

## SRCS : Systèmes Répartis Client/Serveur

### TP 06 – Construction d'un Cache

Janvier 2016

L'objectif de ce TP est de mettre en place un cache pour un serveur de fichier.

#### Exercice 1 : Construction d'un serveur de fichiers

##### Question 1

Programmer un objet serveur CORBA qui fournit un service de serveur de fichiers et qui implémente les méthodes suivantes :

**lire** : qui prend en entrée une référence de fichier et renvoie la donnée du fichier. Cette méthode lève une exception lorsque la donnée n'existe pas ou lorsque le serveur fait une erreur d'entrée sortie.

**ecrire** : qui prend en entrée une référence de fichier et une donnée, et enregistre la donnée localement. Cette méthode lève une exception lorsque le serveur fait une erreur d'entrée sortie. Si le fichier référencé existe déjà, son contenu est écrasé.

Pour simplifier la gestion, le serveur de fichier possède en mémoire une table de hash (voir ) associant les références et les données. Une référence est un nom de fichier (un wstring idl) et le contenu du fichier est une chaîne de caractère (un wstring idl).

Canevas de code à compléter : vous disposez d'une archive [cache.tar.bz2](#) contenant le canevas du code à produire.

## Exercice 2 : Construction d'un cache en lecture pour le serveur de fichier

### Question 1

Reprendre l'exercice précédent en faisant en sorte que le client utilise un cache en lecture. Vous définirez le cache comme un objet CORBA localisé chez le client.

Vous suivrez l'algorithme étudié en TD. Le client possède un cache en lecture. Toute lecture passe par le cache local. Les écritures, en revanche, sont envoyées sur le serveur de fichier. Lorsqu'une écriture a lieu sur le serveur de fichier, elle invalide toutes les copies en lecture des clients.

## Exercice 3 : Construction d'un cache pour le serveur de fichier

### Question 1

Vous allez maintenant construire un cache en lecture et écriture. Le serveur de fichier gère une exclusion de type lecteur écrivain. Il connaît la liste des lecteurs ou bien il connaît un écrivain.

Les règles suivies par le serveur sont les suivantes :

1. Lorsqu'une **lecture est demandée**, le serveur enlève le droit d'écriture à l'écrivain éventuel (qui devient un lecteur). L'écrivain envoie alors une version mise à jour du fichier au serveur. Ensuite, le serveur envoie une copie du fichier au lecteur. Le serveur enregistre le lecteur pour pouvoir invalider sa copie. Le lecteur peut alors manipuler le fichier en lecture tant qu'il ne reçoit pas une invalidation.
2. Lorsqu'une **écriture est demandée**, si il existait un écrivain, l'écrivain perd son droit en écriture et devient lecteur. Il envoie sa copie à jour au serveur de fichier. Ensuite, le serveur invalide alors tous les lecteurs et rend la main à l'écrivain. L'écrivain peut ensuite manipuler le fichier en lecture ou écriture tant qu'il ne perd pas le droit en écriture sur le fichier. Lorsqu'il perd le droit en écriture, il devient lecteur.