

[Tableau de bord](#) / [Mes cours](#) / [INF8480 - Systèmes répartis et infonuagique](#) / Laboratoires Automne 2020 / [Quiz semaine 11 du 09/11](#)

Commencé le	mardi 10 novembre 2020, 22:14
État	Terminé
Terminé le	dimanche 15 novembre 2020, 19:38
Temps mis	4 jours 21 heures
Points	15,67/16,00
Note	19,58 sur 20,00 (98%)

Description

Quiz concernant : conclusion module 8, lectures et conclusion module 9

Votre note est disponible immédiatement à la fin du quiz, mais la correction est disponible uniquement après la fermeture du test.

Les questions à choix multiples disposent de réponses fausses à points négatifs.

Question 1

Terminer

Note de 1,00 sur 1,00

La cohérence causale

Veuillez choisir au moins une réponse :

- ☒ a. Les mises à jour non reliées causalement peuvent être effectuées de manière concurrente, sans ordre imposé par rapport aux autres mises à jour.
- ☒ b. Dans ce modèle, on suppose que toutes les écritures qui ont pu affecter une nouvelle écriture seront propagées avant cette nouvelle écriture.
- ☐ c. La cohérence causale est un concept intéressant mais impossible à réaliser en pratique.
- ☒ d. Des messages de groupe pour les mises à jour des réplicats, avec ordonnancement causal, utilisant des vecteurs de compteurs, pourraient être un mécanisme qui assure une telle cohérence.

Question 2

Terminer

Note de 1,00 sur 1,00

Les méthodes de synchronisation

Veuillez choisir au moins une réponse :

- ☒ a. La méthode de Christian est un algorithme simple pour synchroniser l'heure d'un client sur l'heure d'un serveur.
- ☒ b. Une horloge GPS, qui produit un signal de synchronisation Pulse Per Second, est un très bon moyen pour synchroniser précisément l'heure d'un ordinateur, par exemple pour un serveur de temps primaire sur Internet.
- ☐ c. La méthode de Berkeley est un algorithme simple pour synchroniser l'heure d'un client sur l'heure d'un serveur.
- ☒ d. La méthode de synchronisation utilisée dans le protocole NTP est semblable à l'algorithme de Christian, à la différence qu'on soustrait le temps passé sur le serveur pour répondre. Ceci permet de ne conserver idéalement que le temps passé sur le réseau comme indertitude.

Question 3

Terminer

Note de 1,00 sur 1,00

L'exclusion mutuelle

Veuillez choisir au moins une réponse :

- ☐ a. L'exclusion mutuelle en anneau est sécuritaire, vivace et respecte l'ordre premier arrivé, premier servi.
- ☐ b. L'exclusion mutuelle par serveur central est à éviter car elle est moins efficace et plus fragile que celle en anneau.
- ☒ c. L'exclusion mutuelle assure qu'un seul processus peut effectuer une certaine opération pendant l'intervalle où il possède le verrou associé.
- ☒ d. Un système d'exclusion mutuelle est vivace si aucun client ne peut voir sa requête attendre indéfiniment, alors que les autres réussissent à passer avant lui.

Question 4

Terminer

Note de 1,00 sur 1,00

Pourquoi utiliser la réplication dans un système réparti?

Veuillez choisir au moins une réponse :

- ☒ a. Pour rapprocher géographiquement les données des clients.
- ☒ b. Pour augmenter la disponibilité du système.
- ☒ c. Pour augmenter la performance, avec des accès en parallèle aux réplicats.
- ☐ d. Pour augmenter l'entropie des systèmes et assurer une diversité.

Question 5

Terminer

Note de 1,00 sur 1,00

Réplicats permanents et transitoires

Veuillez choisir au moins une réponse :

- ☐ a. Les mises à jour sur les réplicats sont généralement plus facile à paralléliser que les lectures car les lectures imposent plus de contraintes avec la réplication.
- ☒ b. Deux serveurs en miroir constituent des réplicats permanents
- ☒ c. Un système élastique, en infonuagique, crée des serveurs additionnels à la demande, à des fins de performance, et constitue des réplicats transitoires.
- ☐ d. La réplication, quel que soit le protocole, ne peut fonctionner qu'en l'absence complète de pannes.

Question 6

Terminer

Note de 1,00 sur 1,00

Exclusion mutuelle par envoi à tous

Veuillez choisir au moins une réponse :

- ☒ a. L'exclusion mutuelle par envoi à tous est un algorithme élégant où chaque participant est égal.
- ☐ b. L'exclusion mutuelle par envoi à tous est le meilleur algorithme d'exclusion mutuelle et est particulièrement avantageux lorsque le nombre de participants est très grand.
- ☐ c. L'exclusion mutuelle par envoi à tous ne peut pas fonctionner si toutes les horloges des participants ne sont pas parfaitement synchronisées.
- ☒ d. L'exclusion mutuelle par envoi à tous est un des pires algorithmes car il demande beaucoup de messages, la charge croît rapidement avec le nombre de participants, et son fonctionnement est compromis si le moindre participant tombe en panne.

Question 7

Terminer

Note de 1,00 sur 1,00

Election hiérarchique

Veuillez choisir au moins une réponse :

- ☐ a. L'élection hiérarchique ne fonctionne que s'il n'y a aucun participant en panne.
- ☐ b. L'intérêt de l'élection hiérarchique est qu'on peut garantir qu'il n'y a toujours qu'un seul élu, quelles que soient les pannes de noeud ou de réseau, étant donné qu'il y a un ordre pré-établi.
- ☒ c. Lorsqu'un participant ne parvient pas à contacter celui qu'il considère l'élu, il déclenche une élection et demande dans l'ordre aux participants plus prioritaires.
- ☒ d. Les participants ont un ordre de priorité pré-établi.

Question 8

Terminer

Note de 1,00 sur 1,00

L'état global d'un système réparti

Veuillez choisir au moins une réponse :

- ☒ a. L'état de chaque processus peut être vu comme ce qui résulte de son exécution depuis son initialisation, et peut être représenté comme l'ensemble des événements (messages) qu'il a reçus.
- ☒ b. L'état global est l'union de l'état de chaque processus du système.
- ☐ c. Pour obtenir l'état global d'un système réparti, il suffit d'envoyer un message à tous de mémoriser le numéro d'événement de la réception de ce message.
- ☒ d. Si on est prêt à arrêter tous les processus pendant un certain intervalle, il est assez facile de prendre l'état global du système pendant cet intervalle.

Question 9

Terminer

Note de 0,67 sur 1,00

Protocoles pour la réplication

Veuillez choisir au moins une réponse :

- ☒ a. Un serveur primaire qui reçoit toutes les écritures permet de les sérialiser et de s'assurer que le même ordre d'écriture arrivera sur tous les répliqués, si le serveur primaire propage, dans l'ordre, les écritures reçues.
- ☐ b. Un système de votation pour l'acceptation des mises à jour est plus complexe mais assure que le système peut fonctionner même en cas de pannes.
- ☒ c. Chaque client peut écrire ses modifications sur l'ensemble des répliqués. Cependant, pour assurer un ordre cohérent, il faut qu'il y ait un verrou qui assure qu'il n'y a pas de mises à jour concurrente des mêmes données, ou qu'il y ait un serveur primaire qui détermine l'ordre des mises à jour concurrentes.
- ☐ d. Les clients peuvent sans problème écrire sur le serveur de leur choix, en autant que le serveur note l'heure de la modification. Les serveurs répliqués se propagent ensuite les mises à jour, en tenant compte de l'heure de modification. Ceci assurera quand même une cohérence séquentielle aux clients, autant pour les lectures que les écritures.

Question 10

Terminer

Note de 1,00 sur 1,00

Distribution de contenu

Veuillez choisir au moins une réponse :

- ☐ a. Le mieux est d'envoyer toutes les mises à jour à tous les répliqués. C'est la méthode la plus performante et la plus simple.
- ☒ b. Le serveur peut envoyer un message d'invalidation à tout répliquat (incluant les clients avec cache) qui possède une copie qui était à jour de la variable qui est mise à jour. Ceci est utile lorsque les mises à jour sont très fréquentes et chaque client possède un petit sous-ensemble de valeurs à jour.
- ☒ c. Le serveur peut associer une date de validité à tout élément de donnée (e.g. entrée DNS ou page Web). Ceci est utile si les clients veulent grader ces informations en cache, les mises à jour ne sont pas trop fréquentes et chaque client utilise une très petite fraction des mises à jour.
- ☒ d. Le serveur peut propager les mises à jour à tous les répliqués, voire même à tous les clients. Ceci est utile si chaque récipiendaire est presque certain d'avoir besoin de toutes les mises à jour (par exemple les mises à jour de sécurité mensuelles pour les postes de travail).

Question 11

Terminer

Note de 1,00 sur 1,00

Les horloges logiques

Veuillez choisir au moins une réponse :

- ☐ a. Pour comparer l'ordre de deux vecteurs de compteurs d'événements, on compare la norme de chaque vecteur (la racine carrée de la somme des carrés comme avec la règle de Pythagore).
- ☒ b. L'horloge logique permet de déterminer un ordre mais ne donne aucune indication sur le temps écoulé entre deux incréments.
- ☒ c. En prenant la moyenne de plusieurs horloges logiques, on peut avoir une approximation du temps en milisecondes.
- ☒ d. Une horloge logique est tout simplement un compteur incrémenté à chaque événement significatif.

Question 12

Terminer

Note de 1,00 sur 1,00

Le temps pour un ordinateur

Veuillez choisir au moins une réponse :

- ☐ a. Avec un algorithme approprié, il n'y a aucune difficulté à synchroniser l'heure de deux ordinateurs avec autant de précision que désiré, en autant qu'ils soient connectés en réseau, quelles que soient les caractéristiques du réseau.
- ☒ b. Le circuit RTC est lu au moment du démarrage pour savoir l'heure.
- ☐ c. Les ordinateurs viennent avec une horloge précise, le temps exact est configuré en usine, au moment de l'assemblage, et il n'y a aucun besoin de corriger l'heure par la suite.
- ☒ d. Une fois démarré, l'ordinateur configure une minuterie pour se faire interrompre à intervalle régulier afin de tenir un décompte du temps.

Question 13

Terminer

Note de 1,00 sur 1,00

Groupement des mises à jour. Dans plusieurs cas, un groupe d'opérations doit être effectué de manière atomique sous forme de transaction, par exemple pour acheter 4 billets de spectacle adjacents.

Veuillez choisir au moins une réponse :

- ☒ a. Les verrous sont un mécanisme qui peut être utilisé pour assurer l'atomicité d'un groupe d'opérations.
- ☒ b. La notion de groupement est à la base de nombreux logiciels transactionnels, comme les bases de données, qui souvent supportent la réplication.
- ☐ c. Il n'y a pas d'utilité à grouper plusieurs mises à jour, il suffit de faire les mises à jour du groupe séparément.
- ☒ d. Les verrous sont un mécanisme robuste mais peuvent mener à des interblocages s'ils ne sont pas utilisés de manière rigoureuse.

Question 14

Terminer

Note de 1,00 sur 1,00

Cache dans les clients

Veuillez choisir au moins une réponse :

- ☐ a. Les clients ne peuvent jamais maintenir une cache, sauf s'ils sont enregistrés au niveau du serveur et ont l'assurance que le serveur va les notifier, quel que soit le modèle de cohérence qu'ils veulent offrir.
- ☒ b. Les clients du système de fichiers NFS utilisent une cache au niveau du client mais seulement pour quelques secondes, ce qui évite la plupart des conséquences perceptibles des problèmes de cohérence.
- ☐ c. Les caches au niveau du client, par exemple sur le Web, sont une fausse bonne idée car la plupart des pages Web ne sont lues qu'une seule fois alors qu'avec une cache il faut deux fois les accéder, une fois pour les afficher et une autre fois pour prendre une copie en cache.
- ☒ d. Les fureteurs Web utilisent souvent une cache au niveau du client.

Question 15

Terminer

Note de 1,00 sur 1,00

- Lectures et écritures comme ensemble ordonné (monotonie)

Veuillez choisir au moins une réponse :

- ☒ a. Si un client peut se connecter à différents serveurs répliqués, et que les mises à jour peuvent être retardées sur certains serveurs par rapport à d'autres, le client pourrait voir une nouvelle valeur pour une variable sur un serveur A et un peu plus tard voir une valeur plus ancienne pour la même variable sur un serveur B.
- ☒ b. La cohérence séquentielle implique une monotonie des lectures et des écritures.
- ☐ c. La monotonie des écritures implique nécessairement la monotonie des lectures.
- ☒ d. La monotonie des lectures veut dire qu'après avoir vu une nouvelle valeur pour une variable, un même processus ne pourrait plus voir une valeur plus ancienne par la suite.

Question **16**

Terminer

Note de 1,00 sur 1,00

La réplication synchrone, avec modification atomique de tous les réplicats. Ceci peut être effectué en utilisant des messages de groupe ordonnancés globalement, et envoyant un message de groupe atomique pour désactiver la valeur courante d'un élément de donnée répliqué. Ensuite, un second message atomique propage la nouvelle valeur. Il y a donc trois états en séquence pour chaque serveur: l'ancienne valeur est disponible, la donnée est indisponible, la nouvelle valeur est disponible. Il est alors impossible d'avoir en même temps un serveur qui offre la nouvelle valeur alors qu'un autre offre encore l'ancienne valeur.

Veuillez choisir au moins une réponse :

- ☒ a. Cette technique assure que deux serveurs ne peuvent donner en même temps une combinaison différente de valeurs disponibles (i.e. les données disponibles ont la même valeur).
- ☒ b. Cette technique peut assurer que toutes les modifications sont vues dans le même ordre, quel que soit le serveur auquel on se connecte.
- ☒ c. En cas de panne d'un réplicat, l'envoi de messages atomiques peut être problématique.
- ☐ d. Cette technique ne présente que des avantages car elle assure une grande cohérence, sans imposer de contraintes significatives pour les mises à jour.

[◀ Quiz semaine 10 du 02/11](#)

Aller à...

[Quiz semaine 12 du 16/11 ▶](#)