

# Module SyD – 2011 A

## Labo 1

On cherche à faire partager un même monde virtuel à plusieurs utilisateurs. Chaque utilisateur voit l'image de son environnement. Celle-ci évolue de façon interactive et en temps réel en fonction de ses déplacements (ici uniquement dans le plan horizontal). Cette image contient non seulement une image d'un décor mais également l'image des autres utilisateurs.

Pour cela l'utilisateur va disposer de sa propre représentation (position, angle, vitesse) mais aussi d'une représentation de chaque autre utilisateur (appelée avatar).

### Question 0 :

Lancer la simulation sur plusieurs postes (Adresses IP du serveur dans `diffuseur.py` et adresse IP des connexions dans `contexte.py` à changer ainsi éventuellement que la valeur des ports).

### Question 1

Modifiez le programme de façon à ce que chaque utilisateur diffuse son état à tous les autres sous la forme d'une chaîne de caractères. Cet état contient :

- la position de l'avatar
- l'angle donnant la direction dans laquelle se déplace l'avatar
- la vitesse de l'avatar

Affichez cette chaîne dans un terminal.

### Question 2

Chaque avatar est représenté par un pingouin. Faites en sorte que tous les utilisateurs aient une vision cohérente du monde (la position des avatars d'un utilisateur doit correspondre à la position de cet utilisateur).

Pour cela : à chaque réception d'un message concernant un avatar on modifie sa représentation.

Les avatars sont créés à partir d'un message émis par un nouvel utilisateur puis placés dans l'attribut `avatars` de la classe `contexte`. Il est possible d'ajouter un nouvel avatar ou de récupérer une référence à partir du nom de celui-ci (en vue de le modifier).

### Question 3

Afin de réduire le nombre de messages à transmettre on utilise un modèle prédictif linéaire. Il ne s'agit plus de diffuser sa position mais sa position, sa vitesse et son orientation. Il devient alors possible de prédire la position d'un avatar entre deux réceptions de message donnant son état. Montrer qu'une telle approche permet de réduire le nombre de messages de diffusion d'état.

### Question 4

Dans l'approche développée dans la question 3, les messages sont envoyés par un utilisateur A à un utilisateur B même si A reste immobile !! Imaginez une méthode dans laquelle A évalue l'erreur de prédiction chez les autres utilisateurs et diffuse son état lorsque cette erreur devient supérieure à un seuil fixé a priori.