***Correction Examen BDD 21/12/2018***

Q1 :

MongoDB utilise le protocole de validation à 2 phases : vrai

(Le protocole de validation à 2 phases(2 PC): Phase de préparation (phase de vote) -> préparer les requêtes & Phase de validation (phase de décision)-> COMMIT Si c'est bon SINON ROLLBACK)

Q2 :

Le mode de fragmentation (sharding) de MongoDB est caractérisé comme Horizontal

Le sharding est une méthode pour distribuer les données sur plusieurs instances de BDD. Pour cela on crée un cluster composé de plusieurs machines suer laquelle les données vont être répartis.

Mise à l'échelle horizontale fonctionne avec l'ajout de serveurs supplémentaires et répartit la charge sur plusieurs serveurs

Mise à l'échelle verticale fonctionne avec l'amélioration des performances d'un serveur unique en ajoutant des processeurs plus puissants, en mettant à niveau la RAM ou en ajoutant plus d'espace disque au système.

Q3 :

Avec MongoDB, on utilise le paramètre -configsvr:

* Lors de la spécification du répertoire d'un serveur mongo
* lors de la mise en place d'un cluster fragmenté avec réplication
* lors de la mise en place d'un replica set

Q4 :

On utilise la commande rs.initiate() : Aucune de ces réponses n’est correcte

Re.initiates() pour initialiser un replica set.

Q5 :

L'exécutable mongos gère :

* Le routage des requêtes
* La fusion des résultats intermédiaires d'une requête

Q6 :

A quoi correspond la notion de chunk lors de la fragmentation dans mongoDB ?

Une collection (chunks) contient les données, découpées en morceaux de 255Ko (configurable mais pas plus de 16Mo).

Q7 :

Le système de gestion de base de données MongoDB ne supporte que des indexations sur le

champ \_id : faux

-id c’est par défaut, on peut créer et supporter un indexe sur n’importe quelle variable.

Q8 :

A quoi correspond (fournir le nom et sa signification) le symbole : C’est le symbole de la semi-jointure, on fait une jointure conservant uniquement les attributs s’une table.

Q9 :

Indiquer 3 pannes que l’on retrouve dans SGBD Distribué et pas un SGBD centralisé :

* La perte d’un message
* La panne d’une liaison de communication
* La panne d’un site
* La panne de routage
* Le partitionnement du réseau
* Une panne dans un site d’un SGBDD ou une rupture de ligne de communication isolant un ou quelques sites n’immobilise pas l’ensemble du système

Q10 :

Dans le protocole de validation à 2 phases (2PC), à quoi correspond le problème du blocage :

Le plus grand inconvénient du protocole de validation en deux phases est qu'il s'agit d'un protocole bloquant. Si le coordinateur échoue définitivement, certains participants ne résoudront jamais leurs transactions : après qu'un participant a envoyé un message d'accord au coordinateur, il se bloquera jusqu'à ce qu'un commit ou un rollback soit reçu.

Q11 :

Dans une base de données distribuée, la notion de transparence concerne :

* Le partitionnement
* La localisation de données sur le cluster
* La réplication

Q12 :

Dans l'optimisation de requêtes d'un SGBD distribué, citer 2 contraintes à prendre en compte que l'on ne rencontre pas dans un système centralisé :

* Choix de la meilleure localisation de fragment
* Cout de transfert des données
* Exploitation du paraplasme
* Transfert entre site

Q13 :

Le protocole de validation à 2 phases (2PC) est indispensable dans le cas d'une transaction local d'un SGBD distribué : VRAI

Q14 :

A quoi correspond un mode semi-synchrone de réplication :

Avec une réplication asynchrone des données comités sur le disque du serveur primaire peuvent être perdues en cas de panne.

Q15 :

Les opérateurs disponibles dans la phase 1 de 2PC sont :

Abort T – ready T – no T

Q16 :

Fournir 3 approches différentes pour gérer la réplication dans un SGBD distribué

* Plusieurs leaders
* Sans leaders
* Unique leader

Q17 :

L'architecture de calcul distribuée prédominante du cloud computing est : Shared-nothing

Q18 :

Fournir au moins 2 différences entre un SGBD parallèle et un SGBD distribué :

SGBD parallèle :

* Les nœuds sont physiquement proches
* Nœuds connectés par LAN avec haut débit
* Coût de communication est considéré comme faible

SGBD distribué

* Nœuds peuvent être éloignés les uns des autres
* Nœuds connectés par un réseau public
* Coût de communication n’est pas négligeable et les problèmes associés sont importants

Q19 :

Le fait de découper une requête en plusieurs morceaux et de les exécuter en parallèle sur plusieurs serveurs correspond au parallèlisme : intra-query

Q20 :

Le fait d'exécuter en parallèle la même requête sur plusieurs serveurs correspond au parallèlisme : inter-query

Q21 :

La commande ANALYSE lit l'ensemble des tuples de la table analysée pour construire ses statistiques : Vrai

Analyzer – collecter les statistiques d’une base de données.

Q22 :

Select numEtud from examen where note < 10;

Q23:

Cours : query-proc\_optim.pdf, page : 137

Tq : V(A,R) : le nombre de valeurs distincts de l’attributs A dans la relation R.

Q24 :

Le meilleur plan d'exécution est celui qui fait les sélections le plus tôt possible : Vrai

Q25 :

Donnez un exemple d'opérateur qui ne permet pas de faire du pipeling avec les autres opérateurs du plan d'exécution :

Tri, recherche max et min, élimination des doublons, calcul moyenne et somme, partionnement sont des opérateurs bloquants.

Q26 :

Expliquez une situation où la matérialisation d'un résultat intermédiaire peut améliorer les performances : pas trouvé

Q27 :

L'ordre des jointures d'évaluation peut influencer le résultat d'une requête : faux

Q28 :

Select id\_soiree, count(\*) as total from participle where diguisement=’citrouill’ group by id\_soiree having total>=100;

Q29:

Hash Anti Join (cost=219.74..226.33 rows=16 width=5) :

* 219.74 : le coût de démarrage
* 226.33 : le coût total pour l’exécution si toutes les lignes sont récupérées.
* 16 : l’estimation de nombre de ligne
* 5 : la taille attendue de la ligne

Q30 :

Br+NR+BT

Q31 :

L'ordre des jointures d'évaluation peut influencer le temps de calcul du résultat d'une requête : Faux

Q32 :

La commande EXPLAIN estime le nombre d'enregistrements renvoyés par une requête. Ce nombre est toujours supérieur ou égal au nombre réel d'enregistrements renvoyés : Vrai

Explain Analayze donne une valeur exacte égal au nombre réel d’enregistrement.

Q33 :

Donnez un exemple de situation où l'algorithme de merge join est plus avantageux que celui de hash join :

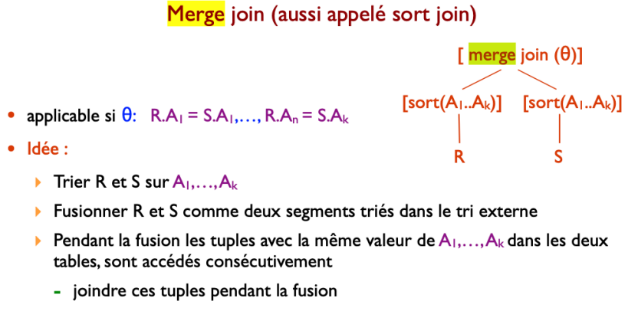
Un exemple de situation ou on utilise un MERGE JOIN => s’il y’a plusieurs jointures sur le même attribut, dans tous les autres cas on utilise un hash join.

Q34 :

Etant donnée une table R(a,b,c), quel(s) index peu(ven)t être utile(s) dans l'évaluation de la requête select a from R where b>10 : Index btree sur b

Q35 :

Expliquez en quelques lignes le principe de l'algorithme de jointure "merge join" :



Q36 :

select a,b from R where c=523 : Seq scan

Q37:

select a from R where a<150 : bitmap scan

Q38:

select a,b from R where a=11 : index scan

Q39:

select \* from R where b=10 : seq scan

**// RESTE A VERIFIERRR ET A CORRIGER …..**