

## **SPRING\* : Provisionnement dynamique de serveurs pour les jeux massivement multijoueurs - approche par simulation**

Porteur : Sébastien Monnet (REGAL)

Equipes :

- REGAL (Réseaux & Systèmes) : Sébastien Monnet (MdC), Olivier Marin (MdC)
- SMA (DESIR) : Maha Abdallah (HDR)

### **Contexte du projet**

Les jeux multi-joueurs en ligne comme World of Warcraft sont devenus une classe d'application très populaire, et constituent une part considérable de l'industrie des loisirs électroniques. D'un point de vue système, ces applications sont très intéressantes car elles présentent des contraintes très fortes. Par exemple, le système doit être très réactif : si la latence pour l'envoi des mises à jours aux joueurs est trop forte, le jeu ne sera pas utilisable. De plus, il est nécessaire d'assurer, à grande échelle, la disponibilité du jeu et la cohérence des données tout en évitant la triche.

Les systèmes commerciaux de jeux en ligne reposent encore largement sur des architectures classiques de client/serveur en centralisé. De telles architectures se heurtent cependant au problème du passage à l'échelle, à la fois en nombre de joueurs et en nombre d'objets gérés pour modéliser des univers virtuels de plus en plus complexes. De nombreux jeux contournent le problème en limitant l'échelle : l'environnement virtuel est partitionné en plusieurs mondes parallèles totalement disjoints, comme par exemple les *royaumes* de *World of Warcraft*. Chaque partition, gérée de manière centralisée, n'accueille qu'un nombre limité d'utilisateurs (joueurs) et des avatars de joueurs créés sur des partitions distinctes ne pourront jamais se rencontrer. Afin de pouvoir gérer un nombre de partitions toujours grandissant, le déploiement des serveurs de jeu sur un cloud (Cloud Gaming) est en plein développement [6, 7].

### **Finalité du projet**

Le verrou scientifique innovant que nous souhaitons adresser est l'utilisation des architectures cloud pour résoudre le problème du passage à l'échelle des univers massivement multi-joueurs.

Avec l'avènement des clouds, il devient possible de concevoir des architectures dynamiques dans lesquelles on provisionne des serveurs en fonction de la charge courante. La manière dont on divise l'environnement virtuel en zones et la stratégie d'affectation des zones à différents serveurs a un impact important sur le coût (nombre de serveurs alloués) et sur la qualité de jeu. En effet, le nombre de joueurs et la densité de joueurs dans chaque zone varient beaucoup au cours du temps. L'architecture doit donc s'adapter continuellement.

Le but de ce projet est d'étudier l'impact de différentes stratégies de provisionnement de serveurs et d'affectation zone(s)-serveur(s). Pour ce faire, nous proposons de concevoir un environnement de

---

\* Sever PProvisionning for massively multi-player Games

simulation basé sur SimGrid (<http://simgrid.gforge.inria.fr/>) afin de représenter l'activité des serveurs et les trafics réseaux inter-serveurs et clients-serveurs. Le prototype prendra en entrée des traces de joueurs obtenues sur des jeux réels comme *World of Warcraft* [1] et les paramètres réseaux devront être réalistes, notamment la bande passante utilisée par l'application [2]. Le simulateur permettra de mesurer l'impact des différentes stratégies sur la qualité de jeu (données reçues ou non par les clients) et sur le coût (notamment en nombre de serveurs utilisés).

## Importance du projet pour le LIP6

Ce projet s'appuie sur l'expertise acquise au sein des équipes REGAL et SMA dans le domaine des jeux massivement multi-joueurs [3, 4, 5]. C'est pourquoi nous avons une grande confiance dans son potentiel fort de publication et de rayonnement scientifique.

Nous avons d'ores et déjà identifié des pistes de recherche pouvant être approfondies à partir de ce projet avec d'autres équipes du LIP6 ou d'autres universités : par exemple, la gestion décentralisée des données partagées mutables qui constituent l'univers de jeu.

## Annexe financière

### Coût global demandé: 5000 €

Stage M2 : " <i>Provisionnement dynamique de serveurs de jeux massivement multi-joueurs</i> "	2500 €
1 mission internationale	2500 €

## Bibliographie

- [1] Yeng-Ting Lee, Kuan-Ta Chen, Yun-Maw Cheng, and Chin-Laung Lei. 2011. World of warcraft avatar history dataset. In Proceedings of the second annual ACM conference on Multimedia systems (MMSys'11). ACM, New York, NY, USA, 123-128, 2011
- [2] P. Svoboda, W. Karner, and M. Rupp. Traffic Analysis and Modeling for World of Warcraft. In Proc. of ICC, Urbana-Champaign, IL, USA, June 2007.
- [3] Eliya Buyukkaya, Maha Abdallah and Gwendal Simon, A Survey of Peer-To-Peer Overlay Approaches for Networked Virtual Environments, in Peer-to-Peer Networking and Applications, 6:3(1-25), 2013
- [4] M. Véron, O. Marin, S. Monnet, Z. Guessoum : "Towards a scalable refereeing system for online gaming", Multimedia Systems, 2014
- [5] M. Véron, O. Marin, S. Monnet : "Matchmaking in multi-player on-line games: studying user traces to improve the user experience", Proceedings of the 24th International Workshop on Network and OS Support for Digital A/V (NOSSDAV), Singapore, 2014
- [6] OnLive - <https://www.onlive.com/>
- [7] NVIDIA Grid - <http://shield.nvidia.com/grid-game-streaming/>

## **Annexe : Sujet de stage**

### **Titre : provisionnement dynamique de serveurs de jeux massivement multijoueurs - approche par simulation.**

#### **Contexte**

Les jeux multi-joueurs en ligne comme World of Warcraft sont devenus une classe d'application très populaire, et constituent une part considérable de l'industrie des loisirs électroniques en termes de revenus.

D'un point de vue système, ces applications sont très intéressantes : elles présentent des contraintes très fortes. Par exemple, le système doit être très réactif : si la latence pour l'envoi des mises à jours aux joueurs est trop forte, le jeu ne sera pas utilisable. De plus, il est nécessaire d'assurer à grande échelle la disponibilité du jeu et la cohérence des données tout en évitant la triche.

Les solutions commerciales existantes simplifient le problème en limitant l'échelle pour revenir à des solutions centralisées. L'environnement virtuel est partitionné, soit en découpant un seul et même monde (à la manière des îles de Second Life), soit en créant plusieurs mondes parallèles disjoints (comme les royaumes de World of Warcraft). Chaque partition, gérée selon un mode client/serveur, n'accueille qu'un nombre limité d'utilisateurs (joueurs).

Il existe des travaux de recherche pour réunir tous les joueurs dans un même monde continu. Ces travaux proposent soit des solutions complètement décentralisées basées sur le paradigme pair-à-pair, soit des architectures distribuées passant à l'échelle côté serveur (Cloud Gaming).

#### **Objectifs du stage**

Nous sommes convaincus que la conception d'architectures passant à l'échelle pour les jeux multi-joueurs en ligne est un défi scientifique important. Avec l'avènement des clouds, il devient possible de concevoir des architectures dynamiques dans lesquelles on provisionne des serveurs en fonction de la charge courante. La manière dont on partitionne l'environnement virtuel en zones et la stratégie d'affectation des zones à différents serveurs a un impact important sur le coût (nombre de serveurs alloués) et sur la qualité de jeux. En effet, le nombre de joueurs et la densité de joueurs dans chaque zone varient beaucoup au cours du temps. L'architecture doit donc s'adapter continuellement.

Le but de ce stage est d'étudier l'impact de différentes stratégies de provisionnement de serveurs et d'affectation zone(s)-serveur(s). Pour ce faire, nous proposons de concevoir un environnement de simulation basé sur SimGrid (<http://simgrid.gforge.inria.fr/>) afin de représenter l'activité des serveurs et les trafics réseaux inter-serveurs et clients-serveurs. Le prototype prendra en entrée des traces de joueurs obtenues sur des jeux réels comme World-of-Warcraft [1] et les paramètres réseaux devront être réalistes, notamment la bande passante utilisée par l'application [2]. Le simulateur permettra de mesurer l'impact des différentes stratégies sur la qualité de jeux (données reçues ou non par les clients) et sur le coût (notamment en nombre de serveurs utilisés).

#### **Prérequis/compétences conseillées :**

Algorithmique distribuée, programmation en C ou Java (le coeur de SimGrid est en C, mais il offre un binding Java), notions de simulation.

#### **Bibliographie**

[1] Yeng-Ting Lee, Kuan-Ta Chen, Yun-Maw Cheng, and Chin-Laung Lei. 2011. World of warcraft avatar history dataset. In Proceedings of the second annual ACM conference on Multimedia systems (MMSys'11). ACM, New York, NY, USA, 123-128.

[2] P. Svoboda, W. Karner, and M. Rupp. Traffic Analysis and Modeling for World of Warcraft. In Proc. of ICC, Urbana-Champaign, IL, USA, June 2007.