

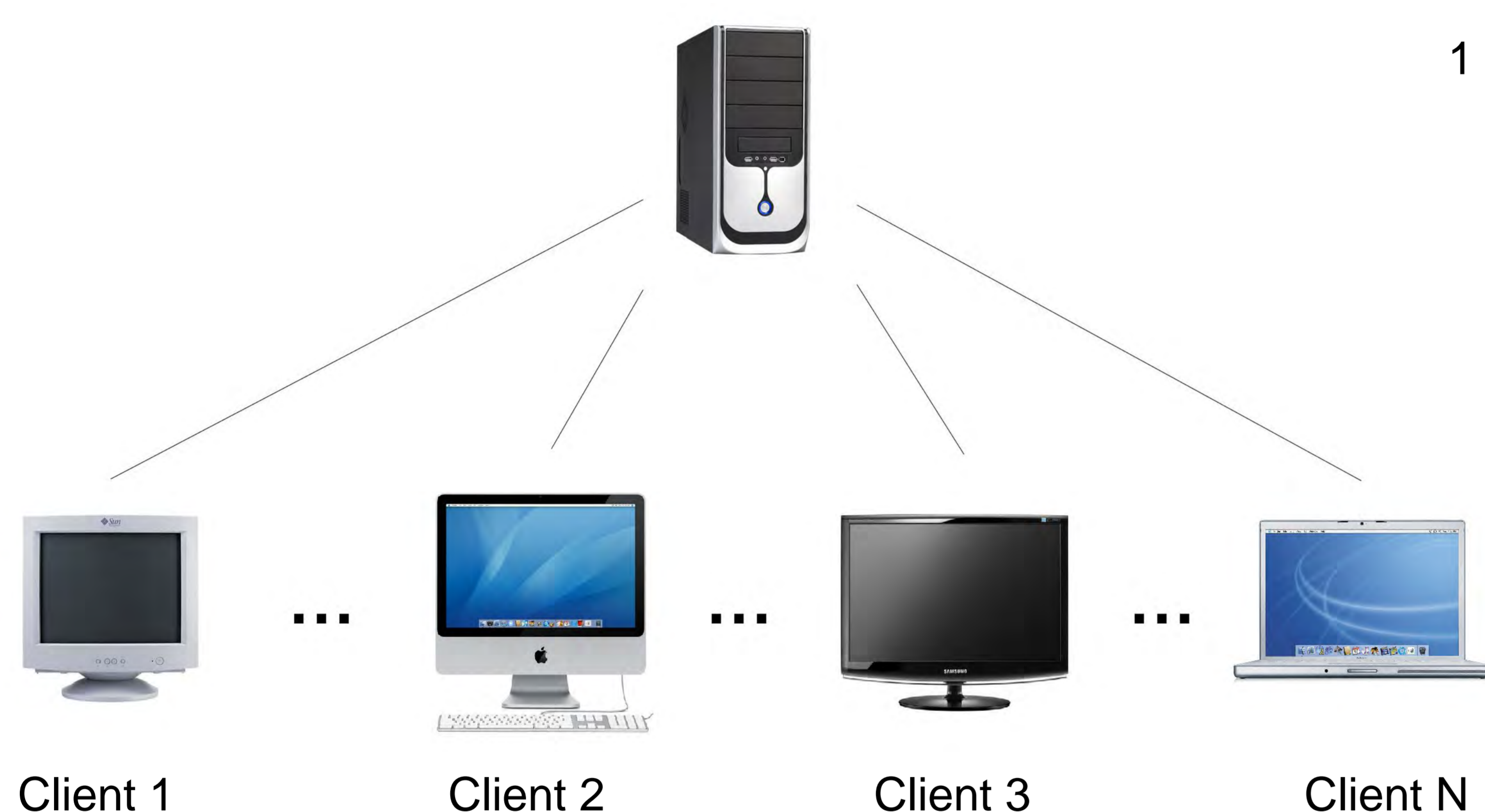
iGalaxy's Life : Architecture client-serveur

Aurélien LEBEAU, Laurent VLEMINCKX, Juraj GRIVNA,
Pedro MACHADO DOS SANTOS, Kévin GORIS, Pierre DAGNELY
Département d'Informatique

Création d'un jeu de colonisation spatiale multi-joueurs et temps réel

Architecture client-serveur

Ou comment relier les différents joueurs au jeu ?



1 serveur : → Contient le jeu
→ Applique les actions des joueurs au jeu
→ Envoie l'état du jeu aux joueurs toutes les x secondes

N clients : → Correspondent aux N ordinateurs des N joueurs
→ Affichent l'interface du jeu
→ Envoyent les actions des joueurs au serveur

Exemple simplifié de communication client-serveur

1

Le joueur Dark vador lance une action.
Ici il envoie un vaisseau C-collector sur une planète pour y collecter des ressources en donnant l'identifiant du vaisseau et les coordonnées de la planète où aller chercher les ressources.

3

Le serveur décompose la commande pour savoir quoi faire.
Et lance l'action demandée par le joueur.

Dans notre exemple, il verra qu'il doit envoyer le vaisseau numéro 6 sur la planète située à la position (4,42) pour qu'il y collecte des ressources.

2

Le client transforme l'action en une commande du type :
« login/Type de la mission/Paramètres de la mission »
Ce qui donne dans notre exemple :
« login/collecter des ressources/identifiantVaisseau/CoordonnéesDeLaPlanète »
Qui devient avec les informations du joueur :
« Dark Vador/Collecter des ressources/6/4/42 »

Il envoie ensuite cette commande au serveur.

4

Parallèlement à cela, le serveur envoie toutes les x secondes l'état du jeu, modifié suivant les actions des joueurs qu'il reçoit.

Avec notre exemple, les prochaines fois que le serveur enverra l'état du jeu à tous les clients/joueurs, ceux-ci verront petit à petit le vaisseau se déplacer dans l'espace vers sa planète de destination.