

# Pair-à-Pair: Architectures et Services

Fabrice Le Fessant  
Fabrice.Le\_Fessant@inria.fr

Équipe ASAP  
(Réseaux très large échelle)  
INRIA Saclay – Île de France

Octobre 2008

## 1 Introduction

- Définition
- Caractérisation

## 2 Architectures

- Réseaux à serveurs
- Réseaux à inondations
- Tables de hachage distribuées (DHT)
- Réseaux épidémiques
- Réseaux sociaux

## 3 Services

- P2P versus Cloud

- 1 Introduction
  - Définition
  - Caractérisation

- 2 Architectures
  - Réseaux à serveurs
  - Réseaux à inondations
  - Tables de hachage distribuées (DHT)
  - Réseaux épidémiques
  - Réseaux sociaux

- 3 Services
  - P2P versus Cloud

Un système **pair-à-pair** ou **peer-to-peer** est un système d'échange de ressources entre utilisateurs.

## Exemple de ressources

- Le contenu : les fichiers présents sur la machine
- La bande-passante : messagerie/téléphonie, streaming audio/vidéo
- La puissance de calcul ou la mémoire : calculs scientifiques
- L'espace disque : sauvegarde croisée

## Un réseau **pair-à-pair** se caractérise par :

- Un ensemble de pairs s'échangeant des ressources
- Une **volatilité importante** des pairs (apparition/disparition imprévisible des pairs dans le système)
- Une **distribution géographique importante** (asynchronisme et communications non fiables)
- Des ressources limitées (mémoire, disque, bande passante, etc)

## Un système **pair-à-pair** vise à mettre en relation l'offre et la demande de ressources

- Faire émerger une organisation dans un tel réseau
- Maintenir cette organisation de façon pérenne

## 1 Introduction

- Définition
- Caractérisation

## 2 Architectures

- Réseaux à serveurs
- Réseaux à inondations
- Tables de hachage distribuées (DHT)
- Réseaux épidémiques
- Réseaux sociaux

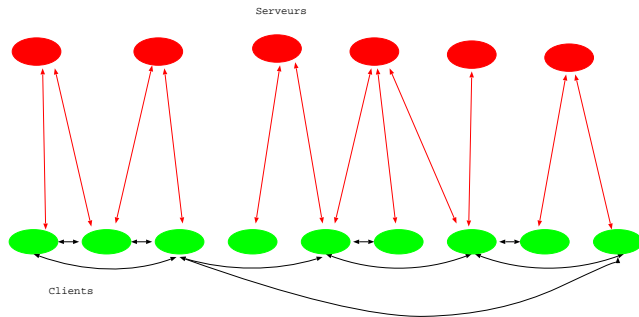
## 3 Services

- P2P versus Cloud

Comment organiser les liens entre pairs pour localiser facilement les ressources

## Chronologie

- Réseaux à serveurs [1999]
- Réseaux à inondation [2000]
- Tables de hachage distribuées (DHT)[2001]
- Réseaux épidémiques [2005]
- Réseaux sociaux [2006]



## Le vieux réflexe

- Napster : un serveur pour tout le réseau
- Edonkey/Emule : un réseau de serveurs
- Fasttrack/Kazaa/Skype : les *superpeers* ou serveurs auto-proclamés
- Gnutella : les *ultrapeers* qui filtrent les messages

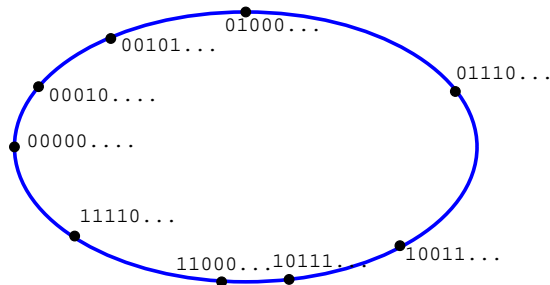


## La simplicité mais pas le passage à l'échelle

- Chaque pair se connecte au hasard à un petit nombre d'autres pairs
- Exemples : premières versions de Gnutella
- Les recherches se font par inondations
- Les résultats reviennent le long du chemin
- Compromis : diminuer l'inondation pour diminuer le coût diminue la probabilité d'obtenir un résultat.

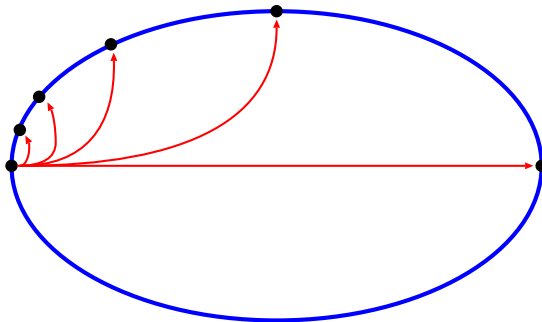
- Premier résultat académique dans le P2P
- Les pairs sont placés dans une organisation logique
- Le routage garantit une complexité limitée des recherches : souvent  $\log(N)$  pairs contactés par recherche
- Exemples : Overnet, Kad (Emule), Azureus DHT

# Fonctionnement d'une DHT (1)



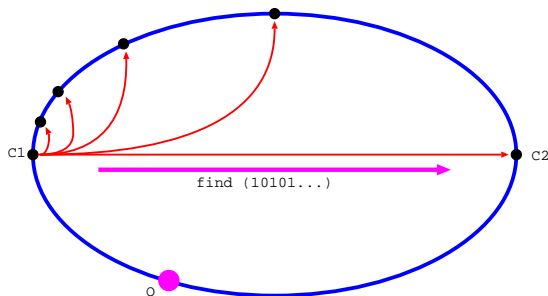
- Chaque pair a un identifiant choisi aléatoirement
- Cet identifiant le place dans une structure logique  
Exemple : un anneau orienté représentant l'intervall  $[0,1]$

## Fonctionnement d'une DHT (2)



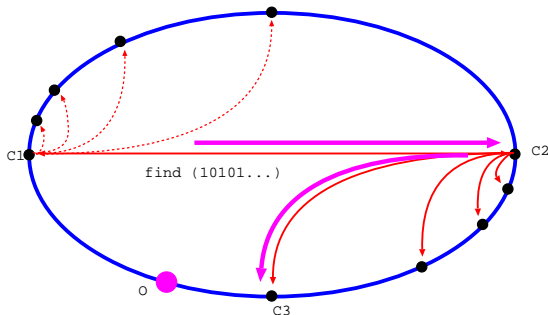
- Chaque pair établit une table de routage vers d'autres pairs
- Ici, les *fingers* de Chord à  $1/2$ ,  $1/4$ ,  $1/8$ ,  $1/16$ , etc...

## Fonctionnement d'une DHT (3)



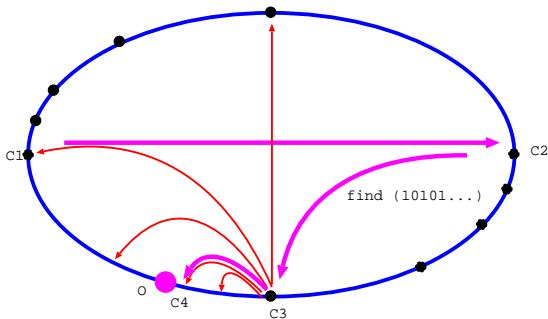
- Chaque ressource est aussi associée à un identifiant
- Pour chaque identifiant, la table indique quel pair est le plus proche.

# Fonctionnement d'une DHT (4)



- La requête est transmise de pair à pair...

# Fonctionnement d'une DHT (5)



- ... pour finir par atteindre le client le plus proche de l'identifiant
- Celui-ci est responsable des informations concernant les identifiants proches de son identifiant

- Chaque pair choisit ses voisins parmi ses connaissances en tentant d'optimiser un critère local
- Les pairs s'échangent leurs voisins (épidémies)
- Le système converge rapidement vers un état stable où chaque pair est dans un optimal local du critère
- Les recherches conformes au critère trouvent facilement des ressources localement



- Inspirés des réseaux sociaux sur le web (Facebook, Orkut, etc)
- Exemples : TribalWeb, Qnext
- Chaque pair choisit comme voisins des pairs qu'il connaît
- Toujours un sujet de recherche pour trouver des protocoles efficaces.
- Anonymat, sécurité et confidentialité

## 1 Introduction

- Définition
- Caractérisation

## 2 Architectures

- Réseaux à serveurs
- Réseaux à inondations
- Tables de hachage distribuées (DHT)
- Réseaux épidémiques
- Réseaux sociaux

## 3 Services

- P2P versus Cloud

- Partage de fichiers (...)
- Téléphonie (Skype)
- Vidéo à la demande (Bittorrent)
- Télévisions libres (Joost)
- Réseaux sociaux ( ? )
- Backup collaboratif ( ? )

# Peer-to-Peer versus Cloud Computing

## Deux approches opposées

- Cloud Computing : un service (payant) qui croît et décroît en fonction des besoins de ses utilisateurs (data-center)
- Peer-to-Peer : un service (gratuit) constitué des ressources fournies par ses utilisateurs

## Le Cloud Computing va-t-il tout résoudre ?

- Amazon, Google, Flickr, Facebook ont des centaines de millions d'utilisateurs
- Mais :
  - Logiciel Propriétaire -> Logiciel Libre -> Stockage Propriétaire
  - Pas d'interopérabilité (kidnapping des données), dispersion des données, pas de confidentialité (vie privée), boîtes noires (sécurité, autres services)