



**Forum: WD 10**

**Topic: application reseau**

**Subject: Re: application reseau**

Posté par: R&B

Contribution le : 3/12/2007 15:11:46

Bonjour Aviasoft.

Votre application monoposte, pour fonctionner en réseau, doit s'assurer que les ordres effectués par les utilisateurs n'entrent pas en conflit.

Le conflit majeur est la tentative d'écriture : deux utilisateurs tentent de modifier une même donnée à un même instant.

Si vous utilisez le RAD, il gère ce cas de la manière suivante : le blocage est effectué uniquement au moment de l'écriture (validation de la fiche). C'est une possibilité suffisamment admise pour avoir fait ce choix.

Maintenant, la plus grande différence se situe dans le déploiement de votre projet. L'installation 'avec mise à jour des données sur le réseau' consiste à déposer la base de donnée sur un serveur de fichier (un lecteur partagé du réseau) et les applicatifs sur les postes de travail.

En réalité, une copie de l'installation est déposée sur un lecteur partagé du réseau. L'installation des postes doit alors être effectuée depuis cette copie (et non le support qui a installé le 'serveur'). En effet cette copie comporte le renseignement qui indiquera aux versions 'poste de travail' la localisation des données sur le serveur.

Par conséquent il est important, pas obligatoire mais nettement plus pratique, que tous les postes partagent le même chemin sur le serveur pour accéder aux données.

La mise à jour est alors simple : on met, depuis un poste de travail, à jour l'installation serveur (l'installation des postes est alors remplacée sur le serveur), puis chaque poste lance la mise à jour (depuis celle sur le serveur). Le premier assure alors la mise à jour automatique des données.

Cette architecture comporte toutefois des limites :

- L'accès aux données passe par le réseau. Cela ajoute un risque considérable dans la communication exécutable-données. En effet un simple problème dans le réseau et la base de données se trouve dans une situation instable.
- le réseau est un vecteur 'lent' pour acheminer les données .

Longtemps, la seule alternative a été l'installation en mode 'Terminal Server' (Windows TSE ou Citrix). Là le serveur est à la fois serveur d'application et de données. Les postes se connectent au serveur via un bureau distant (protocole RDP). Si la solution est plus chère (dimension du serveur et licences spécifiques du service Terminal Server), elle est nettement plus robuste car seuls l'affichage et l'information de souris/clavier transitent sur le réseau (local ou distant). Le serveur est en charge tant de l'exécution du programme que de l'accès aux données qui sont alors locales.

L'installation du projet est alors une simple installation classique. La compilation doit simplement supporter l'exécution multiple sur le serveur. La partie réseau est alors dévolue au service Terminal

serveur.

Enfin, depuis Hyperfile C/S, il est possible de conserver l'exécution sur le poste de travail et la gestion de l'accès aux données sur le serveur. Le service Client/Serveur du moteur HyperFile C/S est piloté par un port TPC/IP sur le serveur en charge des communications entrantes/sortantes sur la base. De la même manière le serveur devrait comporter un serveur http/ftp pour que les postes de travail puissent mettre à jour l'exécutable. L'opération de mise à jour des données est effectuée lors d'une mise à jour de l'installation sur le serveur.

Bien cordialement.