

TP CORBA n° 1

Installation d'ORBacus et présentation d'ORoBo

DESS SRO - Module "Objets Distribués"
UFR Sciences et Techniques - Université du Havre

Exercice 1 : Intallation d'ORBacus 4.1.2

A partir du site web

<http://www-lih.univ-lehavre.fr/~olivier/Enseignement/DESS>,

on téléchargera les 3 archives :

- source d'ORBacus pour C++
- source d'ORBacus pour Java
- documentation ORBacus

que l'on intallera.

Vous configurerez également votre environnement (CLASSPATH, LD_LIBRARY_PATH, ...) afin de pouvoir compiler et exécuter des programmes en Java et C++.

Exercice 2 : Projet "ORoBo", présentation et interfaces graphiques

Les TP de Corba suivront un fil directeur autour d'un projet qui se développera au fur et à mesure de l'introduction des nouveaux concepts.

Le cadre général est celui d'un jeu de robots déployé dans un environnement distribué. Différents joueurs gèrent chacun une équipe de robots qui se déplacent sur une grille bidimensionnelle correspondant à un "objet distribué" géré par un serveur distant. On va décrire une version basique du jeu avec quelques suggestions de variantes et d'extensions.

Pour fixer les idées, on considère la configuration adaptable suivante :

- La grille est un carré de 16 cases pouvant contenir chacune un seul robot ;
- Les joueurs jouent chacun leur tour ;
- Chaque joueur gère initialement 8 robots qu'il peut déplacer chacun d'un mouvement à chaque tour de jeu ;
- Les mouvements de base sont les déplacements dans une des 4 cases voisines latérales (pas de déplacement autorisé en diagonale) ;
- Il y a une phase d'initialisation qui correspond à l'une des 2 alternatives suivantes :
Soit le serveur place aléatoirement tous les robots de tous les joueurs sur le damier, soit chaque joueur pose alternativement un robot à la place qu'il veut sur le damier.

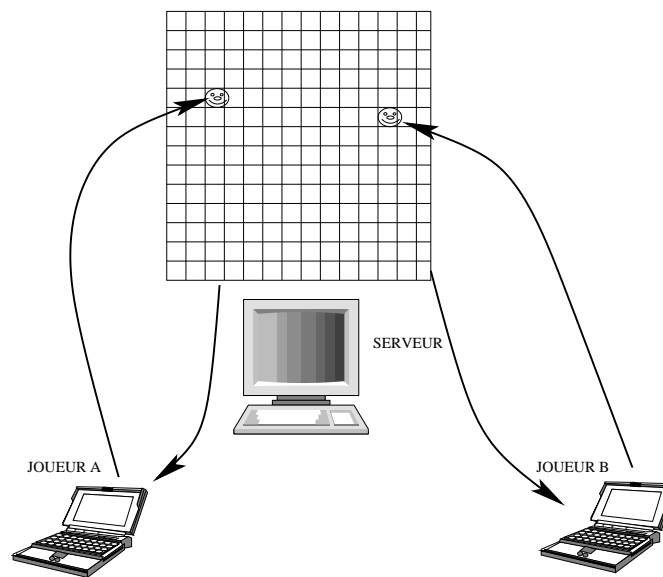


FIG. 1: OROBo - architecture distribuée

Le but du jeu est d'encercler les robots de ou des adversaires. Un robot dont les 4 cases adjacentes sont occupées par d'autres robots est ainsi encerclé. Il est détruit et supprimé du jeu.

Afin de préparer les TP suivants qui vont permettre de gérer le caractère distribué de la simulation, on devra développer les interfaces graphiques dont la configuration minimale est la suivante :

- Une fenêtre doit permettre de visualiser le damier avec les robots qui s'y trouvent ;
- Une fenêtre de jeux décomposée en champs de formulaires permet à chaque joueur de sélectionner un de ses robots et de proposer un déplacement.

La programmation de ce projet se fera en Java.

Remarques et extensions :

- Le jeu peut se dérouler avec deux joueurs ou plus. Dans ce dernier cas, les joueurs peuvent coopérer pour vaincre un adversaire commun. Les alliances entre joueurs peuvent changer au cours du jeu afin de mettre en place des stratégies qui permettent de maximiser les chances de dominer l'ensemble des adversaires.
- Une équipe de robots peut être gérée par un seul joueur ou encore par plusieurs joueurs qui manipulent chacun un robot ou un sous-ensemble de robots.
- A chaque instant, chaque joueur peut connaître l'ensemble des positions de tous les robots ou encore ne connaître que l'état des cases situées dans un voisinage de sa position actuelle. Dans ce dernier cas, si plusieurs joueurs gèrent une même équipe, ils peuvent s'échanger les états connus de la portion de damier qu'ils peuvent percevoir.