

Module 2 : RPC

- 2.1 Un service de vote permet de voter (candidat et numéro d'électeur), et d'obtenir le résultat (nom du candidat et nombre de votes). Donnez le prototype de ces fonctions. Identifiez les paramètres d'entrée et de sortie.
- 2.2 Les électeurs veulent être certains que leur vote soit pris en considération. Quelle sémantique peut les satisfaire?
- 2.3 Un réseau avec paquets qui peuvent être perdus est utilisé pour obtenir une sémantique au moins une fois. L'implantation peut ou non utiliser une hypothèse de délai maximal. Quel est l'effet sur l'implantation?
- 2.4 Comment peut-on générer un proxy pour une méthode comme Vote dans un langage comme C++ qui ne supporte pas la réflexion?
- 2.5 Un client prend 5ms pour calculer les arguments d'une requête RPC, le serveur prend 10ms pour traiter la requête. Chaque envoi ou réception de paquet demande 0.5ms au système d'exploitation et le temps d'envoi d'un message sur le réseau est de 3ms. L'encodage et le décodage des arguments prend 0.5ms par message. Quel est le temps pour effectuer deux requêtes si le client est séquentiel? Si le client est multi-fils?
- 2.6 Pour un système de ramasse-miettes réparti, les opérations AddRef et RemoveRef sont utilisées par les clients. Discutez-en les implications.
- 2.7 Faire une interface CORBA IDL pour un contenant de tâches qui se présentent sous la forme clé/valeur. Les opérations sont ajouter, enlever, lire.
- 2.8 Pourquoi CORBA et les systèmes similaires n'offrent-ils pas de constructeurs?