TP7 : CORBA

Bram De Wachter

23 mars 2005

1 Tetris : Précisions

Les classes requises pour la programmation du jeu Tetris se trouvent à l'adresse internet suivante :

`http ://www.ulb.ac.be/di/ssd/bdewacht/i152`

Les classes `Blocks` et `Display` permettent l'affichage et la gestion d'une partie. Dans le cadre de ce TP, ces deux classes ne doivent pas être modifiées. La classe `EventHandler` est la classe utilisée pour réagir à des événements importants. C'est cette classe qu'il va falloir modifier pour permettre des communications à l'aide de CORBA. Plus particulièrement, il faut implanter un interface, au préalable défini en idl, contenant les functions (i) `startGameWith` recevant la référence du joueur adverse en paramètre et permettant de débutter une nouvelle partie, (ii) `addScore` et `addLines` qui permettent de, respectivement, mettre à jour le score de l'adversaire et le nombre de ligne qu'il a effectué, (iii) `getName` qui renvoie le nom du joueur.
Pour rappel, il est également demandé d'implanter un serveur ayant pour but de distribuer les mêmes pièces aux joueurs lors d'une partie. Celui-ci devra implanter un interface contenant la fonction `connect` permettant à un joueur de s'enregistrer ainsi que la fonction `getNextBlock` permettant au joueur de demander la pièce suivante. C'est aussi le serveur qui mettra les joueurs ensemble en appelant leur méthodes `startGameWith`. Par simplicité, le serveur maintient un vecteur contenant des entiers. Chacun de ceux-ci correspond à un type de pièce possible. Un joueur désirant recevoir la pièce suivante n'a donc qu'a fournir l'indice dans le vecteur de la pièce qu'il désire au serveur.

2 Compilation

Pour compiler un fichier `idl` : `idlj -fall -oldImplBase <nom du fichier idl>`
Pour compiler les classes fournies : `javac *.java`

3 Lancemant d'un joueur

Après avoir lancé le *Naming service* (`tnameserv -ORBInitialPort 2000`), un nouveau joueur se lance en tapant : `java Blocks <nom du joueur> -ORBInitialPort 2000`