

Préparation Pour L'examen

I. Notion des systèmes répartis

Exercice N°1 : QCM

A. Besoin des SIs

- Interopérabilité des plates-formes de développement
- Gestion des accès concurrents
- Etat global

B. Les Eléments des applications répartis

- Processus et voies de connexions
- Seulement les Processus
- Processus et

C. Les éléments de mesure de qualité

- Sureté
- Vivacité
- Trafic engendré

Exercice N°2 :

1. Citer les différences entre système réparti et système centralisé

- Pas de référence temporelle
- Pas de référence spatiale
- Un réseau non géré par le SE

2. Citer les 4 caractéristiques du système repart

- PAS D'ETAT GLOBAL
- PAS D'HORLOGE GLOBALE
- FIABILITE RELATIVE
- SECURITE RELATIVE

3. Citer trois principes des systèmes répartis

- Concurrence d'accès aux ressources (possibilité de plusieurs accès à un même objet, même fichier, même table d'une base de données, ...)
- Disponibilité (duplication, réplication).
- Flexibilité (facilité d'utilisation et de configuration).
- Fiabilité – (N machines plus fiable qu'une seule!)

4. Citer 3 architectures des Systèmes répartis

- Client serveur
- code mobile
- agents

II. Système D'exploitation Répartis

Exercice N°1 :

1. Que doit fournir un système d'exploitation réparti ?
Un système d'exploitation réparti fournit aux utilisateurs l'accès aux divers ressources éloignées en:
 - ☐ Se connectant à la machine éloignée adéquate
 - ☐ Transférant les données de la machine éloignée à leur propre machine
2. Quels sont les types de système d'exploitation réparti ?
Monolithique et à micronoyau
3. Expliquer la différence entre les deux types des systèmes d'exploitation réparti en terme de nombre d'entrée sortie dans le cas d'un appel système.

☐ Appels systèmes

⇒ Noyau Monolithique: 2 entrées/sorties noyau

⇒ Micronoyau: 4 entrées/sorties noyau + 2 Commutations de contexte

4. Qu'est ce qu'un middleware ? préciser ses fonctions et donner un exemple.

Un middleware est un intergiciel qui assure l'interfaçage entre les processus communicants dans une application répartis

Fournir une interface (API) de haut niveau

Masquer l'hétérogénéité des systèmes matériels et logiciels sous-jacents

Rendre la répartition aussi invisible ("transparente") que possible

Fournir des services répartis d'usage courant

III. RPC et RMI

Exercice N°1 :

Quels sont chez le client et chez le serveur, les deux principaux acteurs ?

Talon client (stub) et talon serveur (skeleton)

La plupart des mécanismes d'appel de procédures distantes convertissent les paramètres et les réponses sous une forme canonique (ex, le format XDR-external Data representation- de Sun) Pourquoi ?

Considérant l'hétérogénéité du client et du serveur, il est nécessaire d'utiliser un langage commun de définition des paramètres qui masque cette hétérogénéité.

Les logiciels utilisant le mécanisme d'appel de procédures distantes ont en général accès à un serveur de noms. A quoi un tel serveur sert-il ?

Le serveur de nom est une sorte d'annuaire qui contient tous les services avec leurs paramètres de localisation et d'exploitation.