

PROJET

Jeu type MMORPG de guerre Web pour Facebook apps : BGameSpace

REALISATION 2009

CEDRIC BEAUDIER

SOMMAIRE

1.	ORIGINE DU JEU.....	3
A.	DIFFERENCES	3
B.	LES AVANTAGES D'UTILISER FACEBOOK	3
2.	MATERIEL ET RESSOURCES.....	4
3.	CONCEPTION	5
4.	INTERFACE	6
5.	FONCTIONNEMENT.....	7
A.	EVENEMENTIEL	7
a.	<i>Par requête SQL ciblée.....</i>	7
b.	<i>Par cronstab.....</i>	7
B.	API FACEBOOK	8
C.	CALCUL DU COMBAT.....	8

1. ORIGINE DU JEU

Le jeu s'inspire énormément du jeu de GameForge™ : Ogame (www.ogame.fr). Ce jeu consiste à développer un empire militaire et économique dans l'univers, en colonisant jusqu'à 9 planètes, développant des technologies pour débloquent des vaisseaux ou bâtiments, construire des mines pour exploiter les ressources naturelles des planètes, construire des flottes de vaisseaux pour piller les planètes, ou détruire les flottes ennemies dans le but de récupérer les débris des vaisseaux pour en construire de nouveau. Chaque joueur peut faire partie d'une alliance, pour permettre aux joueurs de s'entraider et de faire des guerres d'alliances. Lors de grosses batailles, il y a une probabilité que le champ de débris généré crée une lune sur la planète en question, qui appartiendra d'office au propriétaire de la planète. La lune permet de construire une porte de saut (permettant de téléporter instantanément et gratuitement une flotte de taille illimitée mais sans pouvoir de transport de ressources, entre ses propres portes), et de pouvoir espionner les mouvements de flottes ennemies, tout en cachant ses propres mouvements de flottes. L'univers comporte 9 galaxies de 250 systèmes solaires de 16 planètes.

BGameSpace pour reprendre le nom OGame (une lettre majuscule suivie de Game), ici B pour Baud, puis Space histoire de ne pas trop coller avec OGame quand même. Consultez « Aide et règlement du jeu » sur mon site pour plus d'informations sur le déroulement du jeu et les règles très strictes.

A. DIFFERENCES

La plupart de ces caractéristiques du jeu ont été gardées dans BGameSpace. Voici les changements :

- une ressource supplémentaire,
- possibilité de créer des flottes (regroupement de vaisseaux),
- des pactes publics,
- possibilité d'avoir plus de planète (20 au maximum selon la race),
- plus de lune par planète (7 au maximum, mais quasi impossible à atteindre),
- des bonus spéciaux gagnés avec des « hauts-faits »,
- des races de spécialisation (défensif/économiste/minier, équilibré, offensif/pilleur/recycleur)
- l'univers comporte une map de 64000*64000 planètes, les coordonnées ne sont pas du même type.
- rapport de combat (RC) publics et infalsifiables,
- pas de limite de vol instantanés, mais limite de nombre de sondage, limite du nombre d'affichage du menu galaxie, limite du nombre d'envoi de flotte (attaque ou non) par jour,
- possibilité d'héberger des flottes alliées en orbite de sa planète,
- possibilité d'envoyer les flottes alliées en orbite pour une attaque groupée.

B. LES AVANTAGES D'UTILISER FACEBOOK

- Pas de gestion de login, mot de passe et compte
- Gestion de la sécurité et cookie simplifié
- Grâce à une limite minimale de « 5 amis », on limite les multi-comptes
- Des notifications Facebook sont envoyées lors d'événements du jeu
- Permet de communiquer en messagerie instantanée (ou pas) avec les autres joueurs
- Permet au joueur d'être tout le temps dans le jeu et sur Facebook en même temps

2. MATERIEL ET RESSOURCES

- 1 compte Facebook comme administrateur de l'application
- 1 application créée dans Facebook Developers
- 1 serveur OVH RPS 1 (Gentoo, Apache 2.2, PHP 5, MySQL 5)
- 1 base de données de grande taille (9 Go)

COPYRIGHT Beaudier 2009

La base de données du jeu contient 40 tables (MySQL 5, moteur MyISAM, interclassement utf8). La table USER contient les informations sur les utilisateurs que ne contiendrait pas Facebook. La base de données a été orientée optimisation des performances et non pas performances de stockages (3^{ème} forme normale non respectée). Voici quelques règles de gestion :

- Le serveur possède une liste des joueurs bloqués ;
- Le serveur possède une liste informations pour bloquer les tricheurs ;
- Un user possède des dons qu'il a fait au jeu ;
- Un user possède des RC qu'il veut conserver et partager ;
- Un user possède des messages entre joueurs et des rapports générés ;
- Un user possède une liste de planètes favorites (accès rapide) ;
- Un user possède une liste d'amis (accès rapide) ;
- Un user possède une liste de personne à ignorer (pour les messages) ;
- Un user possède des technologies acquises et des technologies en cours de développement ;
- Un user peut avoir une guilde ;
- Un user possède des candidatures vers des guildes ;
- Une guilde possède des membres ;
- Une guilde possède des candidatures de membres ;
- Une guilde possède des droits pour chaque membres ;
- Une guilde possède des pactes ou des guerres en cours ;
- Une guilde possède des candidatures de pactes ou des guerres en attentes.
- Un user possède des planètes ;
- Une planète possède des caractéristiques dont le propriétaire ;
- Une planète possède des vaisseaux en orbite, et en construction ;
- Une planète possède des flottes en orbite, et en construction ;
- Une planète possède des lunes ;
- Une planète possède des bâtiments au sol, et des bâtiments en construction ;
- Une planète possède des défenses au sol, et des défenses en construction ;
- Une planète possède des ressources et des revenus ;
- Une lune possède des caractéristiques dont la planète mère ;
- Une lune possède des vaisseaux et des flottes en orbite mais pas en construction ;
- Une lune possède des bâtiments au sol, et des bâtiments en construction ;
- Une lune possède des défenses au sol, et des défenses en construction ;
- Une lune possède des ressources ;
- Une type flotte possède un propriétaire et une liste de vaisseaux ;
- Une flotte possède un type, un nom, et une liste des vaisseaux perdus ;
- Un vaisseau, bâtiment, défense, technologie possède des caractéristiques et des pré-requis ;
- Un user possède des vols en cours ;
- Un vol possède des dates, une route, une liste temporaire ;
- Une liste temporaire possède des vaisseaux et des flottes ;

4. INTERFACE

(À venir)

5. FONCTIONNEMENT

A. EVENEMENTIEL

a. Par requête SQL ciblée

A chaque demande de page, le script PHP effectue quelques requêtes SQL rapide et optimisées, se basant sur la date de fin prévue des événements, tel que :

- Arrivée d'un vol à destination, ou son retour ;
- Fin de la construction d'un bâtiment ;
- Fin de la construction de toutes les défenses ;
- Fin de la construction de tous les vaisseaux ;
- Fin de la construction de toutes les flottes ;
- Fin de la recherche d'une technologie ;

Si la date de fin prévue est inférieure à la date serveur actuelle, alors un événement est lancé : la construction est terminée, et l'objet créé est ainsi ajouté dans la liste des objets disponibles pour l'utilisateur.

Cette requête s'applique seulement pour l'utilisateur qui affiche la page (pour ses constructions), ou bien pour les constructions d'une cible, si celui-ci espionne/attaque quelqu'un.

La seconde méthode consiste à ne pas attendre la fin des constructions (notamment pour les défenses, vaisseaux ou flottes) qui ne sont pas lancées de façon unitaire mais souvent par paquet de dizaine ou centaine. La table qui stock les constructions en cours possède un champ début, fin, et intermédiaire. Ce champ intermédiaire correspond au temps de construction unitaire de la construction en cours. Lorsque le script s'exécute, et que toutes les constructions n'ont pas été terminées, mais seulement quelques unes, alors le second script prend le relais : il fait la soustraction « (heure présente – heure du début) / durée unitaire » et obtient le temps le nombre de construction déjà effectuée. Alors ces constructions sont ajoutées à la table concernée, et le nombre de constructions restantes est décrémenté ainsi que l'heure de début mise à jour.

b. Par crontab

Par crontab, plusieurs scripts PHP sont lancés sur le serveur à différents moments pour :

- Sauvegarder la base de données ;
- Remise à zéro des compteurs de vol ;
- Effacer les messages ayant plus de 20 jours et messages effacés ;
- Effacer les informations anti-triches de plus de 8 jours (prescription) ;
- Retirer les modes débutants pour les joueurs après 7 jours ;
- Augmenter la taille des lunes selon le champ de débris présent ;
- Diminuer les lunes de 1% ;
- Diminuer les champs de débris de 10% ;
- Supprimer les champs de débris inférieurs à 1000 unités ;
- Nettoyer les tables (supprimer les guildes vides, les utilisateurs inactifs, planètes abandonnées) ;
- Compte les points des joueurs ;
- Compte les points des guildes ;
- Calcul le classement général des joueurs ;
- Calcul le classement général des guildes.

B. API FACEBOOK

L'API Facebook permet de se connecter aux bases de données de Facebook via une interface fournie en PHP. Cette interface est très lourde, et j'ai retiré ce dont on n'avait pas besoin pour le fonctionnement, dans le but de gagner de précieuses millisecondes par page.

Les pages sont demandées par les serveurs Facebook apps, et non pas par le client lui-même. Facebook connaissant des problèmes de performances sur leurs serveurs, la qualité de service est moyenne. D'autant que leurs serveurs décortiquent les pages pour retirer tout le code qui pourrait être dangereux pour récupérer des informations, exécuter du code malicieux ou simplement cacher les publicités de Facebook.

Cette API permet néanmoins de :

- Que le joueur est unique, logué et présent sans utilisation d'un script ou d'un bot ;
- Détecter si le joueur possède au moins 5 amis ;
- Promouvoir le jeu sur les murs, de partager et d'en parler sur Facebook ;
- Envoyer des notifications en temps réel (limite le stockage des messages dans la base de données) ;
- Permet aux gens de se rencontrer et discuter grâce au jeu, mais hors du jeu.

C. CALCUL DU COMBAT

Le calcul du combat est très complexe. Chaque vaisseau possède sa propre force de frappe, vitesse de frappe, choque et bouclier... Pour des combats de 200k vs 200k vaisseaux, il est inimaginable de pouvoir calculer plusieurs rounds d'affrontements avec au final 400k boucles...

La solution retenue a été de calculer une force de frappe totale des deux groupes, ainsi qu'un effectif total. Les diviser donne une frappe moyenne. En comparant la force de frappe moyenne de l'un, avec le bouclier moyen de l'autre, on obtient un résultat pour chaque round : est-ce que la frappe moyenne a traversé les boucliers moyens restants ? Si oui, alors ils attaquent la coque moyenne restante. Selon des coefficients aléatoires, la frappe moyenne détruira des vaisseaux adverses quand la coque moyenne descendra en dessous d'un seuil de 60% de vie. Selon le nombre d'effectif restant dans la frappe, un nombre légèrement inférieur de vaisseaux adverses seront donc détruits.

Au début du combat, les flottes sont démantelées en vaisseaux (en tenant comptes des vaisseaux manquants), pour n'avoir que des vaisseaux. Les technologies qui modifient les forces ou les boucliers sont appliqués, pour chaque membre (rappelons nous que plusieurs joueurs peuvent participer à l'attaque ou la défense) de la force d'attaque et de même pour la force de défense. A l'issue du combat, une liste de perte est tirée des deux côtés. Le jeu essaie de reconstruire les flottes initiales en partant des vaisseaux restants. Si les pertes représenteraient plus de 50% de la flotte que l'on essaie de reconstituer, alors la flotte est dissoute en vaisseaux.