

Architecture d'un jeu vidéo en réseau

Julien BERNARD

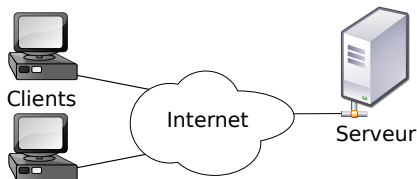
Dead Pixels Society
Université de Franche-Comté

version 1

Architecture

Un jeu en réseau est généralement constitué de deux types d'acteurs :

- le **serveur** qui va gérer l'état global du jeu
- les **clients** qui vont interagir avec le serveur pour modifier l'état du jeu



Côté serveur

- 1 Attendre les événements des clients (clavier, souris)
- 2 Mettre à jour l'état du jeu
- 3 Envoyer le nouvel état du jeu aux clients

Côté client

- 1 Envoyer les événements utilisateurs au serveur (clavier, souris)
- 2 Attendre le nouvel état du jeu
- 3 Dessiner une nouvelle image du jeu

Échanges entre clients et serveur

Protocole

- Nécessité de mettre en place un **protocole**
- Description précise des échanges entre clients et serveur
- Restriction du contenu des échanges au nécessaire !

Exemple (Ce qu'on peut échanger)

- Position et états des entités
- Événements du jeu

Exemple (Ce qu'on ne doit pas échanger)

- Ressources lourdes (images, sons, etc)

Multijoueur

- Une connexion par joueur à maintenir côté serveur
 - L'état du jeu à mettre à jour aussi souvent que possible
- Nécessité d'utiliser plusieurs *threads*

Temps réel

- Sur Internet, impossible d'échanger 60 fois par seconde avec le serveur
- Nécessité de prédire le futur état du jeu côté client