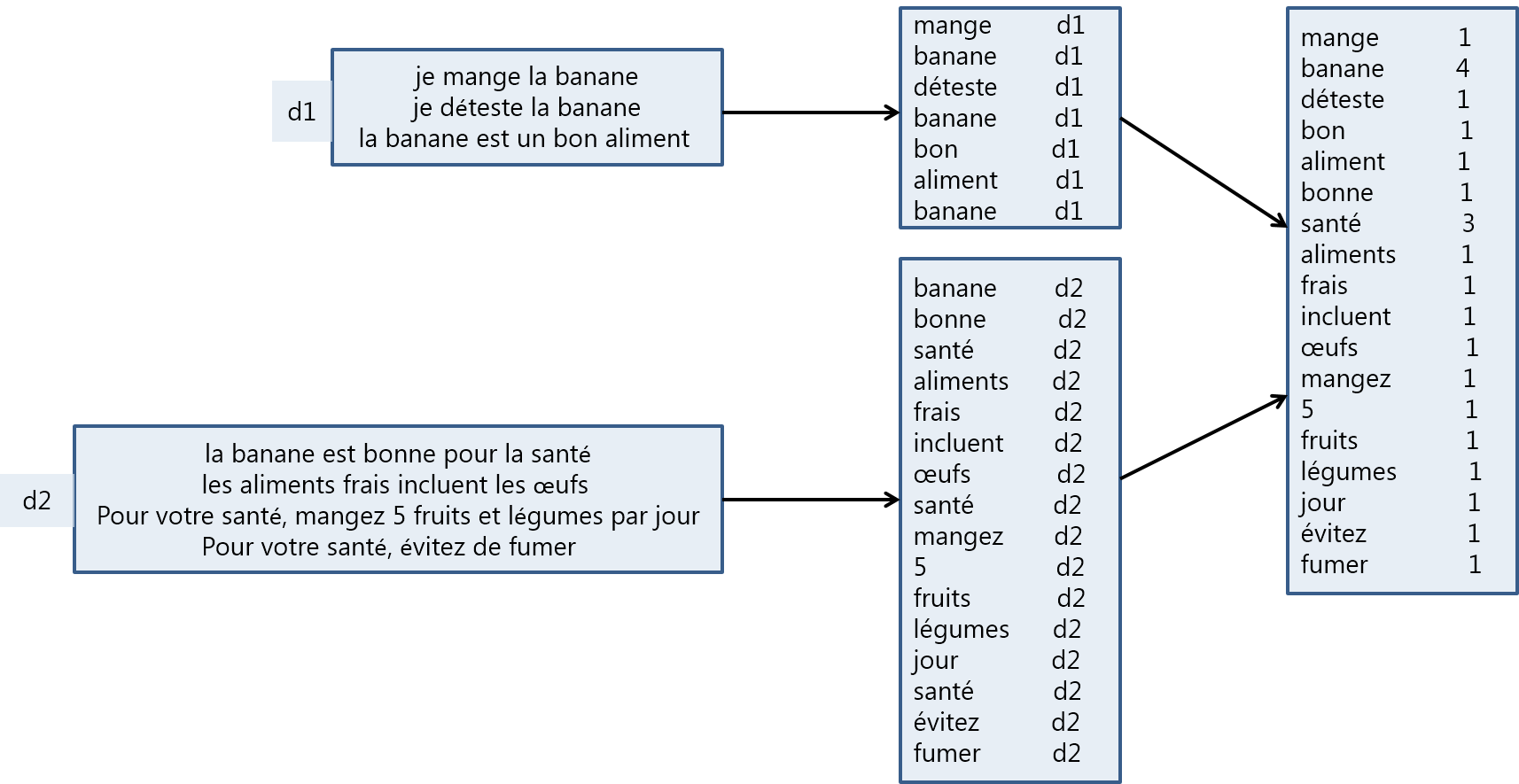
Tweet4 : « la banane est bonne pour la santé »

Tweet5 : « les aliments frais incluent les œufs »

Tweet6 : « Pour votre santé, mangez cinq fruits et légumes par jour »

Tweet7 : « Pour votre santé, évitez de fumer »

Avec m=2 tâches Map et r=1 tâche Reduce, construisez le graphe de flux du MapReduce nécessaire pour l’obtention de l’index inversé.



Soient les tables suivantes :

|  |  |  |  |
| --- | --- | --- | --- |
| Commande | | |  |
| Commande\_ID | Date\_commande | Client |  |
| 1890 | 10/03/2016 | Juvénal |  |
| 1978 | 15/03/2016 | Fabiola |  |
| 2000 | 20/05/2016 | Jean |  |
|  |  |  |  |
| Détail\_commande | | | |
| Commande\_ID | produit | Quantité | Prix unitaire |
| 1890 | livre Big Data | 1 | 33 |
| 1890 | Costume 3 pièces | 1 | 800 |
| 1978 | Gel de douche | 1 | 50 |
| 2000 | Paire de chaussures | 1 | 200 |
| 2000 | Dentifrice | 1 | 10 |
| 2000 | Livre leadership | 1 | 40 |

Nous souhaitons obtenir par jointure la table suivante :

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Commande\_ID | Date\_commande | Client | produit | Quantité | Prix unitaire |
| 1890 | 10/03/2016 | Juvénal | livre Big Data | 1 | 33 |
| 1890 | 10/03/2016 | Juvénal | Costume 3 pièces | 1 | 800 |
| 1978 | 15/03/2016 | Fabiola | Gel de douche | 1 | 50 |
| 2000 | 20/05/2016 | Jean | Paire de chaussures | 1 | 200 |
| 2000 | 21/05/2016 | Jean | Dentifrice | 1 | 10 |
| 2000 | 22/05/2016 | Jean | Livre leadership | 1 | 40 |

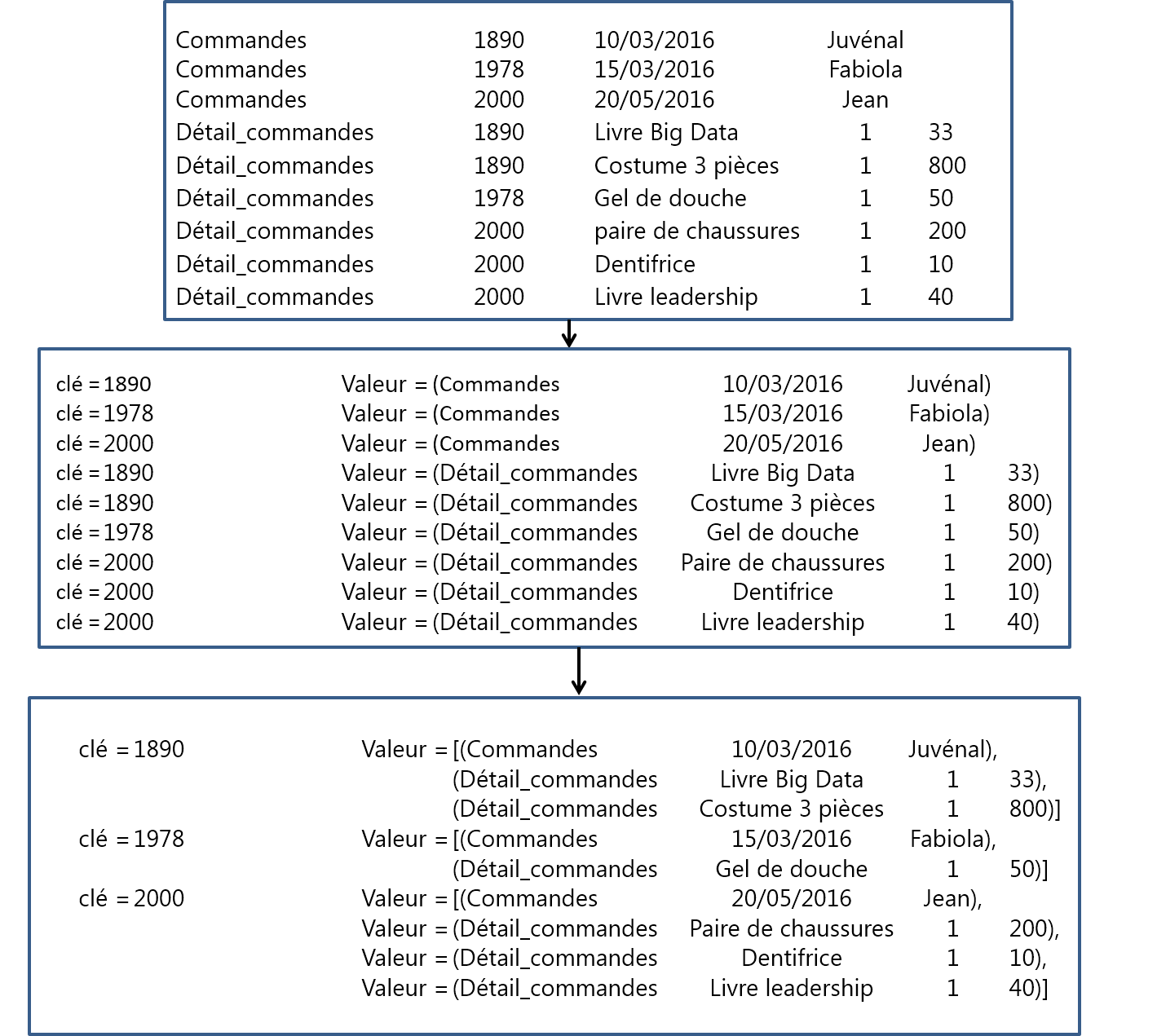
Avec m=2 tâches Map et r=1 tâche Reduce, énoncez les trois étapes nécessaires à la réalisation de la jointure à l’aide du MapReduce, puis construisez le graphe de flux pour l’obtention de la table.

1. Concaténer verticalement les deux tables et ajouter une colonne contenant le nom de chaque table.

2. Passer le fichier obtenu à la fonction map() et la clé de la paire sera la colonne command\_id.

3. Le Reduce regroupe les lignes par clé de façon à obtenir les lignes des deux tables correspondant à la même clé.

Le graphe acyclique direct de la jointure est le suivant :



Citez deux limites majeures du modèle MapReduce.

1. Il est inadapté pour traiter des travaux itératifs.

2. Il est inadapté pour résoudre des problématiques de temps réel.

Comment définiriez-vous Mahout ?

Mahout une bibliothèque de classes Java d’algorithmes d’apprentissage statistique, parallélisés en style MapReduce pour exécution dans un cluster. Cette bibliothèque peut être utilisée par n’importe quel moteur de calcul qui supporte les plans d’exécution acycliques directs tels que le MapReduce.

Quel est le défaut de MapReduce corrigé par Mahout ?

Mahout corrige le problème du manque d’itérativité du MapReduce qui empêche de l’utiliser pour les travaux statistiques.

Mahout fonctionne-t-il en batch processing ?

Oui.

Que pouvez-vous dire d’Hama ?

Hama est un moteur de calcul distribué, spécialisé dans le traitement massivement parallèle des graphes et du calcul matriciel. Il ne s’appuie pas sur un schéma acyclique direct comme le MapReduce, mais sur un modèle de programmation parallèle spécial appelé le BSP - Bulk Synchronous Parallel (lot synchrone parallèle).

Chapitre 2

Qu’est-ce que l’in-memory processing ou traitement en mémoire ?

C’est un traitement qui consiste à charger toutes les données en mémoire avant de les traiter.

Quel est l’avantage de l’in-memory processing ?

L’in-memory processing offre rapidité et interactivité dans le traitement.

Quel est l’inconvénient de l’in-memory processing ?

Le volume de données qui peut être traité est limité à la taille de la mémoire.

Citez les deux tentatives de résolution des limites de l’in-memory processing.

* Compression des données.
* Ajout de barrettes mémoires supplémentaires dans le système.

Selon vous, quel est le meilleur mode de traitement entre le batch processing et l’in-memory processing ? Justifiez votre réponse.

Tout est fonction du besoin de traitement. Si le problème exige l’interactivité et la vitesse, alors l’in-memory sera approprié, mais au prix du volume ; si le besoin concerne des traitements périodiques, alors le batch sera adapté.

Qu’est-ce que l’in-database processing ?

C’est une approche d’accélération des traitements qui consiste à ajouter des services Analytics et OLAP à l’intérieur d’un SGBD, qui s’y exécutent directement.

Expliquez les deux approches qui peuvent être utilisées pour faire du traitement parallèle in-memory sur un cluster.

* Configurer le cluster selon une architecture shared-memory où les données sont chargées dans une mémoire partagée par les nœuds du cluster.
* Utiliser une approche shared-nothing dans laquelle les données sont découpées, distribuées entre les nœuds du cluster et chargées dans la mémoire individuelle de chaque nœud.

À quelle condition est-il approprié d’utiliser l’in-memory en shared-memory ? Justifiez votre réponse.

Cette approche est appropriée lorsqu’il faut traiter de gros fichiers de données qui ne se prêtent pas facilement au découpage en partitions et qui doivent être traités de façon atomique (comme un tout indivisible), à l’instar des graphes des réseaux sociaux. Dans ce cas, aucun nœud ne possède de mémoire ; toute la mémoire est partagée par tous les nœuds du cluster.

À quelle condition est-il approprié d’utiliser l’in-memory en shared-nothing ? Justifiez votre réponse.

Cette approche n’est appropriée que si les fichiers de données se prêtent au découpage et peuvent être traités de façon divisible, car aucun composant n’est partagé entre les nœuds du cluster.

Le Spark est-il un modèle algorithmique ?

FAUX

VRAI

Non.

Quelle différence faites-vous entre Spark et Hadoop ?

Spark est un moteur de calcul in-memory distribué, tandis que Hadoop est l’implémentation du MapReduce, un modèle de calcul batch.

Spark est-il une version modifiée de Hadoop ?

Non.

Qu’est-ce que le RDD et quel est son rôle en Spark ?

Le RDD, Resilient Distributed Dataset (table partagée et résiliente) est une abstraction de données distribuées qui permet aux développeurs d’effectuer des calculs parallèles en mémoire sur un cluster de façon complètement tolérante aux pannes. C’est une abstraction de calcul distribué. Spark est une abstraction du RDD.

Pourquoi dit-on que le RDD est résilient ?

À chaque partition du RDD, Spark mémorise suffisamment d’informations sur elle de manière à ce que, si un nœud venait à tomber en panne, celle-ci serait automatiquement reconstruite et restaurée à l’aide de ces informations, appelées lignage ou lignée du RDD (RDD lineage).

Pourquoi dit-on que le RDD est distribué ?

Le RDD est une collection d’objets ou de partitions qui sont distribué(e)s entre les nœuds du cluster.

Le RDD est-il un modèle de calcul au même titre que le MapReduce ?

Oui.

Spark est-il une implémentation du RDD ?

Oui.

Qu’entend-on par « lignage » du RDD (lineage) ?

C’est l’ensemble des opérations et leur séquencement qui permet de [re]construire le RDD en cas de panne de nœud.

En quoi la programmation Spark consiste-t-elle ?

Elle consiste à programmer des transformations sur les instances de RDD et à invoquer les actions de ces instances pour réaliser vos traitements de données.

Quel est le mécanisme qui permet à Spark de rendre un cluster tolérant aux pannes ?

C’est la persistance des partitions de RDD en mémoire cache.

Spark peut-il tourner sur un cluster Hadoop ?

Oui.

Expliquez brièvement l’exécution de Spark sur un cluster Hadoop.

Spark peut tourner sur le même cluster que Hadoop grâce à un gestionnaire de ressources, YARN par défaut. Le pilote Spark est déployé dans une application YARN, les partitions de RDD sont emboîtées dans des conteneurs qui sont exécutés au niveau des nœuds de données.

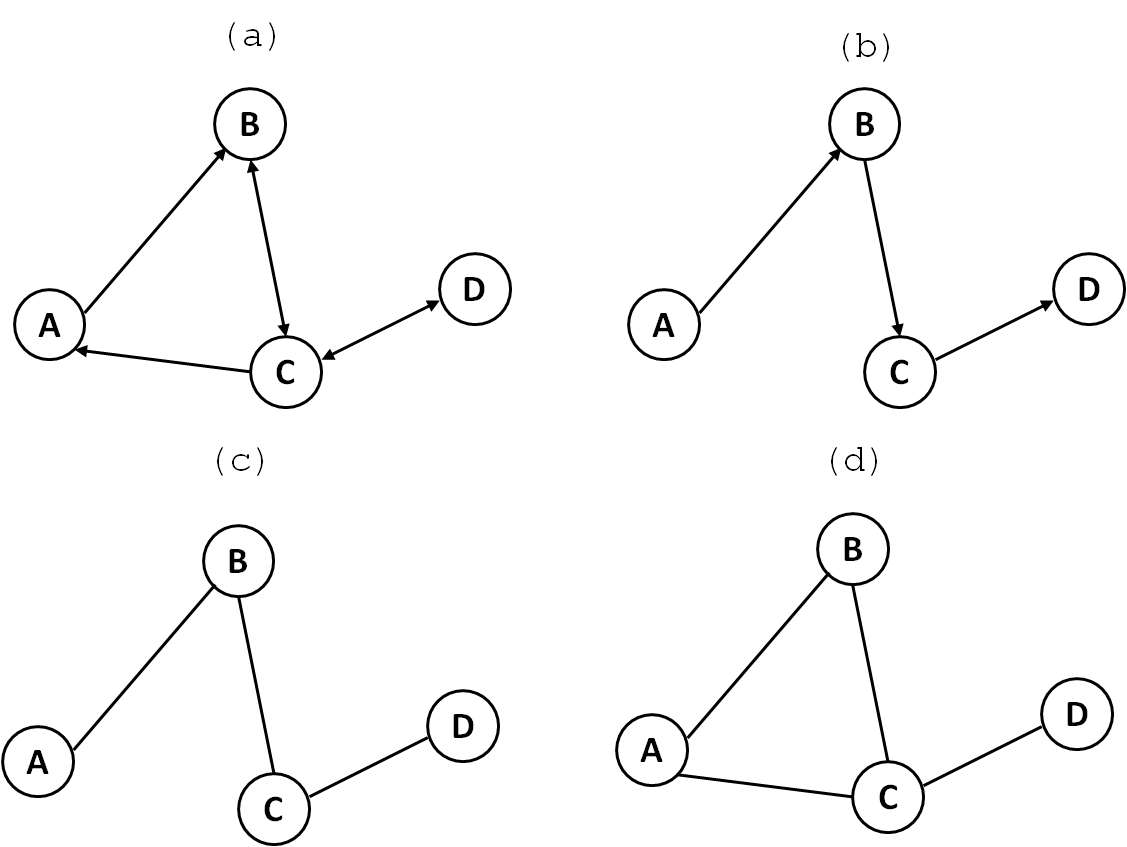
Qu’est-ce qu’un graphe ?

C’est le plan d’exécution d’une requête, d’un traitement. Formellement, c’est un ensemble de vertices qui s’enchaînent selon un ordre précisé par un ensemble d’arcs.

Qu’est-ce qu’un graphe acyclique direct ?

C’est un plan d’exécution non itératif et séquentiel. Les arcs du graphe sont acycliques (séquentiels) et direct (orienté).

Prenons les quatre graphes suivants :



Classez-les selon les indications suivantes (indiquez la lettre à chaque fois) :

|  |  |
| --- | --- |
| Graphe(s) direct(s) | a et b |
| Graphe(s) indirect(s) | c et d |
| Graphe(s) cyclique(s) | a et d |
| Graphe(s) acyclique(s) | b et c |
| Graphe(s) cyclique(s) direct(s) | a |
| Graphe(s) cyclique(s) indirect(s) | d |
| Graphe(s) acyclique(s) direct(s) | b |
| Graphe(s) acyclique(s) indirect(s) | c |

Combien compte-t-on de vertices dans le graphe du MapReduce ?

Le graphe du MapReduce compte trois vertices.

Pourquoi dit-on que le MapReduce est un graphe acyclique direct ?

Son plan d’exécution est orienté, non itératif, direct et s’exécute toujours dans l’ordre suivant : Map > Shuffle > Reduce.

Comment TEZ arrive-t-il à rendre le MapReduce 100 fois plus rapide ?

TEZ optimise le graphe de l’application MapReduce développée par l’utilisateur. Plus précisément, il cherche le chemin le plus court du graphe.

Quelle est la limite majeure de TEZ ?

Il n’est pas itératif. On reste toujours sur un modèle de graphe acyclique direct, inadapté pour les calculs itératifs.

À quelle(s) problématique(s) TEZ répond-il ?

TEZ répond aux problèmes d’analyse interactive.

À quelle(s) problématique(s) Spark répond-il?

Spark répond aux problèmes d’analyse interactive, de jobs itératifs et de streaming.

Que se passerait-il si on utilisait le modèle MapReduce sur un cluster en shared-memory ?

Le MapReduce, de par ses principes, ne peut pas s’exécuter sur un cluster en architecture shared-memory.

Chapitre 3

Qu’est-ce qu’un langage d’abstraction ?

C’est un langage dont la syntaxe est relativement proche du langage humain et permet d’exprimer des problèmes métiers sous forme de requêtes simples. Lorsque l’utilisateur exprime son besoin sous forme d’une requête, cette dernière est transformée par le langage en instructions machine.

Quel est l’avantage d’un langage d’abstraction ?

Un langage d’abstraction facilite la rédaction des requêtes et simplifie la programmation Hadoop pour les utilisateurs métier.

Quel est l’inconvénient d’un langage d’abstraction ?

Un langage d’abstraction ne permet pas d’exprimer des problèmes d’un haut niveau de complexité et est souvent trop lent à cause de la transformation du code en instructions machine.

Cochez celles des propositions suivantes qui sont vraies.

* Plus le niveau d’abstraction offert par un langage est élevé, plus on est éloigné de la machine.
* Plus le niveau d’abstraction offert par un langage est élevé, plus il est simple pour les utilisateurs.
* Plus le niveau d’abstraction offert par un langage est élevé, plus il tend à être fonctionnel.
* Plus le niveau d’abstraction offert par un langage est faible, plus il est difficile pour les utilisateurs.

Hive est-il un langage SQL d’écriture des requêtes sous Hadoop ?

Non.

Hive peut-il exécuter du code Python ?

Oui.

Quel est le rôle de Hive dans l’écosystème Hadoop ?

Il fournit un langage d’abstraction grâce auquel l’utilisateur métier écrit des jobs MapReduce sous forme de script SQL.

Quelle différence faites-vous entre HiveQL et SQL ?

HiveQL est un langage d’abstraction basé sur la spécification ANSI 92 du SQL ; il transforme le script en job MapReduce. Le SQL est nativement le langage de définition et d’interrogation des bases de données relationnelles.

Qu’est-ce qu’une UDF (User Defined Function) ?

C’est une fonction développée dans un langage de script pour étendre les fonctionnalités d’un langage d’abstraction.

Quel est le rôle d’une UDF dans la programmation HiveQL ?

Une UDF sert à développer des fonctions personnalisées pour résoudre des problèmes complexes à rédiger à l’aide du seul HiveQL.

Quel est le rôle du composant Thrift dans l’environnement Hive ?

Thrift transforme en job MapReduce le script HiveQL écrit par l’utilisateur.

Citez deux avantages de Hive.

* Il facilite la rédaction des fonctions MapReduce.
* Il favorise l’adoption d’Hadoop auprès des utilisateurs métiers.

Citez deux inconvénients de Hive.

* Il ne peut pas exprimer facilement des requêtes complexes en raison de sa nature déclarative.
* Il est beaucoup trop lent à cause du temps de conversion du script en job MapReduce.

Quelle différence faites-vous entre HiveQL et Presto ?

HiveQL est un langage d’abstraction ; il transforme les scripts SQL en jobs MapReduce. Presto est nativement un moteur SQL massivement parallèle, qui exécute directement le SQL sur Hadoop.

Soit le fichier de données suivant, stocké sur le HDFS dans le répertoire 'projets/livre\_hadoop/hive/orders.txt'. Utilisez le HiveQL pour répondre aux questions suivantes :

|  |  |  |  |
| --- | --- | --- | --- |
| orderid | customerid | orderdate | qtity |
| 3 | 10007 | 02/06/2011 00:00 | 2 |
| 6 | 10007 | 15/10/2009 00:00 | 1 |
| 9 | 10012 | 15/10/2009 00:00 | 2 |
| 12 | 10005 | 22/11/2010 00:00 | 1 |
| 15 | 10005 | 22/11/2010 00:00 | 1 |
| 18 | 10009 | 22/11/2010 00:00 | 1 |
| 21 | 10011 | 22/11/2010 00:00 | 1 |
| 24 | 10010 | 23/11/2010 00:00 | 1 |
| 27 | 10003 | 23/11/2010 00:00 | 2 |
| 30 | 10006 | 23/11/2010 00:00 | 1 |
| 33 | 10002 | 23/11/2010 00:00 | 1 |
| 36 | 10012 | 23/11/2010 00:00 | 1 |
| 39 | 10004 | 06/04/2010 00:00 | 1 |
| 42 | 10006 | 15/01/2011 00:00 | 1 |

1. Écrivez le script HiveQL de création de la table orders.

USE default;

CREATE TABLE IF NOT EXISTS orders (

orderid STRING,

customerid STRING,

orderdate STRING,

qtity STRING

) ROW FORMAT DELIMITED FIELDS TERMINATED BY '\t',

STORED AS TEXTFILE;

2. Écrivez le script HiveQL d’insertion des lignes de ce fichier dans la table orders que vous avez précédemment créée.

LOAD DATA INPATH 'projets/livre\_hadoop/hive/orders.txt' OVERWRITE INTO TABLE orders;

3. Écrivez le script HiveQL de la somme des quantités par client.

SELECT sum (qtity) FROM orders GROUP BY customerid;

Quelle est la définition de Pig ?

Pig est un environnement d’exécution de flux interactifs de données sous Hadoop.

Quel est le rôle de Pig dans l’écosystème Hadoop ?

Tout comme Hive, c’est un langage d’abstraction.

Quel est le langage offert par Pig pour la programmation des données ?

Pig Latin.

Quel est l’avantage de Pig par rapport à Hive ?

Pig offre un plus bas niveau d’abstraction, ce qui permet d’exprimer des requêtes plus complexes.

Quel est l’inconvénient de Pig par rapport à Hive ?

Pig est plus complexe, donc plus difficile à apprendre.

Reprenez le fichier de données 'projets/livre\_hadoop/pig/orders.txt' précédent. Utilisez le Pig Latin pour répondre aux questions suivantes :

1. Écrivez le script Pig Latin de création de la table orders.

orders = LOAD 'projets/livre\_hadoop/pig/orders.txt' USING PigStorage (‘\t’) AS (orderid:int, customerid:int, orderdate:datetime, qtity:int);

2. Écrivez le script Pig Latin d’insertion des lignes de ce fichier dans la table orders que vous avez précédemment créée.

La première requête répond à cette question. L’instruction LOAD de Pig Latin charge les données du HDFS.

3. Écrivez le script Pig Latin de la somme des quantités par client.

Orders\_group = GROUP orders BY customerid;

somme = FOR EACH orders\_group GENERATE sum(qtity);

DUMP Ventes;

Chapitre 4

Quelle est la différence entre un langage d’abstraction et un moteur SQL sur Hadoop ?

Un langage d’abstraction transforme les scripts SQL en jobs MapReduce, tandis qu’un moteur SQL exécute directement et de façon native le SQL sur Hadoop.

Donnez la définition du SQL.

Le SQL est le langage de définition et d’interrogation des bases de données relationnelles.

Qu’est-ce qu’une base de données parallèle ?

C’est une base de données qui est découpée en partitions réparties entre les nœuds d’un cluster.

Comment les requêtes SQL sont-elles exécutées dans une base de données parallèle ?

Les requêtes SQL sont exécutées de façon concurrente et simultanée sur l’ensemble des nœuds du cluster, qui contiennent une réplique de la base de données parallèle.

Quel est le mécanisme utilisé pour répartir une base de données parallèle dans un cluster ?

Le partitionnement horizontal.

Qu’est-ce qu’un SGBDR MPP (ou moteur SQL MPP) ?

C’est un SGBDR capable d’héberger et de paralléliser l’exécution des bases de données relationnelles sur les nœuds d’un cluster.

Citez un exemple de SGBDR MPP.

GreenPlum

Quelle est la différence entre un moteur SQL et un SGBD ?

Un SGBD gère à la fois la création, l’administration et l’interrogation des bases de données, là où un moteur SQL ne s’occupe que de leur interrogation.

Quelle est la différence entre une architecture client/serveur et une architecture de cluster ?

En client/serveur, les traitements sont faits ex situ, c’est-à-dire que les données sont déplacées pour être traitées où la requête a été exécutée (vers le serveur), tandis que dans un MPP, les traitements sont in situ, c’est-à-dire que les requêtes sont transférées au niveau des nœuds et traitées localement.

Un MPP est linéairement scalable ; les gestions de charge se font simplement par augmentation du nombre de nœuds du cluster, tandis qu’en client/serveur, il faut faire de l’upsizing pour gérer les montées en charge.

Donnez la raison principale pour laquelle une architecture client/serveur n’est pas adaptée pour le traitement SQL massivement parallèle.

Dans une architecture client/serveur, toutes les requêtes sont traitées par une seule machine, le serveur. Avec l’augmentation du nombre de requêtes et l’accroissement du volume de données, le serveur devient rapidement surchargé et crée un goulot d’étranglement sur le réseau.

Sur un SGBDR MPP, toutes les requêtes SQL sont-elles exécutées en parallèle ?

Non.

Tout dépend de la requête. Si elle concerne un filtre de sélection sur une clé primaire par exemple, la requête sera exécutée uniquement par le nœud hébergeant la partition contenant la valeur de la clé interrogée.

Soit la table orders suivante, qui fait partie de la base de données parallèle orders\_system installée sur GreenPlum. Cette base est répliquée sur un cluster de trois nœuds de calcul.

|  |  |  |  |
| --- | --- | --- | --- |
| order\_id | customer\_id | order\_date | quantity |
| 1 | 10006 | 13/10/2009 00:00 | 3 |
| 2 | 10010 | 13/10/2009 00:00 | 1 |
| 3 | 10001 | 02/06/2011 00:00 | 2 |
| 4 | 10003 | 14/10/2009 00:00 | 1 |
| 5 | 10004 | 19/11/2010 00:00 | 1 |
| 6 | 10002 | 15/10/2009 00:00 | 1 |
| 7 | 10001 | 15/10/2009 00:00 | 1 |
| 8 | 10006 | 15/10/2009 00:00 | 1 |
| 9 | 10004 | 15/10/2009 00:00 | 2 |
| 10 | 10004 | 15/10/2009 00:00 | 1 |
| 11 | 10007 | 22/11/2010 00:00 | 1 |
| 12 | 10011 | 22/11/2010 00:00 | 1 |
| 13 | 10007 | 22/11/2010 00:00 | 1 |
| 14 | 10012 | 22/11/2010 00:00 | 1 |
| 15 | 10003 | 22/11/2010 00:00 | 1 |
| 16 | 10012 | 22/11/2010 00:00 | 2 |
| 17 | 10005 | 15/02/2011 00:00 | 1 |
| 18 | 10009 | 22/11/2010 00:00 | 1 |
| 19 | 10012 | 22/11/2010 00:00 | 1 |
| 20 | 10011 | 22/11/2010 00:00 | 3 |
| 21 | 10003 | 22/11/2010 00:00 | 1 |

1. Écrivez la requête SQL de création de la table orders et spécifiez-y qu’elle est distribuée. La clé de partitionnement est le champ order\_id.

CREATE TABLE orders (

orderid INTEGER,

productid VARCHAR(200),

orderdate DATE,

qtity INTEGER

) DISTRIBUTED BY (orderid);

2. Après avoir exécuté la requête, GreenPlum présente les partitions ci-après. À quelle mode d’affectation des partitions ce résultat correspond-il ? Justifiez votre réponse.

|  |  |  |  |
| --- | --- | --- | --- |
| Partition 1 | | | |
| order\_id | product\_id | order\_date | quantity |
| 1 | 10001 | 13/10/2009 00:00 | 3 |
| 4 | 10010 | 14/10/2009 00:00 | 1 |
| 7 | 10007 | 15/10/2009 00:00 | 1 |
| 10 | 10010 | 15/10/2009 00:00 | 1 |
| 13 | 10011 | 22/11/2010 00:00 | 1 |
| 16 | 10001 | 22/11/2010 00:00 | 2 |
| 19 | 10006 | 22/11/2010 00:00 | 1 |
| Partition 2 | | | |
| order\_id | product\_id | order\_date | quantity |
| 2 | 10011 | 13/10/2009 00:00 | 1 |
| 5 | 10004 | 19/11/2010 00:00 | 1 |
| 8 | 10001 | 15/10/2009 00:00 | 1 |
| 11 | 10006 | 22/11/2010 00:00 | 1 |
| 14 | 10012 | 22/11/2010 00:00 | 1 |
| 17 | 10012 | 15/02/2011 00:00 | 1 |
| 20 | 10007 | 22/11/2010 00:00 | 3 |
| Partition 3 | | | |
| order\_id | product\_id | order\_date | quantity |
| 3 | 10003 | 02/06/2011 00:00 | 2 |
| 6 | 10006 | 15/10/2009 00:00 | 1 |
| 9 | 10005 | 15/10/2009 00:00 | 2 |
| 12 | 10009 | 22/11/2010 00:00 | 1 |
| 15 | 10010 | 22/11/2010 00:00 | 1 |
| 18 | 10001 | 22/11/2010 00:00 | 1 |
| 21 | 10011 | 22/11/2010 00:00 | 1 |

Ce partitionnement correspond à l’affectation par hachage, parce que les données sont distribuées aléatoirement sans tenir compte des valeurs de la clé. De plus, les tables sont parfaitement distribuées.

3-Nous écrivons et exécutons la requête suivante : SELECT \* FROM orders WHERE product\_id=10003; Expliquez le processus d’exécution de cette requête.

Le nœud maître du cluster va chercher dans les métadonnées l’index des partitions qui contiennent les lignes avec product\_id=10003. Ensuite, il va transférer la requête uniquement sur les nœuds qui contiennent ces partitions et ceux-ci vont l’exécuter et lui renvoyer les résultats. Dans ce cas de figure, puisque seule la partition 3 contient la valeur 10003, la requête sera envoyée uniquement à ce nœud.

4. Nous écrivons et exécutons la requête suivante : SELECT \* FROM orders WHERE order\_id=1; Expliquez le processus d’exécution de cette requête.

Le processus est similaire à ce que nous avons dit précédemment. La seule différence est que, comme orderid est une clé primaire, elle est dans une seule partition ; donc la requête va s’exécuter uniquement sur le nœud hébergeant la partition contenant la valeur 1. Dans cet exemple, c’est le nœud 1 qui va exécuter la requête.

Cochez la bonne réponse. Dans une base de données parallèle, la compilation des requêtes SQL se fait dans...

... les nœuds de calcul du cluster.

Cochez la bonne réponse. Dans une base de données parallèle, la transformation des requêtes SQL en plan d’exécution se fait dans...

... les nœuds de calcul du cluster.

Qu’est-ce qu’un moteur SQL sur Hadoop (ou moteur SQL MPP sur Hadoop ou moteur natif SQL sur Hadoop) ?

Les moteurs SQL sur Hadoop sont des systèmes capables d’exécuter nativement du SQL tel que défini par la norme ANSI sur un cluster Hadoop (plus précisément sur les données stockées dans un cluster Hadoop).

Citez deux moteurs SQL sur Hadoop de votre choix.

Impala, HAWQ.

Un moteur SQL sur Hadoop est-il un moteur de bases de données relationnelles.

Non.

Qu’est-ce qui différencie un moteur SQL sur Hadoop d’un langage d’abstraction tel que HiveQL ?

Le moteur SQL sur Hadoop exécute nativement du SQL dans le cluster, c’est-à-dire sans transformer le script SQL rédigé par les utilisateurs en MapReduce ou en un quelconque autre modèle de calcul, comme le font HiveQL et les autres les langages d’abstraction.

Citez et expliquez deux différences entre un moteur SQL sur Hadoop et un moteur SQL MPP.

* Là où les SGBDR MPP doivent implémenter des mécanismes propriétaires de tolérance aux pannes, de réplication et de haute disponibilité, les moteurs SQL sur Hadoop s’appuient directement sur le HDFS pour les fournir.
* Les SGDR MPP s’exécutent uniquement sur des tables relationnelles, tandis que les moteurs SQL sur Hadoop s’exécutent sur les données stockées dans le HDFS.

Citez et expliquez les deux aspects majeurs qui gouvernent l’implémentation d’un moteur SQL sur Hadoop.

* Le mode de stockage. Pour faire le traitement in situ, les éditeurs de moteurs SQL sur Hadoop ont le choix entre trois options : s’appuyer directement sur le HDFS pour le stockage, stocker les données dans les nœuds de calcul sur un système différent du HDFS (par exemple un SGBD) ou transformer dynamiquement les données du HDFS en données propriétaires (par exemple en table relationnelle ou en table HBase).
* La structure des données. Dans un cluster Hadoop, les données sont stockées dans le HDFS, sous des formes complexes (JSON, XML, RDF, HTML) ou des fichiers plats. Les éditeurs ici ont le choix entre trois options : reconnaître les formats natifs HDFS, imposer un format propriétaire ou transformer les données de leur structure de base en un format spécifique.

Impala est-il un SGBD MPP ?

Non.

Cochez la bonne réponse. Impala fonctionne sur une architecture...

... asymétrique peer-to-peer.

Citez et donnez le rôle des trois composants principaux d’Impala.

* Les processus Impala, encore appelés impalad (pour Impala Deamon), s’exécutent en permanence sur chaque nœud de données du cluster (un processus impalad par nœud). Ils reçoivent les requêtes SQL transmises par le biais de l’interface de commande Impala, lisent et écrivent les fichiers de données nécessaires, parallélisent et distribuent l’exécution des requêtes dans le cluster et transmettent les résultats intermédiaires des requêtes au Name Node.
* Le Statestore vérifie l’état de fonctionnement de tous les processus Impala du cluster et leur envoie continuellement les informations de cette vérification.
* Le service de catalogue relaie les mises à jour des métadonnées des instructions SQL à tous les nœuds de données du cluster.

Impala supporte-t-il l’exécution des requêtes UPDATE ?

Non.

Le processus impalad d’Impala permet-il de faire les traitements en local ?

Oui.

Quelle est la différence entre HAWQ et Impala ?

HAWQ est un SGBDR MPP qui combine les fonctionnalités d’exécution massivement parallèle du SQL offertes par GreenPlum aux fonctionnalités de réplication et de tolérance aux pannes offertes par le HDFS. En tant que tel, il a l’avantage d’être ACID, ce qui n’est pas le cas pour Impala.

Cochez la bonne réponse. HAWQ est...

... un SGBD MPP.

Dans une architecture HAWQ, les segments sont...

... des nœuds de calcul.

... des processus de calcul.

HAWQ exécute-t-il des jobs MapReduce lorsque vous soumettez des requêtes SQL ? Justifiez votre réponse.

Non. Il exécute du SQL directement sur les données stockées dans le HDFS.

HAWQ peut-il exécuter des procédures stockées ? Justifiez votre réponse.

Oui, parce qu’il est un SGBDR MPP.

Citez et donnez le rôle des trois composants majeurs d’un cluster HAWQ.

* La machine hôte HAWQ est responsable de l’authentification et de l’autorisation, de la transformation des requêtes en plans d’exécution et de la distribution de ces plans aux nœuds du cluster. Elle gère également les métadonnées.
* La machine de calcul ou segment héberge un nœud de données HDFS et plusieurs processus de calcul, appelés les segments, qui tournent en permanence sur la machine et sont responsables de l’exécution des plans des requêtes qu’ils reçoivent.
* Le nœud de référence HDFS est la machine centrale du système de fichiers, qui conserve toutes les métadonnées concernant les données stockées sur le HDFS dans les nœuds de données.

HAWQ fonctionne-t-il avec un cluster Hadoop ?

HAWQ ne fonctionne pas sur un cluster Hadoop (à moins d’utiliser un gestionnaire de ressources comme YARN). Il utilise uniquement le HDFS sur un cluster.

Quelle est la différence entre Phoenix et Hive ?

Phoenix est une couche logicielle installée sur HBase permettant aux utilisateurs d’écrire des requêtes SQL, qu’elle compile en série de commandes scans HBase et dont elle orchestre l’exécution pour produire des jeux de données JDBC. Hive, lui, transforme le script SQL en job MapReduce.

Quelle est la différence entre Phoenix et Impala ?

Impala est un moteur d’exécution SQL MPP sur Hadoop, là où Phoenix est juste une couche d’abstraction qui permet d’exécuter du SQL sur les tables HBase.

Quelle est la différence entre Phoenix et HAWQ ?

HAWQ est un SGBD MPP qui exécute du SQL sur le HDFS. Phoenix, lui, exécute le SQL sur HBase.

Pourquoi dit-on que Phoenix est la « peau SQL de HBase » ?

C’est une couche installée sur HBase qui permet aux utilisateurs d’écrire, non pas des commandes scan ou GET classiques directement, mais des scripts SQL qui sont ensuite transformés en commandes HBase. Cela facilite l’emploi de HBase par les utilisateurs métiers.

Chapitre 5

Qu’est-ce qu’un fait ?

C’est la description de l’état présent d’un objet, de quelque chose.

Qu’est-ce qu’une donnée ?

La donnée est une représentation codifiée et exploitable d’un fait. Cette représentation peut être un symbole, une image, un nombre ou tout autre élément qui pourrait servir de base à la prise de décision. Dans le contexte numérique, une donnée est un chiffre, une image, un objet (vidéo, audio) ou une chaîne de caractères, codé(e) dans un format spécifique permettant son stockage et son traitement par ordinateur.

Quelle est la relation entre une donnée et un fait ?

Le fait décrit un événement, un objet, quelque chose, tandis que la donnée capture ce fait. La donnée est la représentation codée et exploitable du fait. Par exemple, l’ISBN est une donnée qui représente un livre.

Citez et expliquez les trois caractéristiques de la donnée.

* Format d’encodage. C’est la façon dont les données sont représentées. Ici, on s’appuie sur un référentiel de code, par exemple un alphabet, pour spécifier la façon dont les données sont représentées et leur signification.
* Format de stockage. Il fait référence à la façon dont les données sont stockées pour être échangées entre les individus ou les organisations.
* Type. C’est la nature, l’essence de la donnée.

Qu’est-ce que l’encodage d’une donnée ?

C’est la façon dont la donnée est représentée.

Citez et expliquez les deux systèmes d’encodage de données mis au point par l’ISO.

* ASCII : American Standard Code for Information Interchange, dérivé de la norme ISO/CEI 646. L’ASCII définit 128 caractères numérotés de 0 à 127 pour le codage binaire des caractères latins non accentués.
* Unicode, issu de la norme ISO/CEI 10 646. Il définit 128 172 caractères couvrant le codage binaire d’une centaine de langues.

En quoi l’encodage binaire consiste-t-il ?

Il consiste à transformer les données en une suite de 0 et de 1 appelés *bit*.

Qu’est-ce qu’un format de stockage de données ?

Cela désigne la façon dont les données sont stockées pour être échangées entre les individus ou les organisations.

Quelle est la différence entre un format de stockage de données, un format de fichier et une extension ?

Les données sont stockées dans un fichier, selon un format de stockage. À chaque fichier, le système d’exploitation de l’ordinateur associe une extension, grâce à laquelle les programmes reconnaissent ce format de stockage et savent s’ils peuvent traiter les données.

Quelle est la différence entre le HTML et le XML ?

Contrairement au HTML, dans lequel les balises et leur signification sont prédéfinies, le XML fournit la syntaxe du balisage et les utilisateurs s’accordent eux-mêmes sur la structure des données qu’ils vont échanger et, sur la base de cet accord, définissent leurs propres balises. C’est de là que viennent la puissance du XML et son eXtensibilité.

Soit la table suivante. Écrivez sa version JSON.

|  |  |
| --- | --- |
| Id | Nom |
| 1 | Juvénal |
| 2 | Franck |
| 3 | Fabiola |

{ "Id" : [1,2,3],

"Nom" : ["Juvénal","Franck","Fabiola"]

}

Quelle est la différence entre le type de donnée et le type de fichier ?

Le type de fichier est la façon dont les données sont organisées pour être stockées sur le disque dur, tandis que le type de données fait référence au type de raisonnement qui peut leur être appliqué. Les données d’un fichier peuvent avoir le même type ou pas.

Citez et expliquez les trois catégories de formats de fichiers disponibles pour la sérialisation des données.

* Formats structurés. Cette famille regroupe tous les formats de fichiers dans lesquels les données sont visuellement organisées de façon tabulaire, c’est-à-dire sous forme de lignes/colonnes.
* Formats semi-structurés. Cette famille regroupe tous les formats qui fournissent un langage de balisage pour la structuration et l’échange des données.
* Formats non structurés. Cette famille regroupe les formats de fichiers dans lesquels les données ne se prêtent à aucune structure particulière.

Qu’est-ce qu’une ontologie ?

C’est un dictionnaire décrivant complètement l’ensemble des concepts d’un domaine. Par exemple, l’ontologie FOAF, utilisée dans les réseaux sociaux, décrit les personnes, leurs activités et leurs relations avec d’autres personnes.

Quel est le rapport entre une ontologie et le RDF ?

Le RDF est un modèle de description de ressources qui permet à des applications informatiques de communiquer entre elles. L’ontologie est le dictionnaire de l’ensemble de ces ressources. Ainsi, le RDF s’appuie sur une ontologie pour assurer la communication entre les applications.

Quelle est la différence entre le type de donnée dans le sens original et le type de données dans le sens de la programmation informatique ?

Le type d’une donnée en programmation informatique, ou plus précisément le type d’une variable, spécifie le domaine des valeurs que peut prendre cette variable. Le type dans son sens original est la nature de la donnée.

Qu’est-ce qu’un chiffre ?

Le chiffre est un symbole qui sert à représenter graphiquement un nombre selon un système de numérotation. Le système de numérotation est l’ensemble de ces symboles et de règles qui régissent les opérations applicables sur ces nombres et la façon dont elles sont effectuées.

Quelle est la différence entre un chiffre et un nombre ?

Un nombre est un concept qui quantifie une chose ou une situation. Un chiffre représente visuellement cette quantité grâce à l’aide d’un système de numérotation.

Qu’est-ce qu’un alphabet ?

C’est un référentiel de caractères représentant les sons d’une langue émis par des humains appartenant à un groupe sociodémographique et culturel précis.

Quel est le rapport entre un alphabet et un texte ?

Un texte s’écrit toujours en s’appuyant sur un alphabet.

Qu’est-ce qu’une image numérique ?

C’est un ensemble de points converti en fichier binaire. L’image est soit acquise par des convertisseurs analogiques-numériques (scanners, appareils-photo, caméras numériques), soit créée et traitée par des programmes informatiques. Elle est sérialisée dans un fichier sous forme binaire.

Qu’est-ce qu’un pixel ?

Le pixel est un point de couleur ; c’est l’élément de base servant à représenter une couleur dans une image.

Quel est le rapport entre un pixel et une image ?

Les images sont complètement constituées de pixels. Ce sont ces pixels qui sont transformés en bits de valeur 0 ou 1.

Qu’est-ce qu’un codec ?

Le codec, ou codeur-décodeur, discrétise le signal analogique en bits dans un format de fichier particulier et décode ces bits à partir du fichier pour les transformer en ondes.

Qu’est-ce qu’une vidéo ?

Une vidéo est une succession d’images défilant avec une très faible latence (balayage). La vidéo est un cas particulier des images. La cadence est de 25 images/s pour le PAL et 30 images/s pour le NTSC.

Qu’est-ce que la gouvernance de données ?

C’est l’ensemble des politiques et des pratiques qui garantissent l’utilisation adéquate des données dans une organisation.

Quelle est la différence entre la responsabilité de la donnée et la responsabilité de la gouvernance de données ?

La responsabilité de la gouvernance des données est une fonction stratégique et corporative, au même titre que la gouvernance des systèmes d’information. La responsabilité des données est personnelle ; elle renvoie plus au fait que chaque utilisateur doit être garant des données qu’il produit, à la fois d’un point de vue qualitatif et métier.

Qu’est-ce qu’une base de données ?

C’est une représentation d’un processus métier à travers ses données. Techniquement, la base de données est un ensemble de fichiers liés, normalisés, mais gérés et présentés comme une entité (un schéma), pour le stockage et l’exploitation des données.

Quel est le but d’une base de données ?

Le but d’une base de données est de fournir une image claire des activités d’un métier à travers ses données. Cette image claire sera par la suite le socle des opérations et des décisions de l’entreprise.

Qu’est-ce qu’un modèle ?

Un modèle est une représentation simplifiée de la réalité.

Qu’est-ce qu’un modèle de données ?

C’est la représentation des données d’un métier à l’aide d’un formalisme ou méta-modèle.

Quelle est la relation entre un modèle de données et une base de données ?

Le modèle de données est utilisé pour représenter la cartographie d’une base de données.

Qu’est-ce qu’un formalisme ?

Un formalisme est un ensemble de concepts, de notations et de relations entre les concepts permettant de construire une image abstraite d’un modèle. Il est généralement assorti d’une définition et il est visualisé par un symbole graphique qui est sa notation.

Quelle est la relation entre un modèle, un formalisme et un diagramme ?

Le diagramme est la représentation graphique des éléments d’un modèle à l’aide d’un formalisme.

Quelle est la différence entre une application de données et une base de données ?

Une application de base de données est une application informatique qui répond à un besoin précis en faisant appel à une ou plusieurs bases de données.

Quelle est la différence entre une base de données et un tableur ?

Une base de données capture les données d’un processus métier et les différents liens qu’elles entretiennent. Le tableur exploite les données d’une base ; même s’il les stocke, il ne capture pas les relations entre elles. De plus, la base de données est indépendante des applications qui l’exploitent, contrairement au tableur.

Qu’est-ce qu’un SGBD ?

C’est un logiciel qui gère le stockage et l’exploitation de bases de données.

Quel est le but principal d’un SGBD ?

Un SGBD garantit l’indépendance entre les bases de données et les applications qui les exploitent.

Le SGBD permet-il à la fois de gérer et d’exploiter les données ?

Oui.

Citez et expliquez trois caractéristiques des SGBD.

* Indépendance entre les données et les applications qui les exploitent. Un SGBD doit permettre à un utilisateur de développer une application sans avoir à encoder dans celle-ci les aspects structurels des fichiers de la base de données.
* Contrôle centralisé des données pour éviter toute redondance. Le SGBD doit résoudre le problème des « silos de données ». Autrement dit, il doit centraliser toutes les données de l’entreprise et intégrer dans un même espace de stockage plusieurs fichiers, de manière à éviter toute redondance.
* Gestion de la cohérence et de l’intégrité des données. Le SGBD doit pouvoir gérer le processus de mise à jour d’un ensemble de données comme un tout indissociable pour garantir la « cohérence interne » de la base de données.

Qu’est-ce qu’un SGBDR ?

C’est un logiciel qui gère l’implémentation et l’exploitation des bases de données dites relationnelles.

Quelle est la particularité des SGBDR ?

Les SGBDR ont tous en commun le langage SQL, qui leur sert à la fois pour la définition des données et pour leur exploitation.

Quelle est la particularité d’une base de données relationnelle ?

Dans une base de données relationnelle, la structure d’un fichier est définie comme une relation (communément une table) entre les données provenant d’un nombre fini de domaines (communément des champs). Les enregistrements sont des tuples (des enregistrements ou lignes) et constituent des occurrences de la relation.

Quel est le rapport entre le SQL et les SGBDR ?

Le SQL est un langage déclaratif qui permet aux utilisateurs d’exploiter les bases de données relationnelles des SGBDR.

Expliquez deux limites des SGBDR.

* L’augmentation du volume des données augmente la latence des requêtes.
* Les SGBDR ne gèrent que les données de format tabulaire.

Quelle est l’approche conceptuelle des SGBD NoSQL ?

Dans un SGBD NoSQL, c’est une approche non relationnelle qui est adoptée. Elle consiste en deux choix :

* L’unité logique de stockage n’est plus la relation (ou la table), mais l’agrégat ou le nœud.
* Les contraintes d’intégrité référentielles et sémantiques sont complètement relâchées.

Ces deux choix facilitent le partitionnement et la distribution du stockage des données dans un cluster.

Dans un SGBD NoSQL, qu’est-ce qu’une base de données ?

C’est un agrégat (à l’exception des bases de données graphes, dont l’unité de stockage est le nœud). L’agrégat est une collection d’objets (images, vidéos, documents ou tout autre donnée) référencés par une clé.

Quel est le but des systèmes NoSQL ?

Ils doivent assurer la distribution du stockage et du traitement des données sur un nombre théoriquement illimité de nœuds d’un cluster.

Pourquoi est-il nécessaire de relâcher les contraintes de données dans les SGBD NoSQL ?

Le relâchement des contraintes favorise le partitionnement des données, qui favorise à son tour leur distribution dans le cluster.

Citez et expliquez trois problèmes que pose la construction d’un SGBD NoSQL.

* Cohérence et intégrité des transactions. Les systèmes NoSQL dupliquent les données, ce qui pose un problème d’intégrité et d’unicité. Si on enregistre une donnée plusieurs fois, sur plusieurs nœuds d’un cluster, cet enregistrement ne sera pas forcément atomique, cohérent, isolé et durable.
* Distribution des données. Les systèmes NoSQL sont contraints à relâcher les contraintes d’intégrité. Ceci entraîne une forte redondance des données, qui aura un impact sur la performance des requêtes.
* Absence de schéma de données. Dans les SGBD NoSQL, il est du ressort du développeur de coder à la fois la logique applicative et la couche « cohérence de données » dans l’application. Ceci peut rendre difficile soit la mise à jour de l’application, soit la modification de la structure de la base de données.

Les SGBD NoSQL sont-ils portables ? Justifiez votre réponse.

Non, les SGBD NoSQL ne sont pas portables, en raison de l’absence de schéma de données (voir question précédente). Ce problème de portabilité est en partie dû au manque d’un standard commun défini à la fois pour le stockage et les requêtes.

Quel est le facteur qui a favorisé la domination des SGBDR dans le marché ?

Les SGBDR sont depuis longtemps le standard pour l’organisation, le stockage et l’exploitation des données pour deux raisons :

* capacité à fournir des mécanismes ACID de cohérence des transactions ;
* présence du SQL, garantissant la portabilité des applications entre deux SGBDR.

Quelle est la différence entre un SGBD orienté-colonnes et un SGBD NoSQL orienté-colonnes ?

Un SGBD orienté-colonnes est un moteur de traitement analytique de type OLAP, dans lequel les données des colonnes d’une table sont sauvegardées dans la même position sur le disque dur. Les SGBD NoSQL orientés-colonnes sont plus des entrepôts que des moteurs de base de données.

Qu’est-ce qu’un graphe ?

Un graphe est un ensemble de sujets-prédicats-objets.

Quelle est la particularité d’un SGBD orienté-graphes par rapport à d’autres catégories de SGBD NoSQL ?

Comparativement à tous les autres moteurs, qui regroupent toutes les données autour d’une table ou d’un agrégat, les moteurs orientés-graphes les éclatent autant que possible sous forme de nœuds interconnectés.

Un SGBD orienté-graphes peut-il distribuer le stockage du graphe sur les nœuds d’un cluster ? Justifiez votre réponse.

Non, à cause de la forte connectivité du graphe.

Chapitre 6

Quelle différence faites-vous entre un SGBD distribué et un SGBD centralisé ?

Un SGBD distribué gère le stockage et l’exploitation des données sur plusieurs nœuds d’un cluster de machines, tandis qu’un SGBD centralisé centralise le stockage et le traitement des données sur un seul serveur.

Cochez la(les) proposition(s) exacte(s).

* HBase est un SGDB distribué.
* Son unité physique de stockage est le fichier HFile.
* HBase ne peut pas fonctionner sans ZooKeeper.

Donnez la définition de HBase.

HBase est un SGBD distribué, orienté-colonnes, qui fournit l’accès en temps réel, aussi bien en lecture qu’en écriture, aux données stockées sur le HDFS.

Quelle est la différence entre HBase et le HDFS ?

HBase est un SGBD, tandis que le HDFS est un système de fichiers. HBase exploite les données stockées sur le HDFS et celui-ci le considère comme un client à qui il fournit des données.

Peut-on modéliser le schéma d’une base de données HBase en utilisant un MCD ? Justifiez votre réponse.

Non. En HBase, le concept de base de données n’est pas le même que dans l’approche relationnelle. Modéliser une base de données relationnelle avec un MCD a pour but de réduire la redondance et d’assurer la cohérence. En HBase, on s’intéresse plus au partitionnement et à la distribution des données sur un cluster.

En quoi la modélisation en HBase consiste-t-elle ?

La modélisation HBase consiste à concevoir et implémenter un ensemble de tables HBase. Ici, on se préoccupe d’assurer la distribution du stockage et de faciliter l’accès aux données dans un cluster pour les traitements parallèles.

Qu’est-ce qu’une table HBase ?

Une table HBase est un tableau multidimensionnel de données distribué et sauvegardé sur le HDFS sous forme de fichiers spécifiques appelés HFiles.

Citez et donnez le rôle des cinq dimensions qui constituent la structure logique de la table HBase.

* Row key (clé de ligne). Chaque ligne du tableau HBase est identifiée de façon unique par cette clé.
* Famille de colonnes (column family), d’où HBase tire son nom de SGBD orienté-colonnes. Il s’agit des valeurs d’un ensemble de colonnes physiquement colocataires, c’est-à-dire physiquement sérialisées (stockées) dans le même ficher.
* Colonne (column qualifier). Une colonne est l’adresse d’une série de données dans une famille de colonnes.
* Cellule (cell) ou valeur (pour faire référence au contenu de la cellule). Une cellule HBase est l’intersection entre une row key, une famille de colonnes et une colonne. Les données sont stockées dans les cellules.
* TimeStamp. Il sert à distinguer les versions d’une même ligne à l’aide d’une valeur horodatée.

Quel est le but de la modélisation du schéma d’une table HBase ?

Le but est d’assurer le partitionnement et la distribution des tables HBase sur un cluster pour faciliter les traitements parallèles qui vont se faire dessus.

Cochez les propositions exactes parmi les suivantes :

* Les données sont stockées sous le type Byte[]. Les colonnes ne sont pas typées.
* Les familles de colonnes sont statiques et sont définies à la création de la table.
* Les colonnes sont dynamiques et sont définies à l’ajout de ligne.

Les tables HBase peuvent-elles être reliées entre elles au moyen des contraintes d’intégrité référentielles comme dans le SGBDR ?

Non.

Cochez la bonne réponse. Une base de données HBase est...

* Aucune de ces propositions. Le concept de base de données n’existe pas en HBase.

Cochez la bonne réponse. La modification d’une ligne en HBase...

* Conserve les anciennes valeurs et crée une nouvelle version de la même ligne.

Donnez le rôle des composants suivants dans le cluster HBase.

* Le HMaster est le nœud maître du cluster HBase. Il gère les métadonnées des tables et coordonne l’exécution des activités des Region Servers.
* Les Region Servers sont les nœuds de calcul et de stockage du cluster HBase. Ils stockent les données des tables et y effectuent les opérations de lecture/écriture de données.
* ZooKeeper est installé sur un cluster indépendant de celui de HBase et assure la coordination entre les clients HBase et les Region Servers.

Cochez la bonne réponse. Lorsqu’un client HBase envoie une requête de lecture de données à HBase, qui la reçoit ?

* ZooKeeper

Expliquez le mécanisme utilisé par HBase pour distribuer le stockage des tables de données à travers le cluster.

HBase découpe les tables en « régions ».

Par quel mécanisme HBase supprime-t-il les données d’une table ?

HBase supprime les données marquées lors du compactage de la table.

Soit la requête HiveQL suivante. Modifiez-la pour créer la table en HBase.

CREATE TABLEcustomer\_db.orders (

order\_id INTEGER,

product\_id INTEGER,

order\_date DATE,

quantity INTEGER

)

La requête devient :

CREATE TABLE IF NOT EXISTS customer\_db.orders (

order\_id INTEGER,

product\_id INTEGER,

orderdate DATE,

quantity INTEGER

)

STORED BY 'org.apache.hadoop.hive.HBaseStorageHandler'

WITH SERDEPROPERTIES ("hbase.columns.mapping" = "order\_id:key, produit:product\_id, sales:orderdate, sales:quantity)

TBLPROPERTIES ("hbase.table.name" = "orders");

Que fait la requête HBase suivante ?

scan 'Customers', {COLUMNS => ['personal:customer\_name', 'public:enterprise'], VERSIONS => 10};

Cette requête renvoie les données des dix dernières versions des colonnes customer\_name et entreprise.

Que fait la requête HBase suivante ? Cochez la bonne réponse.

delete 'Customers', 'CC09877', 'personal:customer\_name', '1209201600002345'

Elle supprime la version 1209201600002345 de la colonne customer\_name de la ligne CC09877.

Chapitre 7

Quelle différence faites-vous entre une requête SQL et une recherche de contenu ?

La requête SQL renvoie les lignes contenant de manière stricte l’ensemble des mots-clés recherchés par l’utilisateur. Une recherche de contenu permet à l’utilisateur de trouver un contenu d’intérêt juste à partir d’un extrait de mots-clés ou d’informations partielles ; les résultats sont inexacts et n’ont pas besoin d’adhérer strictement à un modèle de données. Un score d’importance est utilisé à la place d’une simple variable binaire « présent/absent ».

Qu’est-ce qu’un index ?

Un index est une liste ordonnée de mots-clés.

Que représente la matrice d’incidence des termes ?

C’est une matrice qui trace pour chaque mot son apparition ou non dans un ensemble de documents.

Qu’est-ce qu’un corpus ?

C’est l’ensemble des documents sur lesquels le moteur effectue les recherches de contenu.

Quelle est la différence entre l’index classique et l’index inversé ?

L’index inversé est une collection de listes, une par terme, indiquant l’ensemble des documents contenant ce terme. L’index simple indique de façon binaire la présence ou l’absence de termes dans une collection de documents.

Citez deux inconvénients de l’utilisation de l’index simple dans la recherche de contenu.

* L’enregistrement de l’absence des termes dans les documents gaspille de l’espace.
* L’index simple fait l’hypothèse que la langue des mots est connue.

Qu’est-ce qu’un index inversé ?

L’index inversé est une collection de listes, une par terme, indiquant l’ensemble des documents contenant ce terme. Il fournit la fréquence d’apparition des termes d’un dictionnaire à l’intérieur d’une collection de documents.

Qu’est-ce que le dictionnaire ?

C’est la collection des termes d’un index inversé.

Citez deux avantages de l’index inversé dans la recherche de contenu ?

* L’index inversé prend moins de place que l’index simple.
* Il prend en compte toutes les langues et toutes les représentations de la donnée sur Internet.

Qu’est-ce qu’un mot vide *(stop word)* ?

C’est un mot qui n’est d’aucune utilité pour la recherche et est ignoré lors de celle-ci. Par exemple, en français, des mots tels que « un », « et », « est » ou « le » sont considérés comme vides de sens.

Quel est l’impact des mots vides dans la construction du dictionnaire de l’index ?

Les mots vides forment la structure sémantique des phrases qui vont être indexées et sont par conséquent très nombreux. Ils ne peuvent pas être ajoutés dans le dictionnaire de l’index.

Expliquez le mécanisme d’indexation de contenu dans les SGBDR.

Dans les SGBDR, la clé primaire de chaque table est indexée et la recherche se fait à l’aide de fonctions fournies par la norme SQL (CONTAINS, LIKE, OR, SIMILAR...) sur les lignes de la table.

Quels sont les trois inconvénients majeurs de l’indexation de contenu dans les SGBDR ?

* Les résultats renvoyés par les SGBDR ne tiennent compte ni de l’intention véritable de l’utilisateur, ni de la sémantique des termes. C’est à l’utilisateur d’être le plus précis possible dans sa requête, ce qui n’est pas toujours pratique et peut être très pénible.
* Les recherches dans les SGBDR ne peuvent se faire que sur les données structurées (les tables).
* La performance des requêtes SQL baisse avec l’augmentation du volume de données.

Qu’est-ce qu’une recherche textuelle intégrale ?

De l’expression anglaise *full text search*, il s’agit d’une recherche linguistique sur des données textuelles, dans des documents structurés ou non, en traitant les mots et les expressions à partir des règles spécifiques d’une langue.

Quelle est la différence entre une recherche textuelle intégrale et une recherche à l’aide de l’instruction LIKE du SQL ?

Une recherche textuelle intégrale est bien plus large qu’une recherche à l’aide de l’instruction LIKE. Cette dernière retourne, pour chaque ligne de la table, une valeur booléenne indiquant strictement si le motif de recherche est trouvé dans la ligne. La recherche textuelle intégrale prend en compte l’intention de l’utilisateur et les spécificités de la langue. C’est un score de correspondance qui est utilisé pour classer les résultats, alors qu’avec le LIKE, c’est un critère binaire (présence/absence du terme) qui est utilisé.

Quels résultats la requête \*cadre\* retournera-t-elle ?

Elle renverra toutes les lignes dont les mots contiennent « cadre », par exemple « encadrer », « encadrement », « recadrer », etc.

Qu’est-ce que l’extraction d’entité ?

C’est une technique du traitement du langage naturel qui permet de détecter et d’étiqueter des entités du texte. Une entité peut être un nom propre de personne, un nom de pays, un nom de produit, etc.

En quoi le N-gramme consiste-t-il ?

C’est une technique qui consiste à sectionner les longues chaînes de caractères en chaînes de taille fixe N, qui sont ensuite ajoutées au dictionnaire de l’index pour effectuer les recherches.

Quelles sont les chaînes fournies par le 5-gramme du terme « Anticonstitutionnellement » ?

Antic, ntico, ticon, icons, const, onsti, nstit, stitu, titut, ituti, tutio, ution, tionn, ionne, onnel, nnell, nelle, ellem, lleme, lemen, ement ()

Expliquez en quoi consiste la recherche géographique ou géolocalisation.

C’est un type de recherche qui consiste à déterminer précisément la position d’un ou plusieurs points par rapport à un point de référence.

Quelle est la différence entre le partitionnement par terme et le partitionnement par document ?

Dans le partitionnement par terme, l’index est divisé en partitions des mots du dictionnaire, qui sont distribuées entre les nœuds. Dans le partitionnement par document, ce sont les documents indexés qui sont distribués et chaque nœud en reçoit à peu près le même volume.

Expliquez les deux modes de stockage d’un index dans un cluster.

* Stockage en mémoire
* Stockage sur disque dur

Expliquez les deux composants qui constituent une solution d’indexation de contenu.

* Moteur de stockage des données. C’est le composant chargé du stockage des documents sur lesquels va se faire la recherche de contenu.
* Moteur d’indexation de contenu. C’est le composant qui est chargé d’indexer les documents, de construire l’index inversé et de fournir des fonctions pour la recherche de contenu dans ces documents.

Qu’est-ce qui différencie un moteur NoSQL orienté-colonnes d’un moteur NoSQL orienté-plein texte ?

À l’inverse des orientés-colonnes, les moteurs NoSQL orientés-plein texte ne sont pas des moteurs de stockage. Ils indexent le contenu des documents ou des bases de données stockées dans ces moteurs et fournissent des fonctions de recherche de contenu sur leurs données.

Expliquez les deux modes architecturaux des moteurs d’indexation de contenu.

* Architectures à index distant. Les index sont stockés séparément de la base de données. Le moteur d’indexation de contenu est installé sur un cluster séparé de celui du moteur de la base et agit comme un « service » (ou API) de recherche distant ».
* Architectures à index local. Les index sont stockés dans le même cluster que la base de données. Le moteur NoSQL d’indexation de contenu est intégré comme une fonctionnalité supplémentaire au moteur de base de données.

Chapitre 8

Qu’est-ce qu’une recherche de contenu ?

Cela consiste à trouver un contenu d’intérêt dans une collection de documents, à partir d’extraits de mots-clés ou d’informations partielles.

Qu’est-ce qu’un index et quel est son avantage ?

L’index est une collection de listes, une par terme, indiquant l’ensemble des documents contenant ce terme. Il indique l’ensemble des documents dans lesquels se trouvent les termes d’intérêt sans avoir à balayer linéairement toute la base documentaire.

Quels sont les deux éléments qui constituent un index inversé ?

* Le dictionnaire des termes
* La base documentaire

Quelle est la relation qui existe entre une application de recherche de contenu et un moteur d’indexation de contenu ?

L’application s’adosse sur un moteur d’indexation pour fournir à ses utilisateurs des fonctionnalités avancées de recherche de contenu.

Citez et expliquez les deux approches de recherche disponibles actuellement.

* Classification manuelle des documents en catégories hiérarchiques
* Indexation de contenu

Citez les six composants qui forment l’architecture d’une application de recherche de contenu.

* Index
* Module d’indexation de contenu
* Module de recherche de contenu
* Interface utilisateur
* Interface analytique
* Interface administrative

Décrivez le résultat de la requête SQL suivante :

SELECT \* FROM livre WHERE description LIKE ‘%hadoop%’;

Cette requête renvoie tous les livres dont la description contient le terme « Hadoop ».

Que mesure la précision d’une requête de recherche de contenu ?

Elle mesure son degré d’efficacité, en d’autres termes le niveau avec lequel les résultats renvoyés correspondent à l’intention véritable de recherche de l’utilisateur.

Quelles sont les trois phases de développement d’une application de recherche de contenu ?

* Construction de l’index
* Construction du module de recherche
* Construction de l’interface utilisateur.

Qu’est-ce qu’un crawler ?

C’est un robot qui récupère le contenu spécifique des documents, par exemple les données des balises spécifiques d’une page web.

Quel est le but d’un crawler dans le développement d’une application de recherche de contenu ?

Il permet d’acquérir le contenu qui va être utilisé pour construire l’index, plus précisément le dictionnaire de l’index.

Citez deux bibliothèques qui peuvent être utilisées pour développer des crawlers.

* BeautifulSoup
* Selenium

Des huit activités suivantes, cochez celles qui sont réalisées manuellement par le développeur de l’application de recherche de contenu.

* Acquisition de document
* Structuration de contenu
* Construction de requête
* Interface utilisateur

Des huit activités suivantes, cochez celles qui sont réalisées par Apache Lucene.

* Acquisition de document
* Structuration de contenu
* Construction de requête
* Interface utilisateur

Quelle définition donneriez-vous de Lucene ?

Lucene est un ensemble de classes Java qui fournissent les fonctions et les procédures nécessaires uniquement pour les tâches d’indexation et de recherche de contenu textuel.

Un moteur de recherche est-il une application de recherche ? Justifiez votre réponse

Non. Le moteur de recherche fournit les outils pour l’indexation et la recherche de contenu. L’application utilise les fonctionnalités de ce moteur de recherche.

Quelle est la différence entre Lucene et un moteur de recherche de contenu ?

Lucene est à la base d’un moteur de recherche de contenu. C’est l’ensemble des classes qui permettent de développer toute application de recherche de contenu. Le moteur est en ce sens une application de Lucene.

Quel est l’avantage de Lucene par rapport aux autres moteurs d’indexation de contenu ?

Étant donné que Lucene est une bibliothèque de classes, aucune hypothèse n’est faite a priori sur les index qui peuvent être construits. Les autres moteurs d’indexation de contenu disposent déjà de modèles sur lesquels il faut s’appuyer pour construire son index.

Cochez la(les) bonne(s) réponse(s).

* Lucene est une bibliothèque de classes.
* Lucene permet de développer des applications de recherche de contenu.

Vous pouvez utiliser Lucene pour :

* Ajouter des fonctionnalités d’indexation et de recherche de contenu dans vos applications.

En développement d’application de recherche de contenu, qu’est-ce qu’une unité ?

C’est la base de documents qui va être indexée. Elle n’est jamais la même que la base initiale, parce que les documents à indexer peuvent provenir de différentes sources et être structurés différemment.

Quelle est l’unité en Lucene ?

L’unité en Lucene est le document. Un document Lucene est tout fichier de contenu textuel.

En Lucene, un index consiste-t-il en un ou plusieurs documents ?

Oui.

Pouvez-vous ajouter Lucene à un SGBDR pour améliorer l’efficacité des requêtes SQL ?

Oui.

Soit une bibliothèque numérique. Nous sollicitons votre aide pour lui ajouter des fonctionnalités d’indexation et de recherche de contenu en utilisant Apache Lucene. Répondez aux questions suivantes.

Quelles sont les données à indexer ?

Les données à indexer sont les livres.

Citez les trois étapes nécessaires et les classes Lucene respectives à utiliser pour développer le module d’indexation de contenu de l’application.

1. Structuration de contenu, avec les classes Document et Field

2. Analyse de contenu, avec la classe Analyzer

3. Indexation de contenu avec la classe IndexWriter

Citez les trois étapes nécessaires et les classes Lucene respectives à utiliser pour développer le module de recherche de contenu de l’application.

1. Construction de requête avec les classes Query et QueryParser

2. Recherche de contenu avec la classe IndexSearcher

3. Restitution des résultats de recherche avec la classe Hits

Chapitre 9

Quelle est la différence entre ElasticSearch et un SGBDR ?

ElasticSearch est un moteur NoSQL auquel ont été ajoutées des fonctionnalités de recherche et d’indexation de contenu. Un SGBDR est un moteur spécialisé uniquement dans la gestion des bases de données relationnelles.

Quelle est la différence entre Lucene et ElasticSearch ?

ElasticSearch s’appuie sur Lucene pour fournir les fonctionnalités de recherche et d’indexation de contenu. Là où Lucene ne gère pas le stockage de l’index, ElasticSearch permet de stocker les documents et de les indexer directement grâce à Lucene.

Quelle est la différence entre un index Lucene et un index ElasticSearch ?

En ElasticSearch, l’index est l’ensemble des documents JSON stockés dans le moteur. En Lucene, l’index est un objet de classe Document qui représente l’ensemble des documents contenant du texte.

Donnez deux avantages d’ElasticSearch devant Lucene.

* ElasticSearch stocke les documents dans son moteur, ce qui permet d’y faire directement l’indexation et la recherche de contenu.
* Développer une application de recherche de contenu en ElasticSearch est plus simple qu’en Lucene. Il n’y a pas de développement Java, juste la manipulation d’une API REST à l’aide des verbes HTTP tels que GET, PUT, DELETE, etc.

Un nœud ElasticSearch correspond à :

... Une instance d’exécution d’ElasticSearch.

Un cluster ElasticSearch peut-il être contenu dans une seule machine ?

Oui.

Citez les trois concepts sur lesquels ElasticSearch s’appuie concernant le stockage des données.

* Le document
* Le type
* L’index

Pourquoi dit-on qu’ElasticSearch est sans schéma ?

Les documents JSON qui sont stockés et indexés n’ont pas besoin d’avoir la même structure.

Quelle est la différence entre Apache Solr et ElasticSearch mentionnée dans le chapitre ? En quoi est-elle importante pour le choix entre les deux outils ?

ElasticSearch est sans schéma, ce qui n’est pas le cas d’Apache Solr où les documents doivent tous de manière stricte adhérer à une même structure. L’un des problèmes majeurs de l’indexation de contenu est la variété des formats de fichier ; pouvoir supporter tous ces types facilite grandement le développement des applications d’indexation de contenu.

Qu’est-ce qu’un type dans ElasticSearch ?

Encore appelé *mapping*, le type fait référence à un ensemble des documents JSON qui partagent des propriétés communes.

À quel niveau Lucene intervient-il dans le fonctionnement d’ElasticSearch ?

Chaque partition des documents est un index Lucene. Une instance de Lucene est affectée à chaque partition et ses répliques ; lorsque les requêtes des utilisateurs arrivent dans le système, chaque instance de Lucene effectue la recherche de contenu sur sa partition.

Quel est le rôle de Lucene dans ElasticSearch ?

Le rôle de Lucene est de fournir l’ensemble des fonctionnalités pour rechercher du contenu dans les documents JSON stockés dans ElasticSearch.

Qu’est-ce qu’un indice ?

L’indice est le conteneur logique de l’ensemble des documents qui ont le même type.

Qu’est-ce qu’une partition et quel est son rôle dans le fonctionnement d’ElasticSearch ?

Une partition est un bloc de documents JSON. Le partitionnement des documents rend l’indexation de contenu scalable et distribuée dans le cluster ElasticSearch. De plus, il assure la haute disponibilité du cluster.

Qu’est-ce que le REST ?

REST (REpresentational State Transfer) décrit un style d’architecture logicielle pour développer une application orientée service fonctionnant en environnement client/serveur. C’est un ensemble de bonnes pratiques à respecter pour développer des services.

Des six propositions suivantes, cochez celles qui décrivent au mieux le REST.

... Un système d’architecture

... Une approche pour construire une application

... Un standard

Qu’est-ce qu’un protocole ?

Un protocole est un standard de communication entre des processus s’exécutant sur différentes machines, c’est-à-dire un ensemble de règles et de procédures à respecter pour émettre et recevoir des données entre plusieurs machines à travers un réseau.

Soit une base de données relationnelle partagée dans une architecture client/serveur. Répondez aux questions suivantes.

1. Quelle est la ressource partagée ?

La base de données.

2. Citez l’un des protocoles utilisables pour que clients et serveur communiquent.

ODBC

Quelle est la différence entre un protocole de communication (HTTP) et un protocole de transfert de données (le TCP) ?

Le protocole de communication définit la façon dont un client et un serveur s’échangent les données. Le protocole de transfert de données spécifie la façon dont ces données transitent via un réseau.

Quelle est la différence entre une architecture distribuée de type client/serveur et une architecture distribuée de type cluster ?

Dans une architecture client/serveur, la charge de calcul est centralisée sur la machine serveur. Dans une architecture de type cluster, la charge est répartie à peu près équitablement entre les nœuds du cluster.

Qu’est-ce qu’une requête HTTP ?

C’est une action du protocole HTTP qui spécifie à un serveur HTTP l’action à mener sur une page web identifiée par une URL. Une requête HTTP a la forme suivante : action + URL.

Que signifie urbaniser un système d’information ?

Urbaniser un système d’information revient à le découper en zones interdépendantes.

Quelle est la différence entre une API et un protocole ?

Un protocole est une spécification, un langage de communication entre plusieurs ordinateurs ou applications. Une API expose les méthodes qui leur permettent de communiquer. Dans une démarche orientée service, l’application cliente et le fournisseur sont capables de communiquer uniquement si le client est compatible avec le protocole du fournisseur du service.

Des huit activités suivantes, cochez celles qui sont réalisées par ElasticSearch.

... Structuration de contenu

... Analyse de contenu

... Indexation de contenu

... Construction de requête

... Exécution de requête

Quel est le but d’une démarche SOA ?

Une démarche SOA vise à réorganiser le système informatique d’une entreprise en services interopérables.

En Big Data, quels sont les avantages d’une démarche SOA ?

La démarche SOA est très efficace pour répondre aux besoins d’interopérabilité et de réutilisabilité de différentes applications. En Big Data, elle permet de développer des applications qui exploitent des fonctionnalités exposées par d’autres outils. Ainsi, cette démarche favorise le développement modulaire d’applications.

Qu’est-ce qu’un service ?

C’est l’ensemble des fonctionnalités d’une application qui sont exposées pour consommation par d’autres applications par le biais d’une API.

Qu’est-ce qu’une API ?

C’est un contrat d’utilisation de service, une interface qui expose les fonctionnalités consommables d’un service.

Pourquoi dit-on qu’une API est un contrat ?

Une API délimite l’usage du service en exposant uniquement les méthodes qui peuvent être consommées par le client.

Quelle est la différence entre une API et un protocole de communication ?

Le protocole de communication permet à deux processus de s’échanger les données. L’API est le canal par lequel cet échange se fait. Pour que les processus puissent s’échanger les données, il faut qu’ils soient compatibles avec le même protocole.

Quel est le rôle du protocole dans l’utilisation des services ?

Le protocole assure que les clients et le fournisseur du service peuvent s’échanger les données.

Quel est le rôle d’une API dans un service ?

L’API est l’interface par le biais duquel les clients peuvent spécifier au fournisseur ce qu’il doit faire.

Qu’est-ce qu’une méthode ?

C’est une action qui peut être utilisée par le client pour envoyer des requêtes au fournisseur.

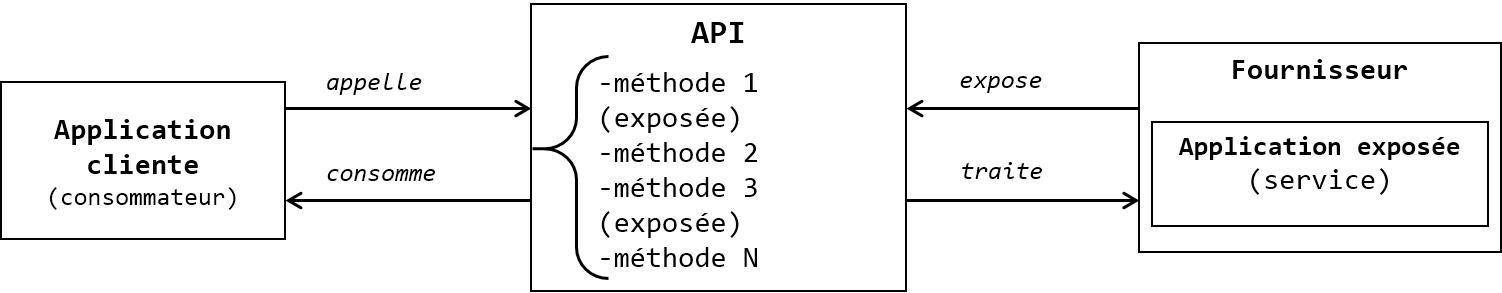
Quel est le rôle d’une méthode dans un service ?

Les méthodes permettent d’appeler les fonctionnalités du service.

Donnez la signification de l’expression « *un service expose ses méthodes à travers une API* ».

Cette expression signifie que les fonctionnalités du service peuvent être utilisées à travers une API.

Dans le schéma suivant, la méthode 2 peut-elle être appelée par l’application cliente ?



Non.

Comment utilise-t-on un service ?

On appelle les méthodes que l’API met à disposition.

Cochez la bonne réponse. Lors de l’utilisation d’un service...

... L’application cliente consomme les résultats des fonctionnalités du service.

Cochez la bonne réponse. Lors de l’utilisation d’un service, les méthodes appelées sont...

... Exécutées par l’application fournisseur.

Qu’est-ce qu’un service RESTful ?

Ce sont des services développés selon l’architecture REST.

Décrivez le résultat de la requête suivante :

curl -XPOST 'http://localhost:9200/clients/adresse/\_search' –d '

{

"query": {

"match": {"nom": "juvénal"}

}

}'

Cette requête renvoie tous les documents dont la propriété « nom » a pour valeur « Juvénal ».

Soit un cluster ElasticSearch installé en local sur une machine.

Écrivez la requête HTTP qui crée un index appelé books.

curl -XPUT 'http://localhost:9200/books'

Écrivez la requête HTTP qui ajoute book1 dans le type computer de l’index books.

curl -XPUT 'http://localhost:9200/books/computer/book1' –d ‘

{

"book1":

{

...

}

}

Chapitre 10

Citez trois facteurs de limitation de Hadoop 1. Expliquez.

* Hadoop est une implémentation du MapReduce. Or, ce modèle de calcul est résolument batch, inadapté pour les calculs itératifs, interactifs et le streaming. Hadoop ne peut pas exécuter d’autres modèles que le MapReduce.
* Le HDFS est accessible de façon séquentielle, ce qui le rend inapproprié pour les problématiques nécessitant un haut niveau d’accès à la donnée. De plus, le nœud de référence sur lequel il s’appuie n’est pas tolérant aux pannes et constitue un point de défaillance unique de tout le cluster.
* Par défaut, Hadoop n’est pas sécurisé. En dehors de l’authentification Kerberos, la politique de sécurité de Hadoop devrait s’étendre et prendre en compte les niveaux de permissions et les autres aspects de la sécurité des données.

En quoi consiste l’authentification Kerberos ?

Kerberos est un protocole d’authentification réseau reposant sur un mécanisme de chiffrement symétrique et non sur les mots de passe en clair, qui risquent d’être interceptés frauduleusement.

Qu’est-ce que YARN et en quoi consiste-t-il ?

YARN (Yet Another Resource Negotiator) est un gestionnaire de ressources ou, plus formellement, une application générique de planification de tâches. Il permet de faire tourner au même moment plusieurs modèles de calcul dans le cluster.

Quelle est la différence entre YARN et le JobTracker ?

Le JobTracker s’occupe à la fois de planifier l’exécution des jobs MapReduce et de suivre la progression de l’exécution des tâches dans le cluster. YARN sépare ces deux rôles en deux processus indépendants :

* Un processus gestionnaire appelé Resource Manager gère l’utilisation des ressources du cluster.
* Un processus applicatif appelé Application Master gère le cycle de vie des jobs tournant sur le cluster.

Les ingénieurs de Yahoo ! ont découvert qu’autour de 4 000 nœuds, les performances du cluster Hadoop baissaient.

1. Expliquez brièvement la cause de cette baisse de performance.

Cette baisse des performances est due au cumul des fonctions (planification des jobs et suivi de la progression) par le JobTracker. Cela limite le nombre de jobs qu’il peut planifier et le nombre de nœuds qu’il peut suivre.

2. Expliquez comment YARN réussit à surmonter ce problème.

YARN sépare ces deux rôles en deux processus indépendants : le Resource Manager et l’Application Master (voir question précédente).

Dans la version 1 de Hadoop, l’unité de travail est le job MapReduce. Dans les versions 2 et ultérieures, comment s’appelle l’unité de travail ?

L’unité de travail est l’application YARN. Par exemple, un job MapReduce est une application « MapReduce-yarn », un job Spark est une application « Spark-yarn ».

Citez les deux composants d’une unité de travail en Hadoop 2.

* Les conteneurs
* L’application master

Quelle est la fonction du Node Manager dans YARN ?

Le Node Manager suit l’exécution des tâches de l’application YARN.

Qu’est-ce qu’un conteneur YARN ?

Ce sont des ressources qui ont été libérées par le Resource Manager pour l’exécution d’une tâche précise. C’est à l’intérieur des conteneurs que s’exécutent toutes les applications YARN.

Cochez les propositions vraies.

... Il y a autant d’Application Masters que d’applications YARN.

... Il y a autant de Node Managers que de nœuds dans le cluster.

... Sous validation de l’Application Master, les Node Managers créent les conteneurs.

... En cas d’erreur ou de panne du nœud, l’Application Master relance le conteneur.

... L’Application Master négocie les ressources avec le Resource Manager.

... YARN peut exécuter l’Iterative MapReduce.

Supposons que nous ayons exécuté une application *mapreduce-yarn* avec 10 tâches Map.

1. Combien faut-il générer de conteneurs YARN pour l’exécution de ces 10 tâches Map ? Justifiez votre réponse.

Dix conteneurs sont nécessaires, un pour chaque tâche.

2. Les 10 tâches vont-elles s’exécuter sur 10 nœuds ? Justifiez votre réponse.

Pas nécessairement. Tout dépend de la disponibilité des ressources sur les nœuds. Plusieurs conteneurs du même job peuvent être lancés sur le même nœud si celui-ci a suffisamment de ressources.

3. Combien d’Application Masters seront-ils nécessaires pour l’exécution de ces 10 tâches Map ? Justifiez votre réponse.

Un seul. Chaque application correspond à un seul Application Master.

Soit l’exécution d’un job MapReduce sur YARN. Sélectionnez le schéma qui correspond à cette exécution sur le cluster (en justifiant votre réponse).

Schéma 1

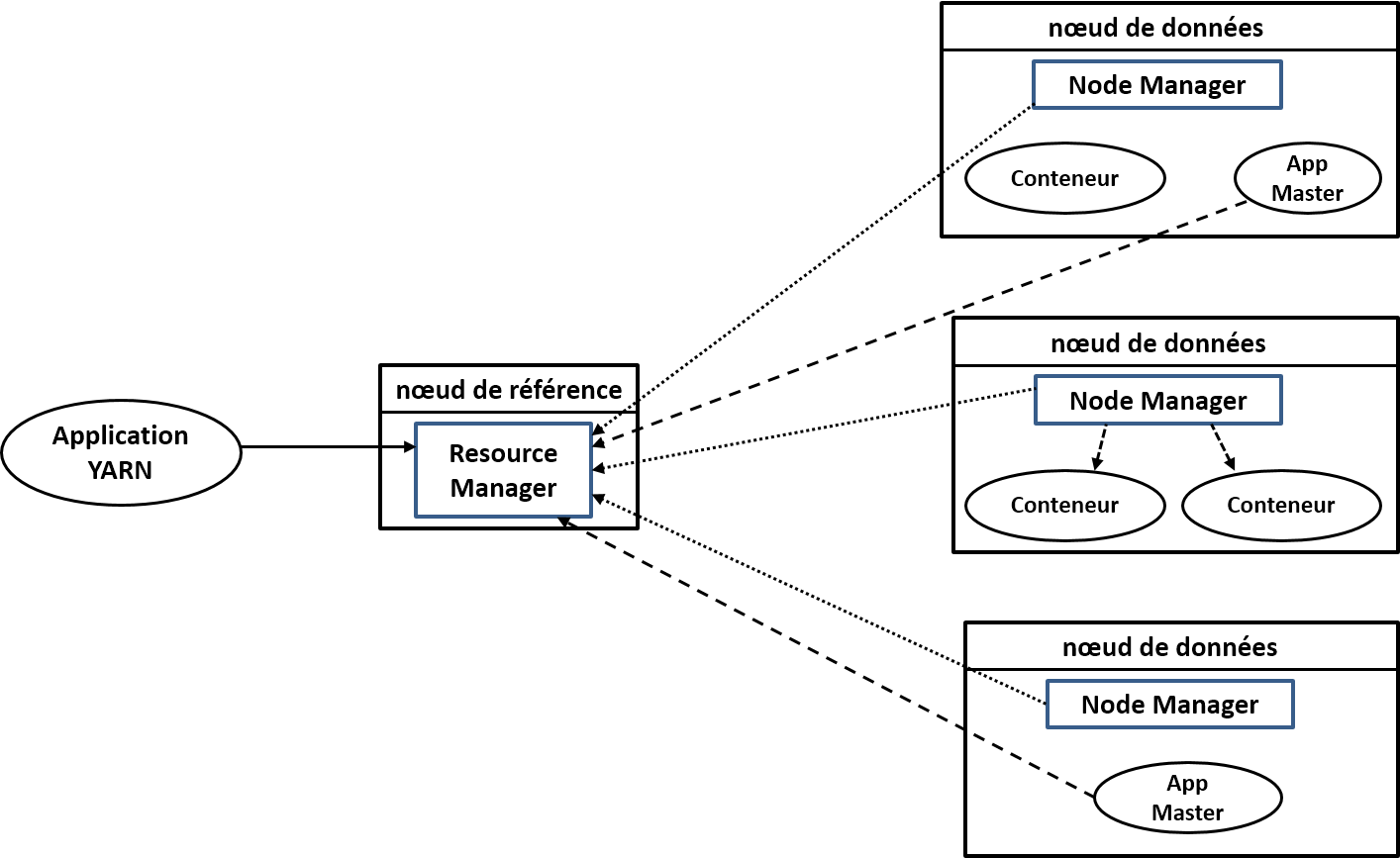


Schéma 2

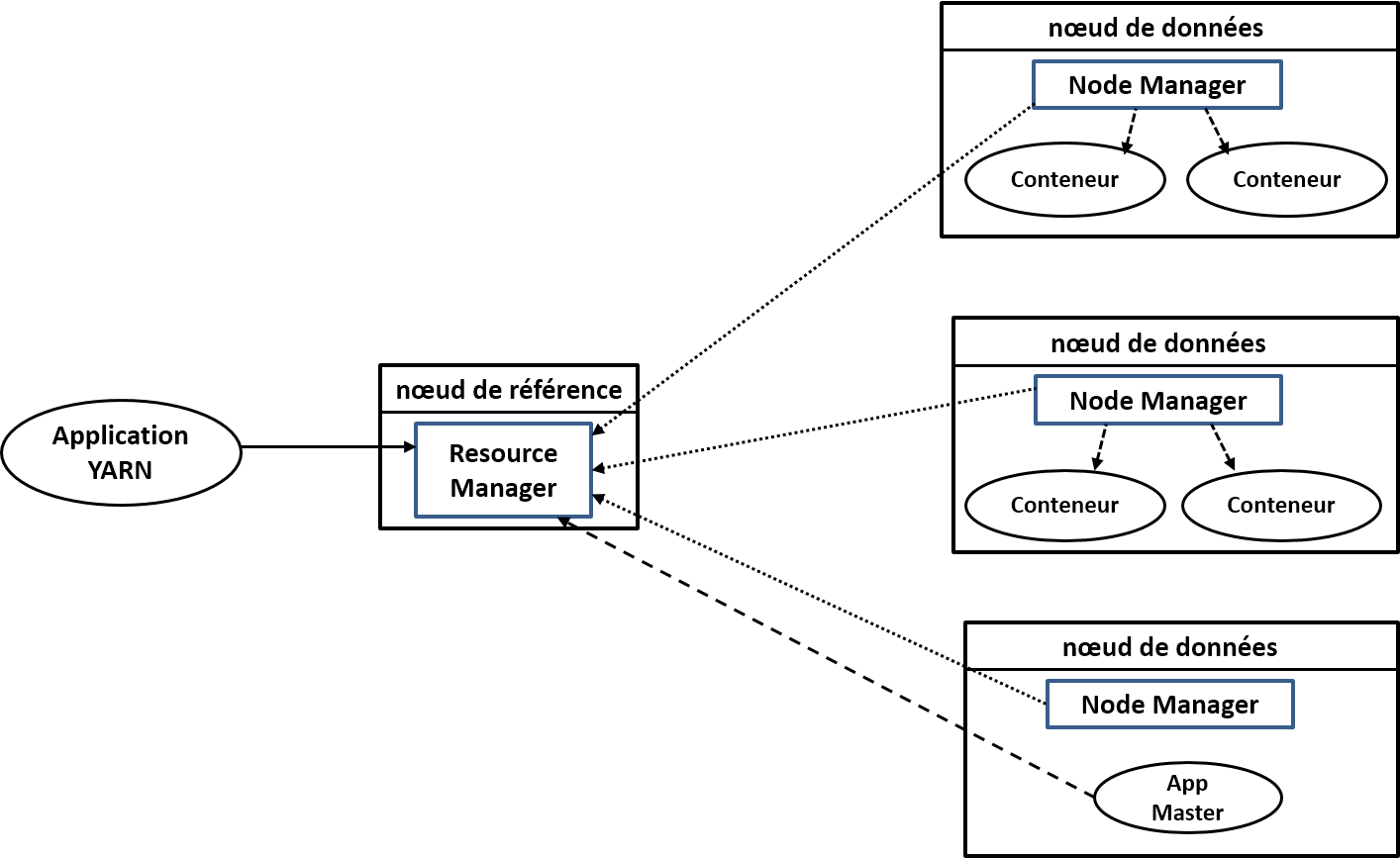
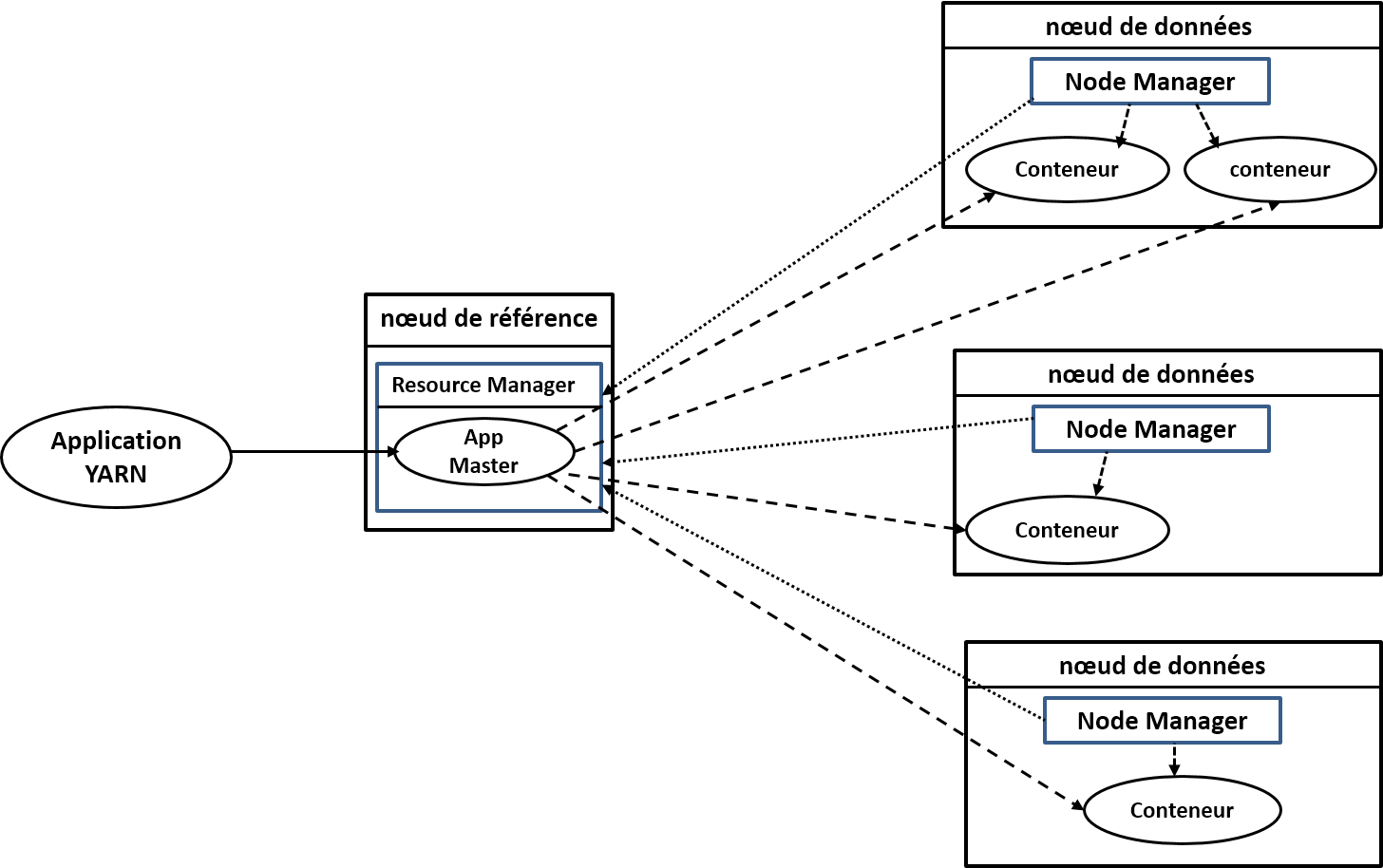
**

Schéma 3

**

Le schéma correct est le n° 2, car à chaque job correspond un seul Application Manager, qui s’exécute sur l’un des nœuds de données du cluster.

Par quel mécanisme Hadoop rend-il le nœud de référence tolérant aux pannes ?

Par la fédération HDFS.

Chapitre 11

Quelle est la différence entre un service applicatif et un service web ?

La différence entre les deux se situe au niveau du protocole de communication. Les services applicatifs utilisent des protocoles d’échange de données tels que l’ODBC. Les services web utilisent des protocoles web tels que le HTTP. De plus, là où un service applicatif utilise une API comme moyen de transfert de données, les services web utilisent plutôt un format de fichier, par exemple le JSON ou le XML.

ZooKeeper est-il un service applicatif ?

Oui.

Expliquez brièvement deux problèmes rencontrés en développement d’applications distribuées.

* Panne partielle. C’est une panne qui se produit lorsque, en cas d’interruption brutale du cluster, les nœuds qui s’échangeaient des données ne sont plus capables de savoir si les messages qu’ils ont envoyés ont été reçus.
* Coordination. Les nœuds d’un cluster ne sont pas « conscients » les uns des autres. Par conséquent, lors de l’ajout de nouveaux nœuds dans le cluster, il faut manuellement synchroniser le partage d’information entre les nœuds existants et ces nouveaux, ce qui est très complexe avec les pannes imprévisibles qui surviennent dans un cluster.

Citez et expliquez brièvement les deux phases de développement d’une application distribuée.

* Développement de l’application métier (logique métier que l’application vient résoudre).
* Développement du service de coordination et de communication des nœuds du cluster (comment les nœuds se synchronisent pour exécuter les tâches de l’application).

Expliquez le processus de communication des nœuds dans un cluster Hadoop.

Les nœuds de données envoient des informations vers le nœud de référence, et inversement. Cette communication est totalement transparente aux yeux de l’utilisateur et est gérée automatiquement par le cluster grâce aux informations de configuration (métadonnées) stockées dans le nœud de référence.

Que signifie l’expression *« les nœuds du cluster ne sont pas conscients des uns les autres »* ?

Chaque nœud se comporte comme s’il était seul dans le cluster. Il ignore que d’autres nœuds existent et ne sait pas reconnaître quand des nouveaux viennent d’être ajoutés au cluster.

En quoi la coordination des processus distribués consiste-t-elle ?

Elle consiste à synchroniser les nœuds lors de l’exécution des tâches d’une application distribuée.

Donnez la définition de ZooKeeper.

ZooKeeper est un service open source de coordination d’applications distribuées.

Quel est l’objectif de ZooKeeper ?

ZooKeeper masque la complexité du développement d’application dans un environnement distribué.

Quel est le rôle de ZooKeeper dans l’écosystème Hadoop ?

ZooKeeper prend en charge la complexité inhérente de la synchronisation de l’exécution des tâches distribuées dans le cluster et évite aux autres outils de l’écosystème de gérer ce problème eux-mêmes.

Complétez la phrase. ZooKeeper est un service de coordination...

1. Distribué.

Qu’est-ce qu’un ensemble ZooKeeper ?

C’est l’ensemble des nœuds sur lesquels est installé ZooKeeper.

Un ensemble ZooKeeper est-il un cluster au sens strict du terme ?

Non.

Quelle est la différence entre l’espace de noms de ZooKeeper et celui d’un système de fichiers ordinaire ?

À la différence de l’espace de noms d’un système de fichiers ordinaire, celui de ZooKeeper stocke également les données et les conserve en mémoire.

Quelle est l’importance de l’espace de noms ZooKeeper pour les processus distribués ?

L’espace de noms est l’élément clé de ZooKeeper, car c’est lui qui permet aux processus distribués de se coordonner entre eux en partageant les ressources (données) communes. Les processus distribués des applications des utilisateurs se donnent "rendez-vous" aux nœuds de l’espace de nom pour récupérer les données dont ils ont besoin pour la suite des opérations qu’ils doivent effectuer.

Pourquoi ZooKeeper est-il utilisé dans HBase ?

ZooKeeper indique au client HBase sur quels Region Servers se trouvent ses données.

Comment ZooKeeper gère-il les opérations d’écriture et de mise à jour dans l’espace de noms ?

Toutes les opérations d’écriture et de mise à jour sont transférées à une seule machine de l’ensemble appelée le *leader*. Les autres machines, les *followers*, reçoivent les propositions de message du leader et s’accordent sur la réponse. ZooKeeper utilise un protocole de messagerie atomique pour la communication entre le leader et les followers.

Quelles sont les trois phases de fonctionnement du service ZooKeeper ?

1. Élection de la machine leader

2. Diffusion du message

3. Stockage local et réplication de l’espace de noms

Chapitre 12

Comment définiriez-vous Storm ?

Storm est un environnement de développement et un moteur de déploiement de calculs distribués des données streaming sur un cluster.

Quelle est l’unité de travail en Storm ?

Les topologies.

Citez les deux composants de l’unité de travail de Storm et donnez leur fonction.

* Les spouts. Ce sont les sources streaming à partir desquelles Storm traite les données. Chaque source est exposée sur un spout différent.
* Les bolts. Ce sont les unités de calcul des traitements Storm.

Quel est l’élément clé de la scalabilité de Storm ?

Les bolts sont distribués sur les nœuds du cluster pour exécution parallèle. Storm peut augmenter la performance du cluster en accroissant le niveau de parallélisme des bolts.

Qu’est-ce qu’une topologie ?

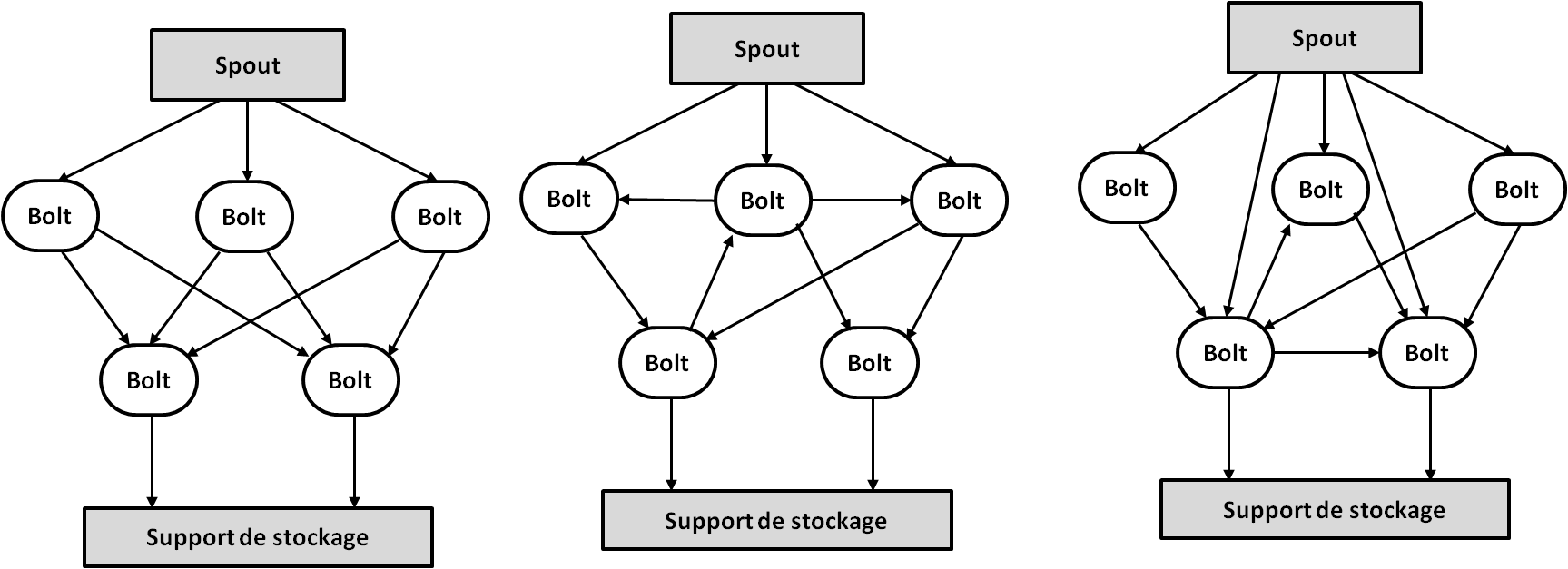
Les topologies sont pour Storm ce qu’est le MapReduce pour Hadoop. Une topologie est un graphe acyclique direct dans lequel les vertices ou nœuds sont soit des spouts soit des bolts.

Citez et expliquez deux différences entre une topologie et le MapReduce.

* Là où le graphe acyclique direct du MapReduce est limité à trois étapes, une topologie Storm peut en contenir davantage.
* Les topologies permettent des traitements à très faible latence, tandis que le MapReduce est un modèle de calcul batch inadapté pour le streaming.

Des trois topologies Storm suivantes, sélectionnez celle qui est correcte et justifiez votre réponse.

(a) (b) (c)



La topologie correcte est la (a), car il s’agit d’un graphe acyclique direct, donc non itératif.

Citez et donnez le rôle des trois composants d’un cluster Storm.

* Nimbus est le processus central d’un cluster Storm. Il tourne au niveau du nœud maître et c’est lui qui est responsable de la distribution des tâches (soit spouts, soit bolts) aux nœuds du cluster. Il est également responsable du rééquilibrage des charges en cas de défaillance dans le cluster et du suivi de l’exécution des topologies dans les nœuds.
* Les superviseurs sont des processus qui s’exécutent sur chaque nœud esclave et gèrent l’exécution locale des tâches. À chaque nœud esclave est affecté soit un spout, soit un bolt. Chaque superviseur peut gérer plusieurs tâches.
* ZooKeeper est un service distribué pour la coordination de la communication entre les nœuds dans le cluster. Il fournit un registre, utilisé par les nœuds du cluster Storm pour stocker et mettre à disposition les données concernant l’état des topologies en cours d’exécution et le statut de tous les superviseurs.

Quel est le mode de communication dans un cluster Storm ?

... Peer-to-Peer

Quelle est la différence entre Nimbus et le JobTracker ?

Nimbus n’est pas un point de défaillance unique comme le Jobtracker ; lorsqu’il tombe en panne, les topologies continuent de s’exécuter sans aucun problème.

Quelle est la différence entre les superviseurs et les TaskTrackers ?

Les superviseurs gèrent l’exécution d’au moins une tâche, tandis que les TaskTrackers correspondent à une seule tâche.

Quel est le rôle de ZooKeeper dans un cluster Storm ?

ZooKeeper fournit un registre, utilisé par les nœuds du cluster Storm pour stocker et mettre à disposition les données concernant l’état des topologies en cours d’exécution et le statut de tous les superviseurs.

Expliquez le mécanisme par lequel un cluster Storm obtient la haute disponibilité.

Toute communication entre Nimbus et les superviseurs transite par ZooKeeper et leur état est enregistré sur leur disque local ou dans le registre ZooKeeper.

Quelle est la différence entre un spout et un bolt ?

Le spout est une source de données, tandis que le bolt correspond à une unité de calcul.

Citez et expliquez trois techniques de partitionnement de tuples offertes par Storm.

* Regroupement par Shuffle (Shuffle grouping) : les tuples sont partitionnés de façon aléatoire.
* Regroupement par champ (Field grouping) : les tuples sont partitionnés selon une colonne précise.
* Regroupement total (All grouping) : la source de données streaming est entièrement répliquée sur tous les bolts.

Cochez la(les) proposition(s) correcte(s).

... Le bolt est distribué et traité en parallèle dans les nœuds du cluster.

*...* Les tâches du bolt sont distribuées dans le cluster et traitées en parallèle.

... Les tâches du bolt sont traitées sur un nœud différent du cluster.

Décrivez en cinq lignes maximum l’exécution d’un calcul stream en Storm.

L’exécution d’une topologie par Storm commence toujours par un spout. Les données streaming y sont transformées en tuples. Ces derniers sont partitionnés en lots et chaque lot est envoyé à un ou plusieurs bolts, lesquels sont répartis et exécutés sur le cluster.

Citez deux langages offerts par Storm pour le développement des topologies.

* Flux
* SQL

Quelle est la relation entre Trident et le SQL ?

Les requêtes SQL de l’utilisateur sont compilées en topologies Trident et ensuite exécutées en Storm. Ainsi, le SQL dans Storm est une abstraction de Trident.

Chapitre 13

Donnez la définition d’Oozie.

Oozie est un planificateur d’exécution des jobs, qui fonctionne comme un service sur un cluster Hadoop. Il est utilisé aussi bien pour Hadoop que pour les autres jobs qui peuvent s’exécuter sur un cluster.

Quelle est la différence entre Oozie et YARN ?

Oozie planifie le moment et l’ordre de soumission des jobs à YARN, qui s’occupe alors de leur exécution proprement dite sur le cluster.

Quelle est la différence entre Oozie et le JobTracker ?

Oozie planifie le moment et l’ordre de soumission des jobs au JobTracker, qui s’occupe alors de leur exécution proprement dite sur le cluster.

Cochez la bonne réponse. Oozie fonctionne comme...

... Un service qui exécute les jobs chaînés sur le cluster.

Oozie peut-il s’utiliser dans un cadre autre qu’Hadoop ? Justifiez votre réponse.

Non. Il tourne comme un service sur le cluster Hadoop.

Un workflow Oozie peut-il exécuter des programmes autres que ceux écrits avec les outils de l’écosystème Hadoop ?

Oui.

Citez les deux composants d’Oozie et donnez leur rôle.

* Le moteur de workflow gère et exécute les chaînes de jobs Hadoop.
* Le moteur de coordination des jobs planifie automatiquement l’exécution des chaînes de jobs.

Qu’est-ce qu’un workflow Oozie ?

Un workflow Oozie ou chaîne de jobs est un graphe acyclique direct de nœuds d’actions (qui déclenchent des actions particulières) et de nœuds de contrôle de flux (qui déterminent l’ordre d’exécution des actions).

Citez les deux composants d’un workflow Oozie et donnez leur rôle.

* Le fichier XML de configuration du moteur Oozie contient tous les paramètres relatifs à la planification et l’exécution des jobs du workflow.
* Le fichier XML de configuration du workflow des jobs contient tous les paramètres relatifs à la planification et l’exécution des actions du workflow.

En quoi le développement d’un workflow Oozie consiste-t-il ?

Il s’agit de développer les jobs à chaîner, puis paramétrer les fichiers XML de configuration du moteur Oozie et de configuration du workflow des jobs.

Supposez que vous voulez exécuter un programme MapReduce tous les jours à 12 h 00. Quelles sont les trois propriétés obligatoires que vous devez renseigner dans le fichier XML de configuration du moteur de coordination d’Oozie ? Quelles valeurs leur attribuerez-vous ?

Dans la balise <coordinator-app>, il faut modifier les valeurs des propriétés frequency, start et end :

<coordinator-app name="oozie\_exemple"

frequency="${coord:days(1)}"

start="2016-01-01T12:00Z" end="2020-12-04T12:00Z"

xmlns="uri:oozie:coordinator:0.1"

>

Quelle définition donneriez-vous de Sqoop ?

Sqoop (SQL-to-Hadoop) est un outil qui permet aux utilisateurs de transférer les données entre le HDFS de Hadoop et les bases de données relationnelles.

Quel est le rôle de Sqoop dans l’écosystème Hadoop ?

Son rôle est de planifier l’ingestion des données des SGBDR dans Hadoop et de Hadoop vers les SGBDR.

Quelle est la différence entre Sqoop et Pig ?

Pig est un environnement qui fournit un langage d’abstraction pour la rédaction des jobs MapReduce. Sqoop est un outil de transfert des données entre un SGBDR et Hadoop, qui utilise le SQL et le MapReduce.

L’import de données par Sqoop se fait-il uniquement d’un SGBDR vers le HDFS ?

Non.

Citez les cinq étapes du processus d’import de Sqoop.

1. Import par le client.

2. Récupération des métadonnées de la base relationnelle à importer.

3. Exécution du job MapReduce d’import.

4. Import des données de la base de données.

5. Écriture des données importées dans le cluster Hadoop.

Expliquez le procédé utilisé par Sqoop pour paralléliser l’import de données.

À l’aide d’une interface d’accès au SGBDR et des métadonnées de la base à importer, Sqoop accède à la table à importer et déclenche au niveau du cluster Hadoop un job MapReduce qui va paralléliser le processus (par défaut, en quatre threads, correspondant à quatre tâches Map). Lorsque le processus d’import commence, Sqoop cherche une colonne comme critère selon lequel il va diviser et répartir la charge de travail entre les tâches Map. Par défaut, c’est la clé primaire de la table qui est utilisée. Ensuite, il calcule les limites inférieure et supérieure de la colonne :

SELECT min(colonne\_critere), max(colonne\_critere) FROM table\_import

Par la suite, il divise les données de cet intervalle en quatre parties de taille égale, qu’il affecte aux quatre tâches Map.

Pouvez-vous utiliser Sqoop pour transférer les données d’une base de données distribuée comme HBase vers le HDFS ?

Non.

L’import est-il une opération de lecture de la base de données relationnelle vers le HDFS ?

Oui.

L’import est-il une opération d’écriture du HDFS vers une base de données relationnelle ?

Non.

Qu’est-ce qu’une transaction ?

Une transaction est une série d’opérations atomiques. La particularité de la transaction est qu’elle s’exécute selon le principe du tout ou rien, de sorte que l’échec d’une des opérations entraîne celui de toute la transaction.

Que représente une écriture en Sqoop ?

L’écriture correspond à l’exécution d’une instruction SQL INSERT ou UPDATE.

Une écriture représente-elle une transaction à part entière pour le SGBDR ?

Non.

Quel est l’impact des transactions dans l’export Sqoop ?

Grâce aux transactions, Sqoop peut paralléliser le processus d’export des données dans le SGBDR.

Expliquez de façon détaillée le processus d’export Sqoop.

La table cible doit déjà exister dans la base de données. Les fichiers à exporter sont lus et structurés en jeux de lignes selon les séparateurs de colonnes que vous fournissez. L’export les transforme en instructions SQL INSERT ou UPDATE qui injectent les lignes des fichiers dans la base de données. Le processus d’export est exécuté de façon parallèle par des tâches Map d’écriture. Chacune se connecte séparément à la base de données et insère les lignes par groupes de 100, que le SGBDR sauvegarde par un commit, entraînant ainsi la persistance de 10 000 lignes par transaction.

Pourquoi le processus d’export Sqoop n’est-il pas atomique ?

Les tâches d’écriture se connectent séparément à la base de données relationnelle.

Combien de transactions Sqoop déclenche-t-il lors de l’export ?

... 100 × (nombre de lignes du fichier/100) × nombre de tâches.

Citez trois raisons qui peuvent entraîner l’échec d’un export.

* Perte de la connexion du cluster Hadoop vers la base de données ;
* Tentative d’insertion de doublons ou de lignes non conformes aux contraintes d’intégrité référentielles ou sémantiques de la base de données ;
* Échec de structuration des fichiers du HDFS.

Supposez que vous souhaitez importer en batch une table MySQL table\_exo dans le HDFS. La colonne qui sert de critère de division pour les *mappers* est id\_exo. Fixez le nombre de tâches Map à 6 et importez la table dans le répertoire HDFS lab/import/data. Écrivez le script nécessaire pour réaliser cette opération.

$ sqoop import \

--connect jdbc:mysql://localhost/sqoop\_test \

--table table\_exo \

--split –by id\_exo \

--num-mappers 6 \

--target-dir lab/import/data

Supposez que vous souhaitez importer de façon incrémentale une table MySQL table\_exo dans le HDFS. L’incrément doit se faire à partir du 15/09/2010 et le *TimeStamp* de la table est date\_exo. La colonne qui sert de critère de division est id\_exo. Fixez le nombre de tâches Map à 6 et importez la table dans le répertoire HDFS lab/import/data. Écrivez le script nécessaire pour réaliser cette opération.

$ sqoop import \

--connect jdbc:mysql://localhost/sqoop\_test \

--table table\_exo \

--split –by id\_exo \

--check-column "date\_exo" \

--incremental "lastmodified" \

--last-value "2010-09-15" \

--num-mappers 6 \

--target-dir lab/import/data

Vous avez un répertoire de huit fichiers à exporter vers une table relationnelle. Le nombre de tâches Map d’écriture sera-t-il égal à 8 ?

Non.

À quoi l’argument direct sert-il dans l’export Sqoop ?

L’argument direct permet à Sqoop d’exporter 100 lignes en une instruction INSERT. Cela correspond à l’instruction BULK INSERT en SQL qui effectue les insertions par lots dans la base de données.

L’argument direct peut-il être utilisé lors des mises à jour dans l’export Sqoop ? Justifiez votre réponse.

Non. Les mises à jour sont des opérations unitaires.

Vous souhaitez exporter les fichiers du répertoire HDFS lab/import/data vers la table MySQL exo\_hadoop. Fixez le nombre de tâches Map à 4. Le séparateur de colonnes est la virgule et le séparateur de lignes est le retour à la ligne. Écrivez le script nécessaire pour réaliser cette opération.

$ sqoop export \

--connect jdbc:mysql://localhost/sqoop\_test \

--table exo\_hadoop \

--export-dir lab/import/data \

--direct \

--input-fields-terminated-by ‘\001’ \

--lines-terminated-by ‘\n’ \

--num-mappers 4

Chapitre 14

Quel est le but de HUE dans l’écosystème Hadoop ?

HUE (Hadoop User Experience) offre une interface utilisateur web qui simplifie le processus de création, de maintien et d’exécution de plusieurs types de jobs. En d’autres termes, c’est une interface unifiée de toutes les technologies de l’écosystème Hadoop.

Citez trois composants de HUE.

* FileBrowser pour naviguer dans le HDFS
* BeeWax pour l’écriture des scripts HiveQL
* Oozie App pour la création et la planification des workflows Oozie

Quel est le but d’Ambari dans l’écosystème Hadoop ?

Ambari a pour but d’aider les administrateurs à gérer les ressources d’un cluster Hadoop, à suivre sa performance et à gérer l’administration de sa sécurité.

Pourquoi est-il important de surveiller l’utilisation des ressources du cluster Hadoop ?

En surveillant les ressources du cluster, on s’assure que ce dernier n’est pas sous-utilisé ou sur-utilisé. S’il est sur-utilisé, il y a de fortes chances que les applications entrent en compétition les unes avec les autres pour l’obtention des ressources, baissant ainsi leur performance. Si le cluster est sous-utilisé, alors vous ne rentrez pas dans vos investissements.

Citez deux fonctions d’Ambari.

* Statistiques via Ambari Metric System pour suivre la performance du cluster.
* Surveillance de l’utilisation des ressources du cluster via Ambari Alert Framework.

Chapitre 15

Quels sont les freins à l’adoption de Hadoop en entreprise ?

Hadoop est open source, ce qui implique :

* une multiplicité de licences et de versions de logiciels difficile à suivre ;
* l’ouverture du code source, ce qui représente un risque de sécurité.

De plus, Hadoop a été conçu à la base pour les développeurs, pas les métiers.

Décrivez le modèle de développement communautaire.

En open source, les logiciels sont écrits par des communautés de développeurs physiquement distantes les unes des autres, mais fédérées par des organisations comme la fondation Apache, GitHub, Hacker News, Stack Overflow ou encore la Free Software Foundation. Lorsqu’une communauté finit le développement d’un logiciel, elle le distribue sous licence libre ou GNU GPL, qui autorise n’importe qui (développeur ou pas) à utiliser le logiciel, à en modifier le code source et à le redistribuer librement.

Quels sont les avantages du modèle de développement communautaire ?

Ce modèle favorise un haut niveau de réactivité lorsqu’il s’agit de corriger un bogue ou d’améliorer le logiciel. De plus, il met à disposition des logiciels de qualité supérieure à des coûts dérisoires.

Quelle est la différence entre une licence libre et une licence commerciale ?

La licence libre autorise n’importe qui (développeur ou pas) à utiliser le logiciel open source, à en modifier le code source et à le redistribuer librement. Dans la licence commerciale, l’éditeur détient la propriété exclusive du code du logiciel, qui est alors dit *propriétaire*.

Comment justifieriez-vous l’utilisation par une entreprise d’une distribution Hadoop plutôt que de sa version open source pure ?

* Gouvernance (maîtrise de l’évolution de Hadoop)
* Sécurité
* Facilité d’utilisation

Expliquez la stratégie de distribution.

Une entreprise (un éditeur de solutions logicielles) sélectionne une version open source de Hadoop, concentre ses efforts et développe des outils autour de cette version de sorte à l’améliorer, la commercialise et en offre le support.

Qu’est-ce qu’une distribution Hadoop ?

C’est une version commerciale de Hadoop.

Quel est l’avantage d’une distribution ?

L’avantage de la distribution est que l’évolution de Hadoop est plus maîtrisée, intentionnelle et suit une feuille de route développée par l’éditeur. De plus, ce dernier se porte garant de la sécurité de sa distribution et fournit le support nécessaire pour son exploitation.

Quel est l’inconvénient d’une distribution ?

La distribution n’offre pas toute la flexibilité offerte par l’open source.

Quel est le lien entre la fondation Apache et Hadoop ?

La fondation Apache est l’organisation open source qui fédère les communautés de développeurs qui travaillent sur Hadoop et son écosystème.

Lorsqu’un éditeur intègre Hadoop dans sa distribution, en obtient-il les droits de propriété intellectuelle ?

Non.

Citez les trois distributions majeures de Hadoop sur le marché.

* Cloudera CHD
* Hortonworks HDP
* MapR CDP

Quelle est la particularité de la distribution offerte par Hortonworks ?

Hortonworks est plus proche du monde open source de Hadoop que les autres éditeurs. Elle intègre dans HDP pratiquement tous les outils qui sont développés par les communautés.

Quelle est la particularité de la distribution Cloudera ?

Cloudera développe ses propres outils et met beaucoup l’accent sur l’innovation technologique et la formation.

La distribution Cloudera CDH est-elle la seule à proposer Impala ?

Non.

Quelle est la caractéristique principale de la distribution MapR ?

Elle combine l’innovation technologique au travail de la communauté Apache Hadoop.

Citez et précisez les quatre composants de la distribution MapR.

* MapR-FS (MapR File System) est le système de fichiers distribué de la distribution. Techniquement, c’est une implémentation de l’API HDFS combinée aux fonctionnalités d’un système de fichiers distribué en lecture et écriture.
* MapR Stream est une implémentation de l’API Kafka, optimisée pour gérer les streams de données sur le MapR-FS.
* MapR DB (MapR Data Base) est une implémentation de l’API HBase, optimisée pour profiter des avantages du MapR-FS.
* Apache Hadoop. La distribution de MapR inclut également certains projets du monde open source.

Quel est le but d’un benchmark de solutions d’éditeurs ?

Un benchmark a pour but de comparer comment les solutions d’éditeurs répondent au besoin métier.

Sur quelle base comparerez-vous les solutions d’éditeurs ?

Vous les comparerez sur la base de votre besoin ou des besoins du client.

Citez et expliquez quatre critères sur lesquels vous vous appuierez pour choisir une distribution Hadoop.

* Disponibilité des briques logicielles. Vous devez vous assurer que l’éditeur intègre dans sa distribution les briques logicielles open source dont vous avez besoin ou qu’il en propose des substituts.
* TCO (Total Cost of Ownership). Il est important de savoir combien vous coûte votre cluster Hadoop. Le TCO intègre les coûts de constitution du cluster, de sa maintenance et des licences de distribution.
* Convivialité. Cela décrit le niveau d’aisance avec lequel les utilisateurs peuvent travailler avec une distribution.
* Performance. Au départ, la performance d’un cluster Hadoop se mesurait par le débit des données qu’il était capable de traiter par batch et par son niveau de scalabilité. Avec l’introduction des nouveaux cas d’usage, et spécialement des problématiques temps réel, le concept de performance a évolué pour tenir compte du niveau de latence des traitements. Aujourd’hui, évaluer la performance d’un cluster Hadoop signifie évaluer sa vitesse d’ingestion des données, la vitesse de traitement de ses modèles de calcul et sa capacité à s’étendre techniquement et géographiquement.

Citez trois modes de sécurisation d’un cluster Hadoop.

* Authentification Kerberos
* Autorisations de niveau de service
* Confidentialité des données

En quoi consiste l’authentification Kerberos ?

Il s’agit d’un protocole d’authentification réseau, qui repose sur un mécanisme de chiffrement symétrique et non sur les mots de passe en clair, évitant ainsi le risque d’interception frauduleuse de ces derniers.

En général, comment les distributions Hadoop sont-elles facturées ?

Les éditeurs facturent la location de leur distribution sous forme de licence souscrite annuellement. Son prix varie en fonction du nombre de nœuds du cluster et du niveau de support offert. D’autres facteurs peuvent également être pris en compte.

Donnez la définition et le rôle des protocoles SSL et TLS.

TLS (Transport Layer Security) est le prédécesseur de SSL (Secure Sockets Layer). Ce sont tous deux des protocoles de sécurisation des échanges de données sur un réseau. Ils fonctionnent selon un mode client-serveur et satisfont les objectifs de sécurité suivants : authentification du serveur, cryptage des données échangées et authentification du client.

Des distributions Hortonworks, Cloudera et MapR, laquelle vous semble la meilleure ? Justifiez votre réponse.

Il n’y a pas de meilleure distribution. Tout dépend de votre besoin.

Chapitre 16

Décrivez en quoi consiste l’interfaçage logiciel et quel est son but.

On parle d’interfaçage lorsque des logiciels s’échangent des données via une passerelle, qui peut prendre la forme d’un format de fichier standardisé ou de l’API d’un service.

Décrivez en quoi consiste l’intégration logicielle et quel est son but.

On parle d’intégration lorsque les logiciels utilisent les mêmes données. Cela consiste à conformer les applications au standard ou aux conditions imposées par l’application qui centralise les données qu’elles utilisent.

Quelle différence faites-vous entre l’intégration et l’interfaçage de systèmes ?

Dans l’intégration, toutes les applications dépendent de l’application qui possède les données utilisées et, donc, doivent se conformer à l’interface qu’elle offre. Dans l’interfaçage, les applications sont indépendantes les unes des autres, utilisent un standard (API ou format de fichier) pour échanger les données. L’API utilisée dans l’intégration logicielle n’est pas nécessairement un standard.

Expliquez en quoi consiste l’intégration horizontale.

L’intégration horizontale consiste à interfacer son application avec un grand nombre d’autres logiciels afin qu’elle soit compatible avec eux.

Selon vous, quel est le coût à payer pour un éditeur lorsqu’il intègre ses solutions horizontalement ?

L’éditeur doit rendre son application compatible avec plusieurs autres logiciels, ce qui n’est pas toujours évident compte tenu des évolutions de ces logiciels.

Selon vous, quel est l’avantage pour un éditeur à intégrer ses solutions horizontalement ?

L’entreprise concentre ses efforts de développement sur les fonctionnalités de son application, sans avoir à développer elle-même celles qui ne sont pas en lien direct avec ses processus métiers.

Expliquez en quoi consiste l’intégration verticale.

L’intégration verticale consiste à développer une solution qui incorpore les fonctionnalités d’un autre logiciel. Par exemple, si l’éditeur propose une application métier, il peut lui incorporer un SGBD. Cette stratégie est très courante dans la commercialisation des ERP où les applications sont développées pour fonctionner avec un SGBDR spécifique qui est compris dans l’offre.

Selon vous, quel est le coût à payer pour un éditeur lorsqu’il intègre ses solutions verticalement ?

Il faut développer des fonctionnalités qui ne sont pas le cœur de son application. Cela peut finalement rendre le logiciel très complexe, difficile à maintenir et plus coûteux.

Selon vous, quel est l’avantage pour un éditeur à intégrer ses solutions verticalement ?

Son offre est plus complète.

Quand dit-on qu’une solution est intégrée verticalement à Hadoop ?

Hadoop est nativement intégré ou embarqué dans cette offre.

Quand dit-on qu’une solution est intégrée horizontalement à Hadoop ?

Elle s’interface avec Hadoop, qui devient juste une source de données comme une autre.

À votre avis, quelle est la meilleure stratégie entre l’intégration horizontale et l’intégration verticale à Hadoop ? Justifiez votre réponse.

Il n’y a pas de meilleure stratégie. Tout dépend de la vision stratégique de l’éditeur et du besoin auquel il souhaite répondre.

Citez et donnez la fonction des trois composants de la solution proposée par SAS intégrée verticalement à Hadoop.

* SAS Grid Computing permet à d’autres solutions SAS de répartir les charges de calcul dans un pool de ressources partagées, comme un cluster, ou un groupe de machines connectées à un même réseau informatique.
* SAS In-Database permet d’intégrer des applications SAS dans les bases de données. Elle intègre l’architecture de calcul massivement parallèle directement sur les moteurs de bases de données ou sur Hadoop, pour accélérer les traitements et améliorer la performance de calcul.
* SAS In-Memory Analytics permet de profiter de la disponibilité de la mémoire et de la puissance de calcul d’un cluster Hadoop.

Citez et donnez la fonction des trois composants de la solution proposée par Pivotal intégrée verticalement à Hadoop.

* Pivotal GemFire est une version commerciale d’*Apache Geode*, un SGBD *in-memory* open source capable de fournir des accès cohérents à des applications de traitement de données à fortes charges de calcul, sur des architectures Cloud distribuées.
* Pivotal GreenPlum est une version commerciale de GreenPlum, un moteur de bases de données open source conçu spécialement pour les Data Warehouses.
* Pivotal HBD est une version commerciale d’Apache HAWQ.

Qu’est-ce qu’un projet Apache en cours d’incubation ?

C’est un statut que reçoivent tous les nouveaux projets soumis à Apache, indépendamment de leur niveau de maturité ou de leur performance.

Pourquoi est-il nécessaire qu’un projet soumis à Apache passe par le statut d’incubation ?

Apache s’assure par là que le projet est suffisamment stable et s’intègre de façon cohérente aux autres projets de la fondation.

Chapitre 17

Quelle différence y a-t-il entre la scalabilité au sens économique du terme et la scalabilité au sens informatique du terme ?

La scalabilité informatique fait référence à la capacité d’ajouter de nouvelles ressources à un système informatique pour augmenter sa performance de calcul. La scalabilité au sens économique du terme fait référence à la capacité de fabriquer un produit supplémentaire sans entraîner des coûts supplémentaires significatifs.

Quel est le facteur principal de l’émergence du Cloud ?

Il s’agit de la baisse des coûts des infrastructures informatiques.

Qu’est-ce que le Cloud computing, économiquement parlant ?

C’est un modèle économique dans lequel une entreprise (fournisseur Cloud) offre via Internet l’excédent de ses ressources informatiques à d’autres entreprises (clients), qui les consomment sur la base d’un paiement à l’usage.

Qu’est-ce qu’un service Cloud ?

C’est toute ressource informatique qui peut être partagée via Internet : RAM, micro-processeur, bande passante, logiciel, interface applicative, environnement de développement, clusters entiers d’ordinateurs...

Quelle est la relation entre un service web et un service Cloud ?

Les services Cloud sont offerts sous forme de services web. Leur accès est contrôlé par une API et leur exploitation fait l’objet d’une facturation fonction du volume de ressources utilisées.

Qu’est-ce qu’une offre Cloud ?

Ce sont des tarifications d’accès à des quantités précises de ressources durant un certain intervalle de temps ; par exemple, accès à 40 Go de disque, micro-processeur de 1 GHz 4 Go de RAM à 5 euros/heure.

Expliquez en quoi consiste la tarification Pay-as-you-go.

C’est une tarification à l’usage. Le client s’abonne à un produit sur une base temporelle (mois, jour, heure) et paye uniquement pour ce qu’il a consommé dans cette période. Il peut à tout moment se désabonner de l’utilisation du service.

Expliquez en quoi consiste la virtualisation.

La virtualisation fait référence à l’abstraction physique des ressources d’un système informatique. Elle a permis d’optimiser l’usage des ressources matérielles en les partageant simultanément entre plusieurs utilisateurs.

Qu’est-ce qu’une machine virtuelle ?

C’est une application logicielle qui est l’abstraction des ressources de la machine hôte (ou machine sur laquelle est démarrée cette machine virtuelle). La machine hôte voit cette machine virtuelle comme une application à laquelle elle distribue ses ressources.

Qu’est-ce qu’un hyperviseur ?

L’hyperviseur est une couche logicielle qui gère les ordinateurs et les serveurs d’un système informatique comme un ensemble machines auxquelles il affecte dynamiquement les ressources en fonction des besoins des utilisateurs.

Quel est le rôle de l’hyperviseur dans le processus de virtualisation ?

Il gère les ordinateurs et les serveurs d’un système informatique comme un ensemble de machines auxquelles il affecte dynamiquement les ressources en fonction des besoins des utilisateurs.

Citez et expliquez les trois niveaux de service Cloud.

* L’IaaS (Infrastructure-as-a-Service) permet à des entreprises de louer des centres de données, serveurs, réseaux, bref de s’équiper d’un système informatique complet sans s’inquiéter de créer et de maintenir la même infrastructure en interne.
* Le PaaS (Platform-as-a-Service) est un niveau de service du Cloud dans lequel le fournisseur héberge et fournit un environnement de développement intégré (EDI), qu’un développeur peut utiliser pour créer et développer des logiciels, éliminant de ce fait la nécessité pour l’entreprise d’acheter constamment des EDI.
* Le SaaS (Software-as-a-Service) est un niveau de service du Cloud dans lequel le fournisseur héberge et met à disposition un logiciel ou une application, sans que l’entreprise cliente n’ait à se soucier de l’installation et de la maintenance de cette application en interne.

À quel(s) niveau(x) de service Cloud l’utilisation de Hadoop dans le Cloud correspond-elle ?

... L’IaaS

... Le SaaS

À quel niveau de service Cloud l’utilisation des ressources d’un cluster de machines en ligne correspond-elle ?

... L’IaaS

Quelles sont les deux options d’utilisation de Hadoop dans le Cloud ?

* L’IaaS préconfiguré
* Hadoop as a Service

Expliquez en quoi consiste l’IaaS préconfiguré.

Le fournisseur Cloud propose des instances préconfigurées de machines virtuelles avec une image de distribution Hadoop pré-packagée à déployer sur ces instances. Chaque instance correspond à une configuration précise de machine virtuelle.

Expliquez en quoi consiste Hadoop-as-a-Service.

Hadoop est offert comme un service web.

Quelles sont les trois différences majeures entre Hadoop-as-a-Service et l’IaaS préconfiguré ?

* Dans l’option IaaS préconfiguré, seules les instances de machines virtuelles sont facturées. Dans l’option Hadoop-as-a-Service, ce sont des instances de service Hadoop qui sont facturées.
* En Hadoop-as-a-Service, le cluster peut s’approvisionner automatiquement en ressources si la charge de travail vient à dépasser la capacité de calcul du cluster, ce qui n’est pas le cas dans l’IaaS préconfiguré, où il faudrait arrêter le cluster, acheter des instances de plus grande taille, redimensionner manuellement le cluster et le relancer.
* Hadoop-as-a-Service est recommandé lorsqu’on veut travailler avec un composant spécifique de l’écosystème Hadoop, tandis que l’IaaS préconfiguré est recommandé pour les tests des versions bêta des composants Hadoop.

Qu’est-ce qu’un *Object store* ?

C’est l’offre Cloud de stockage de données associée à une offre Cloud Hadoop. C’est en quelque sorte un disque dur en Cloud.

Expliquez de façon globale le processus de fonctionnement de Hadoop dans le Cloud.

Lorsque vous exécutez un traitement, les données sont chargées de l’environnement local du client vers un *Object store*, puis les machines virtuelles des nœuds de données sont instanciées et provisionnées conformément à la configuration à laquelle vous avez souscrit, les traitements sont réalisés par ces machines qui lisent les données depuis l’*Object store*. À la fin du traitement, si vous avez souscrit à un service Cloud de stockage, le résultat est renvoyé dans cet *Object store* et les machines sont supprimées ; sinon, les résultats et les données sont supprimés à la déconnexion du service.

Quel est le type de l’offre Cloud Azure HDInsights de Microsoft ?

... Hadoop-as-a-Service

Quel est le type de l’offre Cloud EMR d’Amazon ?

... Hadoop-as-a-Service

Sur quelle distribution de Hadoop l’offre Amazon EMR s’appuie-t-elle ?

... MapR

Expliquez de façon détaillée le mode de facturation d’Amazon EMR.

Le modèle économique d’Amazon EMR consiste à facturer l’utilisation de chaque instance du cluster à l’heure. Trois éléments principaux composent cette tarification : le type d’instance, l’édition de distribution MapR sélectionnée et la zone géographique.

Hadoop Cloud Amazon = instances Amazon EMR + instances Amazon EC2 + abonnement stockage Amazon S3

Sur quelle distribution de Hadoop l’offre Azure HDInsight s’appuie-t-elle ?

... Hortonworks

Expliquez de façon détaillée le mode de facturation d’Azure HDInsight.

Microsoft Azure HDInsight est une offre qui est facturée sur les composants Hadoop utilisés, les types de machines virtuelles qui forment le cluster et l’éventuel compte de stockage.

Hadoop Cloud Microsoft = instances Azure + instances HDInsight + abonnement stockage Azure

Des trois modes d’adoption de Hadoop (distribution, embarqué, dans le Cloud), lequel recommanderiez-vous à vos clients ? Justifiez votre réponse.

(Réponse laissée à la discrétion du lecteur)

Des trois modes d’adoption de Hadoop (distribution, embarqué, dans le Cloud), quel est celui qui permet de diffuser Hadoop plus rapidement dans les entreprises ? Justifiez votre réponse.

(Réponse laissée à la discrétion du lecteur)

Que peut faire une entreprise pour s’assurer de la viabilité à long terme d’un fournisseur Cloud ?

L’entreprise peut faire une étude *due diligence* du fournisseur Cloud, incluant au minimum l’analyse de sa stabilité financière.

Que signifie *« être responsable des données »* ?

Dans le cadre du Cloud, cela fait référence à la personne qui est pénalement ou juridiquement responsable en cas de piratage ou de fuite de données hébergées dans le Cloud.

Quels sont les facteurs qui influencent la responsabilité pénale d’un fournisseur Cloud vis-à-vis des données de ses clients ?

Deux facteurs majeurs influencent la responsabilité pénale d’un fournisseur Cloud vis-à-vis des données de ses clients :

* la localisation géographique du fournisseur ou la juridiction de l’État dans lequel il a sa raison sociale ;
* la localisation géographique des données hébergées dans le Cloud.

La juridiction du pays d’origine du fournisseur peut-elle violer le critère de propriété des données ?

Oui.

Quelle est la portée géographique du *Patriot Act* ?

Le *Patriot Act* autorise l’État américain, en cas de nécessité, à fouiller parmi les données hébergées dans le Cloud des entreprises américaines (Amazon et Microsoft notamment), indépendamment de la localisation de leurs *data centers* dans le monde.

De manière générale, les fournisseurs Cloud sécurisent-ils mieux les données que les entreprises ? Justifiez votre réponse.

Oui. La sécurité est un processus support pour une entreprise classique, alors qu’elle est un processus métier clé chez les fournisseurs Cloud. Ceux-ci sont conscients des enjeux, savent que leur réputation dépend de leur capacité à garantir la protection des données de leurs clients. Par conséquent, ils investissent massivement dans le développement des systèmes hautement sécurisés.

Expliquez en quoi consiste le cryptage des données et quel est son apport dans la sécurité des données en Cloud.

Le cryptage assure que, même si le pirate arrive à franchir les barrières de sécurité du système du fournisseur et à voler les données, elles seront inutilisables s’il n’arrive pas à les décrypter.

Qu’est-ce que le SLA dans le Cloud et quel est son but ?

Le SLA (Service Level Agreement) est un document légal qui spécifie l’ensemble des fonctionnalités attendues du service Cloud, aussi bien de la part du client que de celle du fournisseur. Il sert à protéger le fournisseur Cloud et le client en cas de conflit sur la gestion des niveaux de services.

Chapitre 18

Que vous inspire le terme *Big Data* ?

D’un point de vue technique, le Big Data peut se voir comme les défis et opportunités qui sont associés à l’explosion des données.

Donnez la définition universellement acceptée du Big Data.

*« Le Big Data est un fort volume de données très variées et produites très rapidement, qui exigent des techniques innovantes et rentables de traitement d’information pour une meilleure prise de décision »* (Gartner, Doug Laney).

Qu’est-ce que le *time to market* ?

C’est le temps nécessaire à une entreprise pour répondre aux opportunités ou aux menaces qui surviennent sur son marché.

Quel est le rapport entre le *time to market* et le Big Data ?

La réactivité implique la capacité à exploiter rapidement les volumes de données importants, porteurs des changements en cours sur le marché.

Quels sont les trois aspects mis en avant dans la définition du Big Data fournie par le Gartner ?

* Données : volume, variété, vitesse
* Moyens techniques et technologiques pour l’exploitation de ces données
* Opportunités que cachent les données.

Que pouvez-vous dire de l’expression *« faire du Big Data »* ?

Cette expression n’a pas de sens. On ne *fait* pas un phénomène, on y répond, en s’en protégeant ou en y bénéficiant.

Le volume de données est-il un élément suffisant pour caractériser le Big Data ? Justifiez votre réponse.

Non. Le marché sait déjà comment traiter les gros volumes de données. Ce qu’il ne sait pas bien faire, c’est profiter des opportunités qu’ils recèlent.

Citez trois facteurs qui ont favorisé l’émergence du Big Data.

* Évolution culturelle des *Digital natives*
* Open source
* Baisse des coûts de l’infrastructure informatique

Quel est le rapport entre le Big Data et l’économie numérique ?

Le Big Data est l’aspect quantitatif de l’économie numérique.

Les *Digital natives* sont les individus nés dans les années 1980. Quel est leur rapport avec le Big Data ?

Les années 1980 correspondent au début de la révolution informatique entraînée par les PC et Internet, qui a profondément modifié les besoins de consommation des *Digital natives*. Répondre à ces besoins demande d’innover et l’innovation passe forcément par l’exploitation efficace de la donnée.

Qu’est-ce qu’un *business model* ou modèle économique ?

C’est la façon dont une entreprise combine un ensemble de facteurs de production pour produire un bien ou service qui lui est profitable.

Qu’est-ce qu’une start-up ?

Une start-up n’est pas une entreprise qui vient d’être créée, mais une entreprise, de grande ou petite taille, qui est en réinvention continue de son modèle d’affaires.

En quoi les start-up se démarquent-elles des entreprises traditionnelles ?

Les start-up sont en réinvention constante, alors que les entreprises traditionnelles fonctionnent avec un modèle d’affaires fixe.

Quel est le rapport entre le Big Data et les start-up ?

Les start-up ont démontré qu’il est possible d’exploiter l’explosion des données qui caractérise le Big Data pour créer des produits et services innovants, basés exclusivement sur la donnée. Elles ont également démontré qu’il est possible de profiter des données pour créer des modèles économiques innovants et rentables, à structure de coûts flexible, basés exclusivement sur l’exploitation du Big Data. Les start-up, appuyées sur des technologies comme l’écosystème Hadoop, le Cloud, la virtualisation, l’open source et la disponibilité de la donnée, réussissent à aligner leur modèle d’affaires avec les évolutions culturelles de la génération Y.

Quel est le rapport entre l’*Internet des objets* et Hadoop ?

L’Internet des objets consiste à prendre par Internet le contrôle des objets de la vie courante (téléviseur, véhicule, maison). Pour ce faire, des capteurs sont intégrés dans ces objets et génèrent un volume énorme de données, qui est ensuite exploité. Les objets connectés sont une source de données en *streaming*. Les outils d’ingestion et de traitement Streaming d’Hadoop seront donc utilisés pour traiter toutes ces données.

En quoi le Cloud computing et la virtualisation ont-ils favorisé l’émergence du Big Data ?

Grâce au modèle de paiement Pay-as-you-go du Cloud, associé à la baisse des coûts d’infrastructures informatiques et à l’utilisation de la virtualisation, la constitution des clusters est devenue accessible au grand public, ce qui favorise l’expérimentation et l’exploitation à grande échelle des données.

Quel est l’impact de l’open source dans l’économie numérique ?

En partageant leur code et en autorisant sa réutilisation et son amélioration par d’autres, les développeurs mettent à disposition de tout le marché des ressources logicielles peu coûteuses et quasiment supérieures aux solutions propriétaires. Avec l’open source, l’infrastructure de gestion des données peut être formée à partir de la combinaison de plusieurs outils et applications ; le développement logiciel sort de l’approche monolithique pour intégrer une approche modulaire. Cela rend l’exploitation du logiciel plus simple et favorise une innovation plus rapide.

Selon la Commission générale de terminologie et de néologie, le Big Data, ou mégadonnées, fait référence aux « données, structurées ou non, dont le très grand volume requiert des outils d’analyse adaptés ». Qu’en pensez-vous ?

Définir le Big Data sur le seul aspect volume des données est sans intérêt et beaucoup trop réducteur.