



# Jeux en ligne et sociaux

# A propos du cours

- Maxime Montasheri
- Plan du cours
  - Ce matin:
    - Bases
    - Analyse du marché
  - Cet après midi:
    - Présentation de l'analyse
    - Concepts avancés
    - Gestion de carrière

# Concepts fondamentaux des jeux sociaux

Objectifs:

- Comprendre comment les gameplays des jeux sociaux sont construits

# Free-to-play

« There is no such thing as a free lunch. »

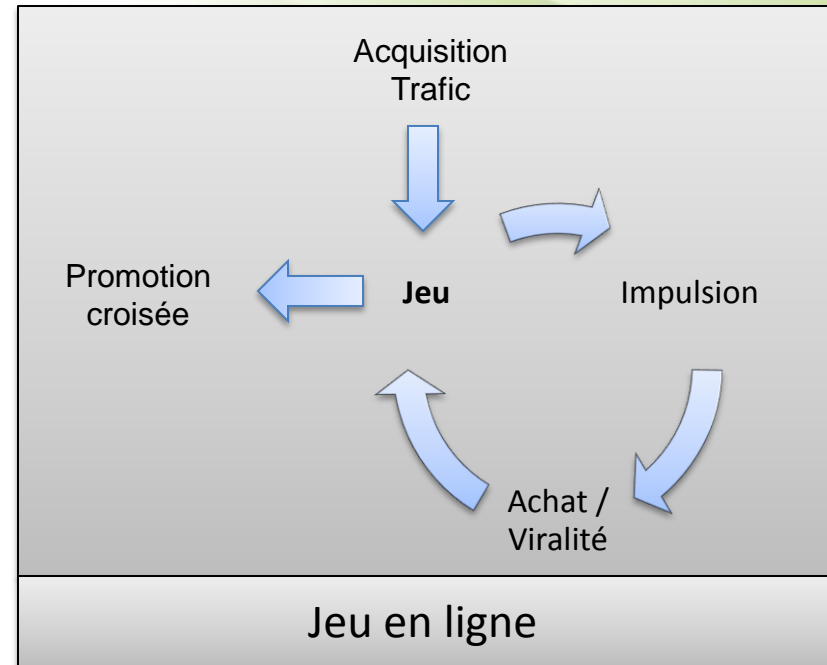
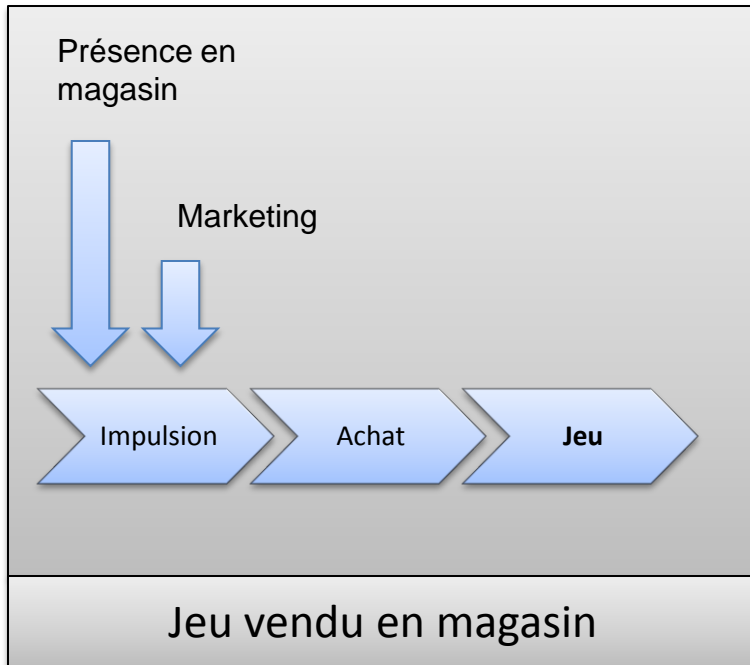
⇒ Free-to-play ≠ revenus nuls



## Monétiser la gratuité

- subsides directs (1 acheté = 1 gratuit)
- monétiser le trafic (publicité, par ex: TV)
- monétiser du contenu premium

# La révolution free to play



- **Jeu en ligne / social**

- L'impulsion d'achat est créée directement par le jeu
- Comme dans un casino, le paiement entre dans la boucle compulsive: Addiction = Monétisation
- Le trafic généré peut être monétisé indirectement

# Free-to-play

Marge = Nb Joueurs X Marge<sub>moyenne par joueur</sub>

**Marge =**

**( Nbj<sub>acquis</sub> + Nbj<sub>viraux</sub> )**

**X**

**( Revenus<sub>moyen vie du joueur</sub> – Coût<sub>acquisition</sub> )**

# Free-to-play

Optimiser l'acquisition

Optimiser la viralité

Marge =

$( \text{NbJ}_{\text{acquis}} + \text{NbJ}_{\text{viraux}} )$

X

$( \text{Revenus}_{\text{moyen vie du joueur}} - \text{Coût}_{\text{acquisition}} )$

Optimiser la durée de  
vie, maximiser la  
monétisation

Optimiser la  
promotion

# Acquisition – effet entonnoir

- Vous payez les clics qui amènent l'internaute sur votre page, mais vous risquez de le perdre !



- Login
- Droits d'accès
- Premier contact
- Tutoriel



# Durée de vie: les basiques

Plusieurs axes pour développer l'addiction

## 1. les mêmes ressorts que les jeux traditionnels

- Plaisirs de jeu, surprises, sensations viscérales
- Phases, rythme, émotions
- Progression & récompenses
  - Difficulté
  - Niveaux
  - Personnage, items
- Le « Almost »



# Rappels: Le core gameplay

- Le cœur de jeu est la mécanique principale du jeu, ex
  - Tetris: les pièces tombent et il faut les empiler au mieux
  - Tekken: frapper des ennemis pour les mettre KO
  - Mario: arriver au bout d'un niveau en sautant de plateforme en plateforme
- Un jeu peut comprendre plusieurs cœur de jeu, ex:
  - Uncharted: explorer des niveaux en 3D + FPS
- core(s) gameplay(s)  $\approx$  genre

# Rappels: Mécaniques de jeu

- Ce sont les règles qui constituent le **systeme** de jeu et délivrent le core gameplay, ex Tetris:
  - Une pièce sélectionnée aléatoirement tombe du haut de l'écran
  - 4 formes de pièces existent formées chacune de 4 carrés
  - le joueur peut tourner la pièce de 90° autant de fois qu'il le souhaite
  - la pièce se déplace par à-coups de case en case
  - ...
- Ce sont les règles qui constituent le **systeme** de jeu

# Rappels: Système de jeu

- Un bon système de jeu
  - Toutes les idées sont liées et renforcent un unique objectif: l'USP
  - Chaque idée a une raison d'être
  - L'ensemble est meilleur que l'addition des composantes
- Un système élégant
  - Peu de mécanismes, mais ensemble ils participent à créer richesse et profondeur
  - Simple à expliquer, simple à comprendre, facile à jouer
  - Un système élégant est généralement plus simple à produire

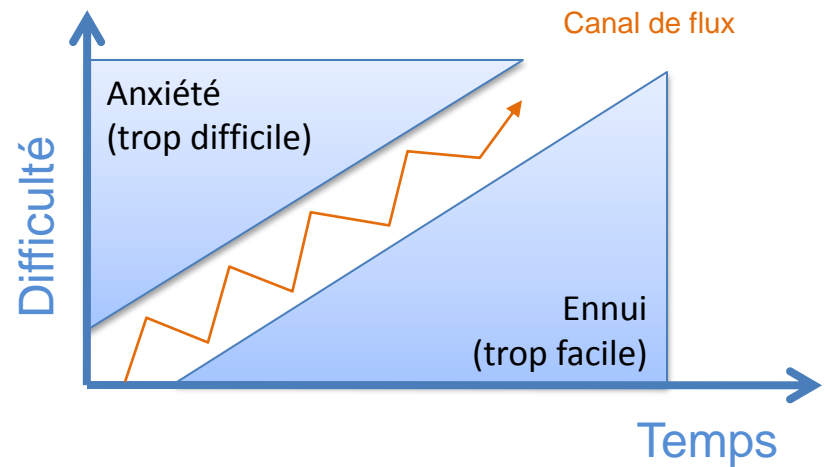


# Rappels: Phases de jeu, rythme, émotions

- Généralement le jeu passe par différents « états » qui s'enchainent
  - Quelles sont ces phases
  - Quel est l'objectif de chaque phase
  - Quel est le rythme pour passer d'une phase à une autre ?
  - Contraste entre les phases ?
- Emotions
  - Quelle émotion créer durant cette phase
  - Comment créer cette émotion/atmosphère
- Enrichir l'expérience et créant des phases
  - Simplifier/clarifier l'expérience en créant/supprimant de phases
  - Phases parallèles ou séquentielles ?

# Rappels: Difficulté

- Jeu trop difficile: le joueur s'énerve et finit par renoncer
- Jeu trop simple: le joueur ne trouve pas de challenge et s'ennuie
- Le joueur s'améliore au cours de la partie: augmenter le niveau de difficulté au cours du jeu



# Rappels:

## Récompenses, signes et feedbacks

- Opportunités de récompenser le joueur
  - A quel moment ?    Micro ↔ macro
  - Comment ?            Micro ↔ macro
  - En rapport avec l'objectif et la difficulté ?
- Signes
  - Comment le joueur va savoir ce qu'il doit faire ensuite ?
  - Comment donner envie au joueur d'agir ?
- Feedback
  - Comment indiquer au joueur qu'il a réussi ou échoué ?
  - Comment expliquer le fonctionnement du jeu au joueur de façon intuitive ?

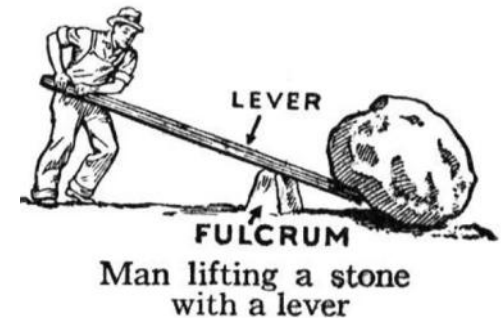
# Addiction - Nouveautés

## 2. Leviers sociaux

- La preuve sociale (tous mes amis jouent)
- La réciprocité (cadeaux)
- Personnalisation / valorisation sociale
- Comparaison par rapport aux amis

## 3. Gestion du temps

- Boucle principale limitée (énergie, temps de construction, production, etc.)
- Déblocage de la boucle à intervalles réguliers





# Addiction - Nouveautés

## 4. Renforcer le retour au jeu

- L'escalade d'engagement
- Prise de rendez-vous
- Perte de récolte



# Viralité

- Imaginons un jeu ou chaque joueur infecte en moyenne 0.5 joueur



Acquisition:  
10 joueurs

Viralité  
niveau 1:  
5 joueurs

Viralité  
niveau 2 :  
2.5 joueurs

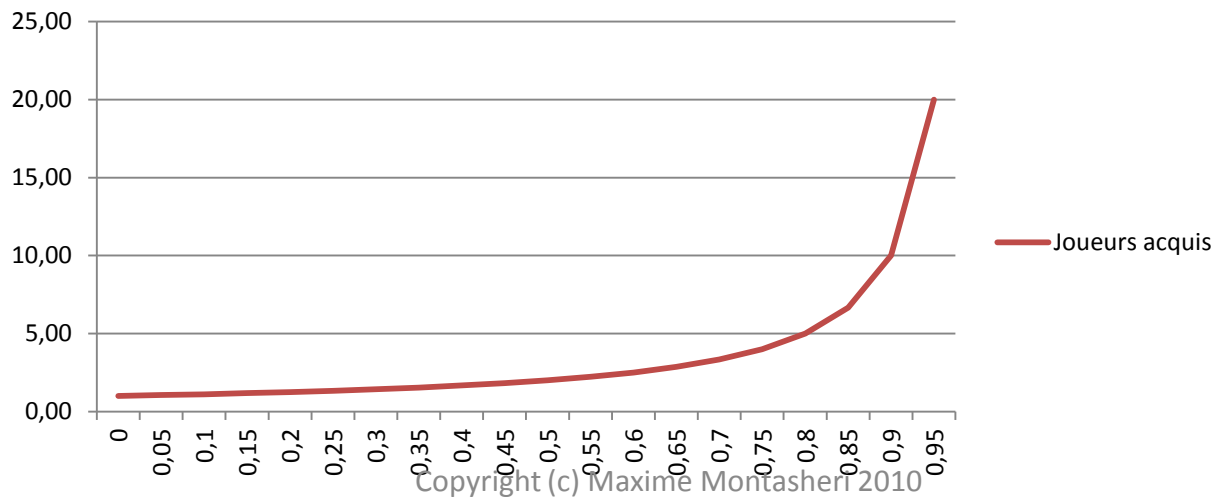
# Viralité

- Suite géométrique:

$$U_{n+1} = K.U_n \quad \text{où } K \text{ est le facteur de viralité}$$

- Comment calculer le total des joueurs acquis ?

$$\sum_{(n=0..∞)} U_n = U_0 / (1-K)$$



# Viralité

- Exemple: 1 joueur acquis à \$3
  - $K=0$       \$3/joueur
  - $K=0,5$     \$1,5/joueur
  - $K=0,75$    \$0,75/joueur
  - $K=0,8$     \$0,6 / joueur
- Que se passe-t-il lorsque  $K \geq 1$  ?
- **Conclusions**
  - Le facteur de viralité réduit les coûts d'acquisition
  - L'impact du facteur de viralité n'est pas linéaire d'où la force de ce levier



# Viralité: avenir

- Les messages viraux sont de moins en moins tolérés
  - Les non-joueurs se plaignent du spam
  - Mesures pour limiter ces messages
  - A terme: + contrôles , +bridages
- Les messages viral = communiquer comme sur le réseau social
  - S'exprimer
  - Demander de l'aide (pour quelque chose d'important)
  - ...

⇒ **L'expéditeur décide si le message mérite d'être partagé ou non !**

⇒ **Le destinataire décide de répondre au message ou non**

**K = taux d'infection \* taux de conversion**
- La viralité doit être incorporée au cœur de jeu



# Monétisation: les payeurs

- Un pourcentage réduit (2-10%MAU ou 1-3% DAU) de joueurs payeurs

⇒ qu'est ce qui justifie le paiement ?

- Rattraper ses amis
- Battre ses amis / rester en tête des joueurs
- Se différencier des autres
- Passer une frontière
- Eviter la frustration
- Récupérer une erreur
- Profiter d'une opportunité
- Gagner du temps
- Faire un cadeau à un ami
- Parier pour gagner de l'argent
- ...



# Monétisation: timing et format

- Moment
  - « Almost » (frustration maximale)
  - Accélérateur: quand le joueur ne peut pas attendre
  - Opportunité
  - Crainte puis soulagement
  - Passage clef (notamment extension de l'aire de jeu)
- Format
  - Monnaie premium





# Optimiser la promotion

- Promotion croisée

The image shows a Facebook interface for a game page. At the top is the Facebook navigation bar with the logo and a search box labeled 'Recherche'. Below this is a red-bordered banner containing several game advertisements: 'playfish', 'NEW BRES', 'MADDEN SUPERSTARS', 'RESTAURANT CITY', 'pet', 'Country story', and 'TRADE PLAYERS! FIFA'. Below the banner, there are several notification boxes: 'Game Installed!', 'Bookmark added!', and 'You like this game!'. To the right of these are buttons for 'Share Email' and 'Publish posts'. A green progress bar indicates 'Nearly there!' with a '60%' completion mark. Below the progress bar are buttons for 'Get the latest news!' and 'Publish posts faster!'. At the bottom of the page, there is a navigation bar with icons for 'Invite friends', 'Blog', 'Forum', 'Support', and 'My games:'. The 'My games:' section includes icons for a football, FIFA, a star, a fork and knife, a paw print, and a plant.

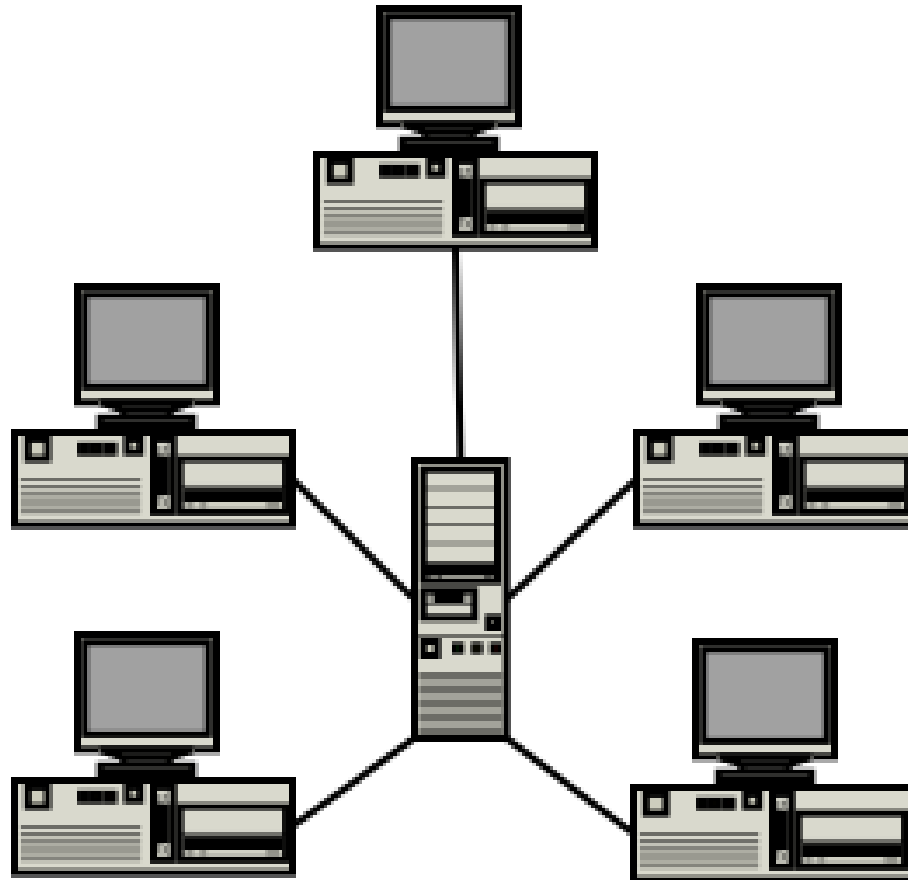


# Technologie des jeux sociaux

Objectifs:

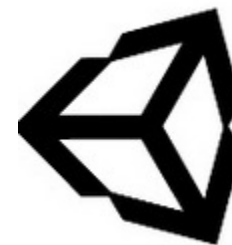
- Comprendre les contraintes imposées par la technologie derrière les jeux sociaux

# Applications client-serveur



# Technologies client

- HTML
  - Accumulation de technologies
  - Compatibilité explorateur
  - Pas besoin de plugin
- Flash
  - Cross platform
  - Performances moyennes, bonnes avec AS3
  - Technologie la plus adoptée pour les jeux sociaux
- Unity
  - Moteur 3D cross-platform + portage console
  - Faible parc installé
  - C # (basé sur Mono)



# Technologies serveur : Stack HTTP

- Stack HTTP
  - Latence = plusieurs secondes
  - Le serveur ne peut pas envoyer de message au client
  - PHP
    - Gratuit, bien connu, orienté BDD
    - Pas adapté au temps réel: Polling met le serveur à genoux
  - Développement de « Web services »



# Technologies serveur : Stack MMO

- Stack MMO
  - Latence dixièmes de seconde lorsque tout va bien
  - Le serveur peut solliciter le client
  - Serveur de sockets (Smartfox, Electroserver, ...)
    - Langage : généralement Java
    - Relativement complexe
    - Payant, licence dépend du nombre d'utilisateurs



# Latence et implications sur le gameplay

- Exemple pour un simple tchat (Au tableau)
- La plupart des jeux sociaux sont développés sur un stack HTTP
  - ⇒ Jeu multi-joueurs Asynchrone



# Bases de données

- Les données persistantes du jeu sont généralement stockées dans une base de données
    - Technologies utilisée le plus fréquemment: MySQL
    - Comment faire lorsque le serveur de bases de données est trop sollicité
      - Mettre les données en cache
      - Scinder la base
      - Consolider à intervalles réguliers
- ⇒ Les données doivent être découplées et dépliées au maximum



# Exploitation

- Les jeux tournent sur des serveurs qu'il faut maintenir
  - Stabilité / Plantages
  - Sauvegarde des données
  - Optimisation des ressources
  - Mises à jour système
  - Déploiement des updates
- Les mises à jour lorsque le système est en production sont très complexes



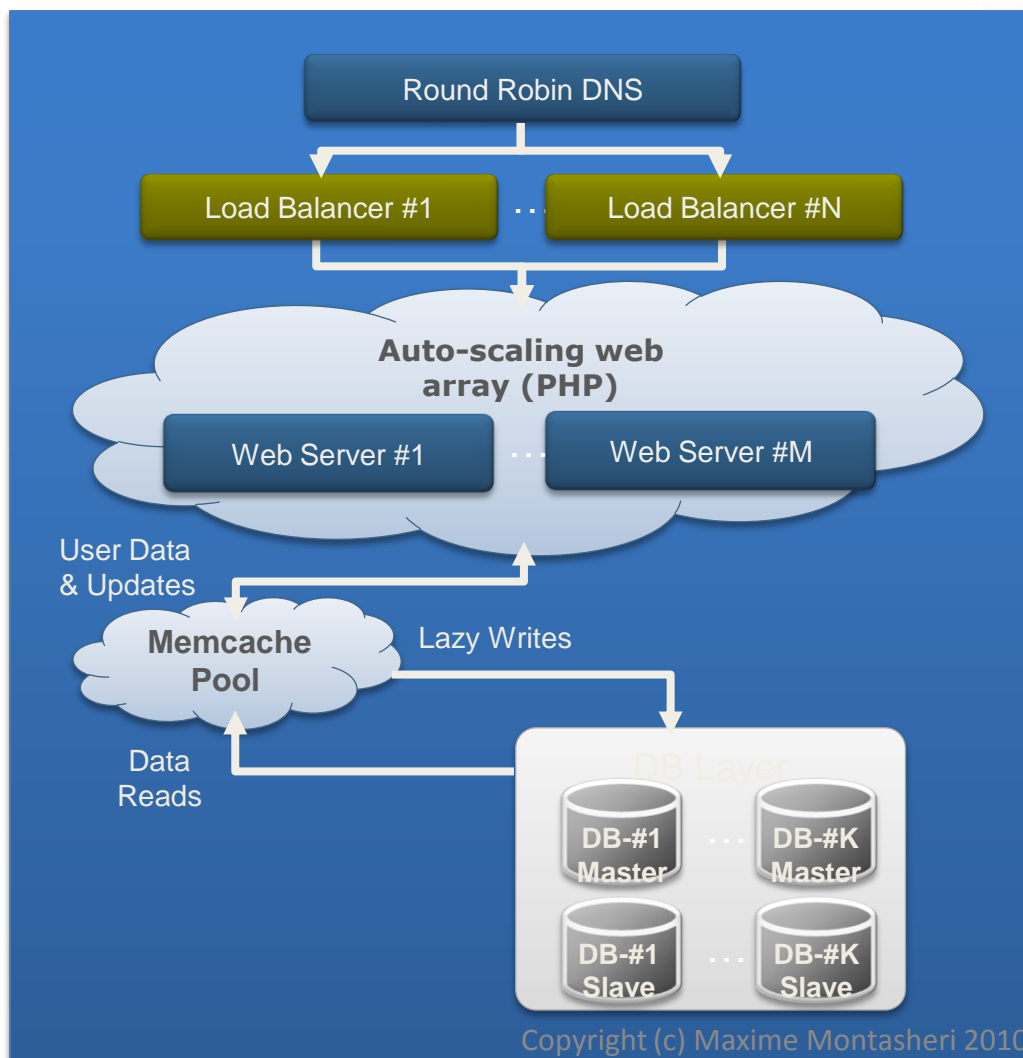


# Montée en charge

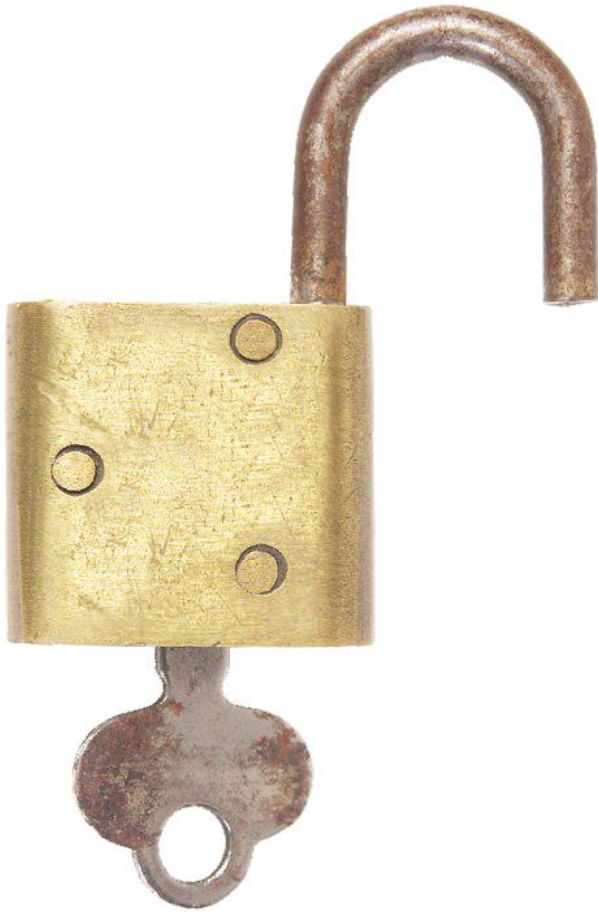
- Exemple Farmville



# Montée en charge



# Sécurité



- Application flash décompilée
- Les paquets trafiqués
- Failles dans le jeu ou jeu trafiqué  
exemple: ralentir son système pour avoir plus de temps de réflexion

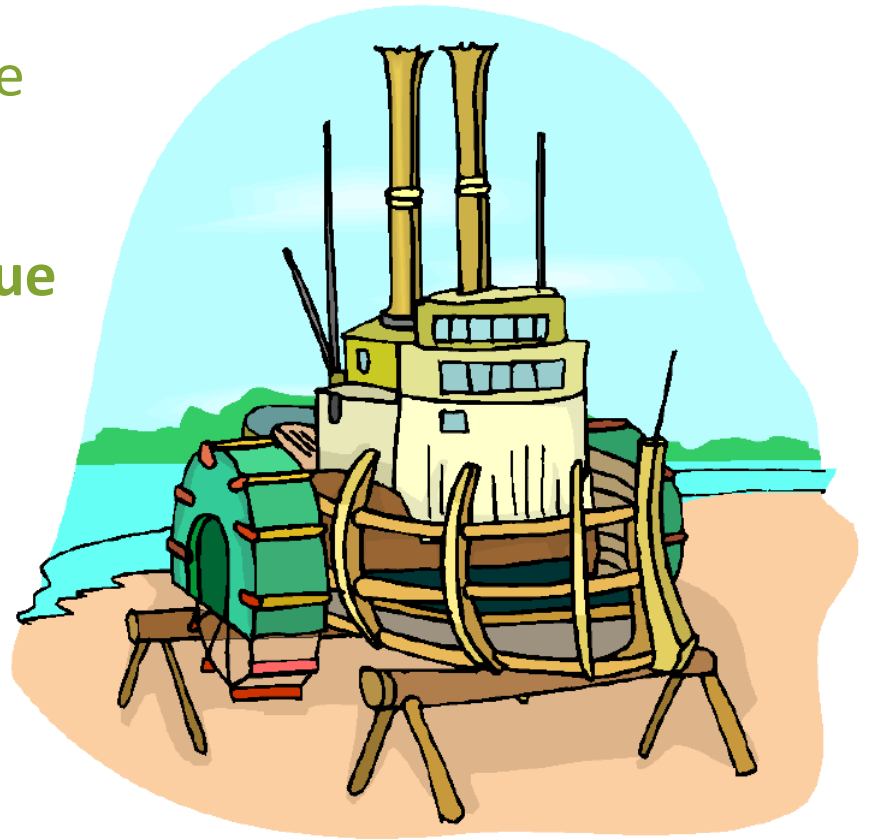
⇒ Le serveur ne doit jamais faire confiance au client

# Conclusions de cet aperçu technique

La technologie complexe

+ montée en charge très forte

⇒ **Contraintes très fortes sur le design pour essayer d'éviter que le bateau ne coule !!!**



# Evolution des jeux sociaux

Objectifs:

- Comprendre comment les développeurs font évoluer les jeux sociaux

# Le jeu social évolue

- Effort à fournir avant le lancement =  $1/3$  seulement !
- Ensuite
  - Identifier les problèmes non résolus et les corriger
  - Gérer la montée en charge
  - Faire des mises à jour pour maintenir l'intérêt



# Gestion de communauté

- Le jeu est en ligne: les développeurs peuvent recevoir du feedback direct
  - Blog
  - Forum
  - ...
- Des nouveaux postes de « gestionnaires de communauté » voient le jour
- Le développeur inclut les demandes les plus pertinentes dans sa roadmap de mise à jour



# Data mining

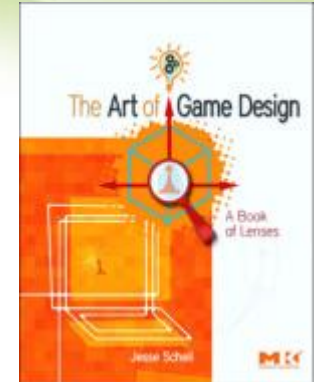


- Analyse des comportements des joueurs dans la BDD
  - Chiffrer pour valider des hypothèses
  - Comparer des versions différentes
  - Calculer des indicateurs pour mettre en évidence
    - Les incidents/anomalies
    - L'évolution (positive/négative) du service



# Suggestions de lecture

- The art of game design: a book of lenses – Jesse Schell



- 21<sup>st</sup> Century game design – Chris Bateman/Richard Boon



- Concevoir un jeu vidéo - Marc Albinet



- Articles de Pascal Luban sur l'AFJV



# Conclusion

- Principaux acquis du cours
- Questions
- Gardons le contact  
[maxime@montasheri.com](mailto:maxime@montasheri.com)
- Merci de remplir l'enquête de satisfaction

