

# Applications Multimédia

Jean-Loup Guillaume  
Jean-loup.guillaume@lip6.fr



Laboratoire d'Informatique de Paris 6



CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UNIVERSITÉ  
PARIS-UNIVERSITÉS

# Le module APMM

- Aperçu des différentes phases du traitement des données multimédia :
  - Applications du multimédia.
  - Codage et décodage, compression
  - Structuration et description
  - Programmation multimédia
  - Diffusion de contenus multimédia



Laboratoire d'Informatique de Paris 6



CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UNIVERSITÉ  
PARIS-UNIVERSITÉS

# Le module APMM - Notation

- Contrôle continu 40% :
  - Rapport/code à rendre pour un TME
    - Monôme, binôme
- Examen 60% :
  - Correction des examens 2007 et 2008 pendant le dernier cours.



Laboratoire d'Informatique de Paris 6



CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UNIVERSITÉ  
PARIS-UNIVERSITÉS

# Me contacter !

- Par mail :
  - [Jean-Loup.Guillaume@lip6.fr](mailto:Jean-Loup.Guillaume@lip6.fr)
- Par téléphone :
  - 01.44.27.88.44 (en journée)



Laboratoire d'Informatique de Paris 6



CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UNIVERSITÉ  
PARIS-UNIVERSITÉS

## Programme

Type	Date	Lieu	Contenu
Cours 1	10/02	313 - 41	Introduction Compression son – images - vidéo
TD 1-2	10/02	313 - 41	Codage
Cours 2-3	12/02	313 - 41	Compression son – images - vidéo Systèmes - Programmation
Cours 4-5	13/02	203/205-41	Localisation des contenus CDN – P2P
TME 3-4	11/03	208 - 31	SMIL
TME 5-6	12/03	208 - 31	P2P
TME 7-8	13/03	208 - 31	JPEG

## Auteurs

- Des slides/images sont empruntés à :
  - Timur Friedman
  - Anne Fladenmuller
  - Autres auteurs
    - Wikipedia...

## Le multimédia

## Plan

- Introduction
- Numérisation / Compression
- Compression audio
- Compression image/vidéo

## Origines

- Terme apparu fin des années 1980, lorsque les supports de stockage se sont développés comme les CD-ROM.
- Applications pouvant utiliser plusieurs médias distincts simultanément :
  - Texte,
  - Musique, son, voix,
  - Images fixes ou animées,
  - Vidéos,
  - Interface homme-machine interactive.
- Désigne aussi un PC avec des composants multimédia :
  - Écran, carte son, webcam, etc.



Laboratoire d'Informatique de Paris 6

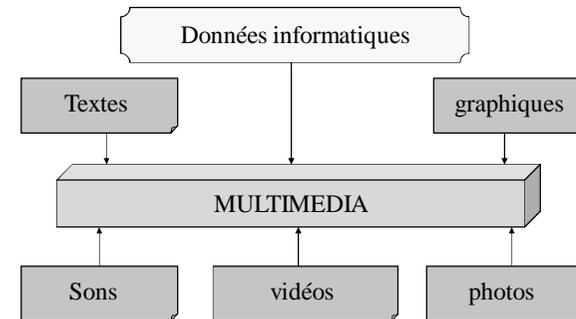


CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UNIVERSITÉ PARIS  
SUD - PARIS  
NORD

## Composants du multimédia



Laboratoire d'Informatique de Paris 6



CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UNIVERSITÉ PARIS  
SUD - PARIS  
NORD

## Applications/Outils

- Graphisme :
  - Infographie, dessin bitmap ou vectoriel, images de synthèse ;
- Vidéo :
  - Dessins animés, montage assisté par ordinateur ;
- Jeux vidéo
- Web
- PAO, CAO, etc.
- Enseignement assisté par ordinateur
- VoIP, Visio-conférence, etc.



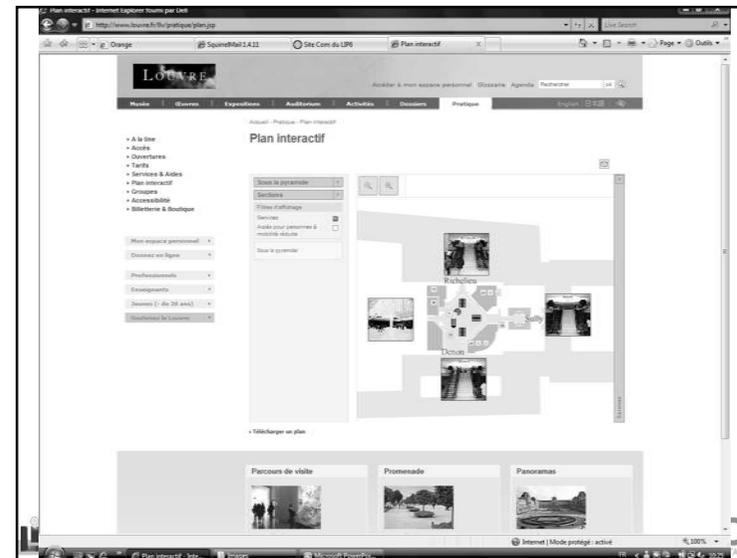
Laboratoire d'Informatique de Paris 6

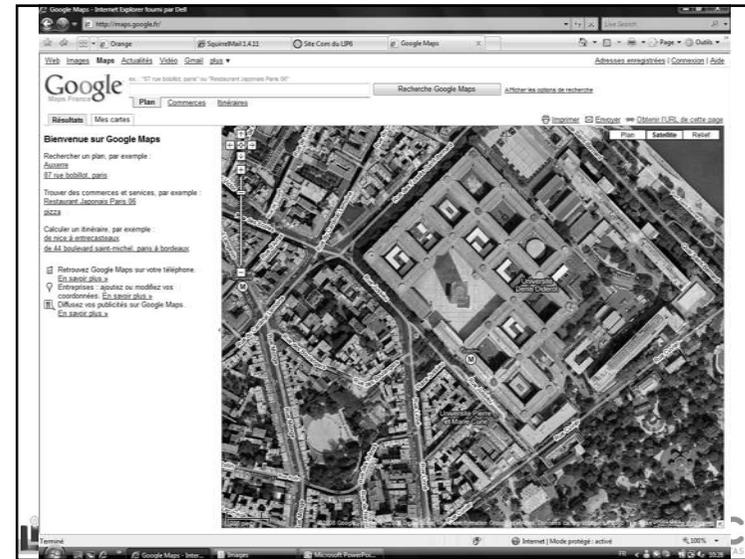


CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UNIVERSITÉ PARIS  
SUD - PARIS  
NORD





## Média passif

- Présentation indépendante des aspects temporels (entiers, réels, chaînes de caractères, textes ou images)
- En pratique, les données passives engagent des retardements pour leur présentation
- La temporalité (propriété mineure) n'a pas besoin d'être ajoutée au niveau modèle de données

Laboratoire d'Informatique de Paris 6

INRIA

UPMC PARIS UNIVERSITÉS

## Média actif

- Le temps détermine plusieurs propriétés du média.
  - Données actives (son ou vidéo) nécessitent des intervalles de temps pour leur présentation (ex. un temps de lecture pour sa restitution)
  - Références croisées qui portent sur un intervalle temporel valide (ex. sur une vidéo l'apparition et la disparition d'un personnage de la scène)
  - Contraintes temporelles pour la récupération des données pour l'affichage du média (ex. sur des données de type son, la récupération de deux échantillons consécutifs doit se faire à un certain seuil pour ne pas percevoir de distorsions lors de la restitution sur un haut parleur)

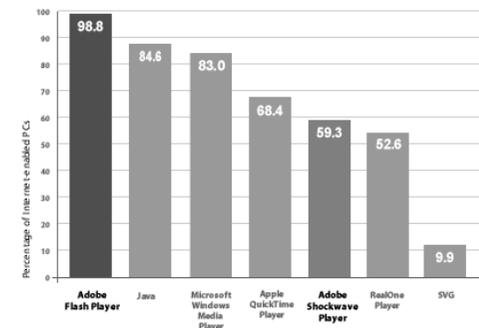
## Média adressable

- Possibilité de soumettre des requêtes sur son contenu.
- Interface homme-machine, accès non séquentiel
  - Recherche de mots dans un document texte,
  - Avancer, reculer, aller à un endroit précis, etc. sur un document vidéo,
  - Sur une image cela n'a pas de sens (média non adressable).
- Hypermédia : désigne l'utilisation de liens hypertextes dans le contenu.

## Comment lire du multimédia

- CD-ROM multimédia
- Téléphones 3G, PDA, etc.
- Télévision interactive
- Le web prend une place de plus en plus importante :
  - HTML permet déjà du multimédia/hypermédia
  - Norme HTML 5.0 en cours de rédaction (draft disponible)
  - Utilisation de Flash, SMIL permet d'incorporer vidéos, sons, interactivité.

## Lecteurs multimédia (12/07)

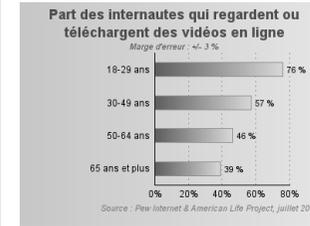
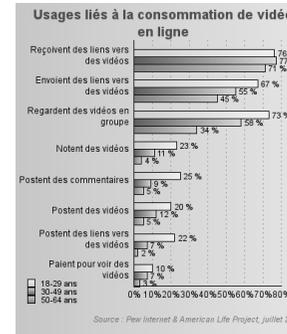


## Versions (12/07)

	Flash Player 6	Flash Player 7	Flash Player 8	Flash Player 9
Mature Markets <sup>1</sup>	98.8%	98.8%	98.3%	95.7%
US/Canada	99.0%	99.0%	98.5%	96.8%
Europe <sup>2</sup>	98.1%	98.1%	97.6%	94.3%
Japan	99.5%	99.5%	99.3%	95.3%
Emerging Markets <sup>3</sup>	97.4%	97.3%	95.5%	93.3%

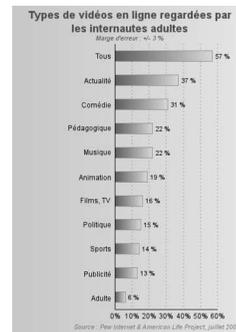
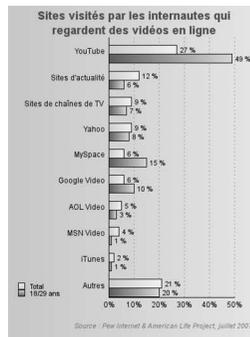
[http://www.adobe.com/products/player\\_census/flashplayer/version\\_penetration.html](http://www.adobe.com/products/player_census/flashplayer/version_penetration.html)

## Vidéo en ligne



Sources : Journal du Net

## Vidéo en ligne



## Numérisation/Compression sons / images / vidéos

### Partie 1

## Numériser

- Le multimédia, nécessite de structurer et synchroniser des contenus avec une possibilité d'interaction (liens hypertextes, recherche, navigation vidéo, etc.)
- Média très différents par nature :
  - Discrets et continus (son analogique)
  - Sonores et visuels
  - Fixes et animés
- Nécessité d'avoir un mode de représentation unique



Laboratoire d'Informatique de Paris 6



CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UPMC  
UNIVERSITÉ PARIS UNIVERSITAS

## Numériser

- Principe général : transformer des objets continus (à valeurs « réelles ») en objets discrets (à valeurs entière)
  - Son
  - Images
  - Vidéos
- Une représentation numérique binaire



Laboratoire d'Informatique de Paris 6



CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UPMC  
UNIVERSITÉ PARIS UNIVERSITAS

## Numériser

- Caractéristiques quantifiables d'un son ou d'une image
  - fréquence des ondes sonores émises dont la combinaison définit physiquement la hauteur et le timbre du son
  - l'amplitude des ondes qui en définit la « puissance »
- Ces deux données sont purement quantitatives, elles désignent un nombre de périodes par seconde (fréquence), et un écart mesurable par rapport à un point central d'une vibration (amplitude).



Laboratoire d'Informatique de Paris 6



CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UPMC  
UNIVERSITÉ PARIS UNIVERSITAS

## Compresser

- Action de transformer des données en une représentation plus compacte :
  - Occupent moins d'espace qu'avant compression
  - Il n'existe pas de compresseur absolu
- Pourquoi compresser ?
  - Espace disque
  - Temps de transmission sur un réseau (images, vidéo à la demande, etc.)



Laboratoire d'Informatique de Paris 6



CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UPMC  
UNIVERSITÉ PARIS UNIVERSITAS

## Un exemple simple

- Codage ASCII : 1 octet par caractère
- Fichier texte contenant 100 'a'
  - Taille du fichier : 100 octets
- Stockage sous la forme "100a"
  - Même information stockée
  - Taille du fichier : 4 octets
  - 4/100   taux de compression de 96%



Laboratoire d'Informatique de Paris 6



CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UPMC  
UNIVERSITÉ PARIS UNIVERSITAS

## Terminologie

- Compression sans pertes/codage réversible :
  - Les données originales peuvent être reconstituées
- Compression avec pertes/codage non réversible :
  - Les données originales ne peuvent être reconstituées exactement
  - Deux contenus différents peuvent être codés de la même manière
- Compression spatiale :
  - 2D ou une seule image
- Compression temporelle :
  - 3D ou vidéo



Laboratoire d'Informatique de Paris 6



CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UPMC  
UNIVERSITÉ PARIS UNIVERSITAS

## Compression sans pertes

- Retrouver les données originales en décompressant :
  - Gain d'espace sans perte d'information
- Utilisé si l'intégrité des données est importante :
  - Document texte, code, etc.
  - winzip, gzip, compress, bzip, GIF



Laboratoire d'Informatique de Paris 6



CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UPMC  
UNIVERSITÉ PARIS UNIVERSITAS

## Compression avec pertes

- Données originales perdues :
  - Taille réduite par l'élimination d'information
  - Choix d'un taux de compression plus ou moins préjudiciable au contenu
  - Faire en sorte que l'utilisateur ne s'en rende pas compte (compression dédiée)
- Applications :
  - Audio : mp3, ogg, wma, etc.
  - Images : jpeg, etc.
  - Vidéos : MPEG, « divx », etc.



Laboratoire d'Informatique de Paris 6



CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UPMC  
UNIVERSITÉ PARIS UNIVERSITAS

## Terminologie

- Chaque symbole à coder est représenté par un mot de code
- Un mot de code est composé de signes élémentaires
  - La valence est le nombre de signes élémentaires utilisés
  - Un code est binaire s'il est de valence 2



Laboratoire d'Informatique de Paris 6



CENTRE NATIONAL DE LA RECHERCHE SCIENTIFIQUE



UNIVERSITÉ PARIS UNIVERSITAS

## Codes de longueur fixe

- Codes de longueur fixe
  - Les mots de code sont tous de même longueur
  - Par exemple : ASCII, Unicode
- Supposons qu'il y ait  $n$  symboles à coder :
  - Nombre minimum de bits :  $\lceil \log_2 n \rceil$
  - Exemple avec 6 symboles : {a, b, c, d, e, f}
    - $\lceil \log_2 6 \rceil = \lceil 2.58... \rceil = 3$  bits par symbole
    - Un codage : a=000, b=001, c=010, d=011, e=100, f=101



Laboratoire d'Informatique de Paris 6



CENTRE NATIONAL DE LA RECHERCHE SCIENTIFIQUE



UNIVERSITÉ PARIS UNIVERSITAS

## Codes de longueur variable

- Les mots de codes ont des tailles variables selon les symboles
  - $L_p$  = nombre de mots de code de longueur  $p$
- Un code est :
  - intelligible s'il a un unique décodage
  - préfixe si aucun mot de code n'est préfixe d'un autre (il est alors intelligible) :
  - instantané si on peut le décoder au fur et à mesure de l'arrivée des mots de code (CNS : il est préfixe)



Laboratoire d'Informatique de Paris 6



CENTRE NATIONAL DE LA RECHERCHE SCIENTIFIQUE



UNIVERSITÉ PARIS UNIVERSITAS

## Exemples

	a	b	c	d	e	f
Fréquence	25	12	10	9	8	5
Code 1	01	10	100	101	1011	0011
Code 2	0	101	100	111	1101	1100

- Code 1 : 10101 = « bd » ou « da », pas intelligible
- Code 2 : préfixe (donc intelligible)



Laboratoire d'Informatique de Paris 6



CENTRE NATIONAL DE LA RECHERCHE SCIENTIFIQUE



UNIVERSITÉ PARIS UNIVERSITAS

## Codes complets

- Un code de longueur max n est complet si
  - Il est intelligible
  - Tout ajout d'un mot de code de longueur inférieur ou égale à n le rend inintelligible

- Égalité de Kraft-McMillan (CN)

$$\sum_{p=1}^n \frac{L_p}{V^{p-1}} = V$$

- $L_p$  = nombre de mots de code de longueur p



Laboratoire d'Informatique de Paris 6



CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UPMC  
PARIS UNIVERSITAS

## Codage compressif

- L'objectif est d'obtenir un codage compressif :
  - Intelligible : il est possible de décoder de manière unique
  - Efficace : les symboles les plus fréquents sont codés par des mots de code les plus courts



Laboratoire d'Informatique de Paris 6



CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UPMC  
PARIS UNIVERSITAS

## Entropie

- Donne une limite au codage compressif sans perte
- Soit n le nombre de symboles et  $p_i$  la probabilité d'apparition du i-ème symbole :

$$H = \sum_{i=1}^n -p_i \log_2 p_i$$

- $H$  = plus petit nombre moyen de bits utilisables (pour les codes binaires)
  - Exemple : {A,B} avec  $p(A)=0.6$  et  $p(B)=0.4$  :  $H = 0.97$  bits
  - Entropie maximale quand les symboles apparaissent avec même probabilité (moins prévisible)



Laboratoire d'Informatique de Paris 6



CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UPMC  
PARIS UNIVERSITAS

## Théorème du codage de la source

- Il existe un code préfixe avec une longueur moyenne à moins d'un bit de l'entropie
  - Exemple : {A,B} avec  $p(A)=0.6$  et  $p(B)=0.4$
  - $H = 0.97$  bits
- Il existe des codes aussi proches que l'on veut du minimum théorique :
  - Groupement de symboles nécessaire
- Efficacité nécessite de prendre en compte les fréquences d'apparition des symboles



Laboratoire d'Informatique de Paris 6



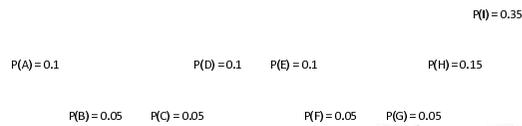
CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UPMC  
PARIS UNIVERSITAS

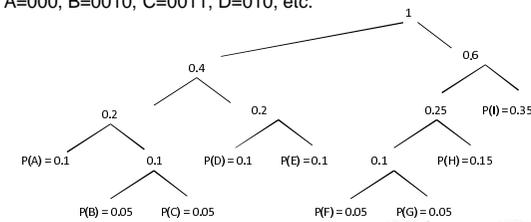
## Codage de Huffman

- Disposer les symboles par fréquence
- Connecter récursivement les deux nœuds de plus petite probabilité en créant les nœuds intermédiaires
- Mettre un label sur les branches
- En déduire le codage :
  - A=000, B=0010, C=0011, D=010, etc.



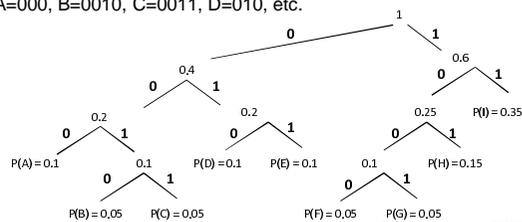
## Codage de Huffman

- Disposer les symboles par fréquence
- Connecter récursivement les deux nœuds de plus petite probabilité en créant les nœuds intermédiaires
- Mettre un label sur les branches
- En déduire le codage :
  - A=000, B=0010, C=0011, D=010, etc.



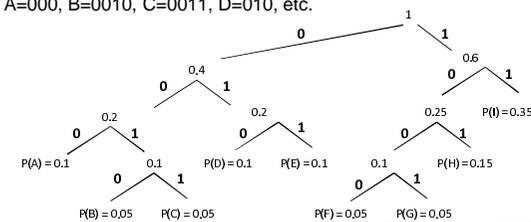
## Codage de Huffman

- Disposer les symboles par fréquence
- Connecter récursivement les deux nœuds de plus petite probabilité en créant les nœuds intermédiaires
- Mettre un label sur les branches
- En déduire le codage :
  - A=000, B=0010, C=0011, D=010, etc.



## Codage de Huffman

- Disposer les symboles par fréquence
- Connecter récursivement les deux nœuds de plus petite probabilité en créant les nœuds intermédiaires
- Mettre un label sur les branches
- En déduire le codage :
  - A=000, B=0010, C=0011, D=010, etc.



## Caractéristiques

- Le code utilise un nombre minimal de bits par symbole :
  - Optimal en utilisant des mots de codes
  - Peut être éloigné de la borne théorique
  - Optimal si les fréquences sont des puissances inverses de 2 : 1/2, 1/4, 1/8, 1/16...
- Le dictionnaire doit être envoyé avec le code
- Peut-être amélioré si les fréquences sont variables dans les données
  - Codage adaptatif



Laboratoire d'Informatique de Paris 6



CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UPMC  
UNIVERSITÉ PARIS UNIVERSITÉS

## Exercice

{A,B} avec  $p(A)=0.6$  et  $p(B)=0.4$

- A=0, B= 1 :
  - longueur moyenne =  $0.6*1+0.4*1 = 1$
- AA=00, AB=01, BA=10, BB=11 :
  - longueur moyenne = 1
- AAA=000, AAB=110, ABA=100, ABB=101, BAA=010, BAB=011, BBA=1110, BBB=1111 :
  - Longueur moyenne = 0.981

Et avec 4 caractères ?

Limite théorique = 0.97



Laboratoire d'Informatique de Paris 6



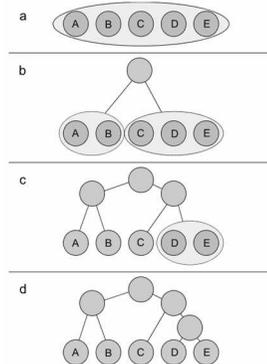
CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UPMC  
UNIVERSITÉ PARIS UNIVERSITÉS

## Codage de Shannon-Fano

- Calcul des fréquences d'apparition
  - A=15, B=7, C=6, D=6, E=5
- Tri puis séparation en deux ensembles de taille aussi proches que possible (et recommencer sur chaque ensemble)
- La branche de gauche est associée à 0, celle de droite à 1 (ou l'inverse)
  - A=00, B=01, C=10, D=110, E=111
  - Taille moyenne ~ 2.28b/s
  - Optimum ~ 2.18b/s



Laboratoire d'Informatique de Paris 6



CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UPMC  
UNIVERSITÉ PARIS UNIVERSITÉS

## Codage dictionnaire

- Typiquement LZ ou LZW :
  - Dictionnaire de symbole créé dynamiquement
  - Initialement le dictionnaire contient les 256 caractères ASCII
  - Fichier codé
    - Liste des adresses dans le dictionnaire
    - Le dictionnaire est reconstruit à la décompression



Laboratoire d'Informatique de Paris 6



CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UPMC  
UNIVERSITÉ PARIS UNIVERSITÉS

## LZW compression

w := NULL

**Tant qu'**il y a des caractères  
lire un caractère K (sur 8 bits)  
**si** wK existe déjà dans le dico

w := wK (1)

**sinon**

écrire code de w (/ 12 bits) ---- limite la taille du dictionnaire  
ajouter wK dans le dico

w := K (1)

**fin si**

**Fin tant que**

Écrire code de w (/ 12 bits)

## LZW Compression

/QED/QE/QEE/QEB

Entrée Lire	Sortie encodée	Code et valeur correspondante dans la table	Valeur de w après (1)
			/
Q	/	256 = /Q	Q
E	Q	257 = QE	E
D	E	258 = ED	D
/	D	259 = D/	/
Q			/Q
E	256	260 = /QE	E
/	E	261 = E/	/
Q			/Q
E			/QE
E	260	262 = /QEE	E
/			E/
Q	261	263 = E/Q	Q
E			QE
B	257	264 = QEB	B
FIN	B		

/;Q;E;D;256;E;260;261;257;B

## Conclusion

- Approches simples pour compresser :
  - Huffman
  - Shannon-Fano
  - LZW
- Sans pertes
- Ne nécessitent aucun a priori sur les données
  - Compression pas forcément élevée
  - Peu adaptés au contenus son/image/vidéo

## Compression audio

## Plan

- Introduction
- Numérisation
- Etendue dynamique
- Compression et codage



Laboratoire d'Informatique de Paris 6



CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UPMC  
UNIVERSITÉ PARIS UNIVERSITAS

## Transmission d'audio

- Téléchargement de fichier, lecture après
  - systèmes de partage de fichier
- En streaming : lecture pendant le téléchargement
  - Web
  - fonctionnalités de lecteur (avancer, reculer, etc.)
- Interactive
  - VoIP
  - musique collaborative



Laboratoire d'Informatique de Paris 6



CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UPMC  
UNIVERSITÉ PARIS UNIVERSITAS

## L'impact de l'interactif

- Non-interactif
  - on peut passer longtemps à coder
    - codecs qui sont asymétrique par rapport au temps nécessaire pour codage, décodage
  - si non-streaming : on peut passer longtemps à décoder aussi
- Interactif
  - codage et décodage doivent être rapides



Laboratoire d'Informatique de Paris 6



CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UPMC  
UNIVERSITÉ PARIS UNIVERSITAS

## Interactivité : trois niveaux

- Non-interactif non streaming
  - Codage long et décodage long possibles
- Non interactif streaming
  - Codage long possible
  - Buffer réduit en réception : décodage rapide
- Interactif
  - Codage et décodage très rapides
  - Contraintes temps-réel



Laboratoire d'Informatique de Paris 6



CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UPMC  
UNIVERSITÉ PARIS UNIVERSITAS

## Formats communs

- Musique
  - CD, DVD PCM
  - MP3, Ogg Vorbis
- Voix
  - A-law/ $\mu$ -law (téléphonie numérique de base)
  - GSM (téléphonie mobile)



Laboratoire d'Informatique de Paris 6



CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UPMC  
UNIVERSITÉ PARIS-UNIVERSITÄT

## Compression

- Sans pertes
  - Méthodes classiques (Huffman, LZW, etc.)
- Avec pertes
  - Méthodes basées sur les caractéristiques de l'audio et de l'ouïe
  - Plus efficaces / perte de qualité



Laboratoire d'Informatique de Paris 6



CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UPMC  
UNIVERSITÉ PARIS-UNIVERSITÄT

## Un signal audio analogique



Laboratoire d'Informatique de Paris 6



CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UPMC  
UNIVERSITÉ PARIS-UNIVERSITÄT

## Deux axes réels

- $t$  : temps
  - unité le plus fin (typiquement) :  $\mu\text{sec}$
  - échelle : milliers de sec
- $p$  : pression relative
  - unité le plus fin (typiquement) :  $20 \mu\text{Pa}$
  - échelle : environ 100 Pa



Laboratoire d'Informatique de Paris 6

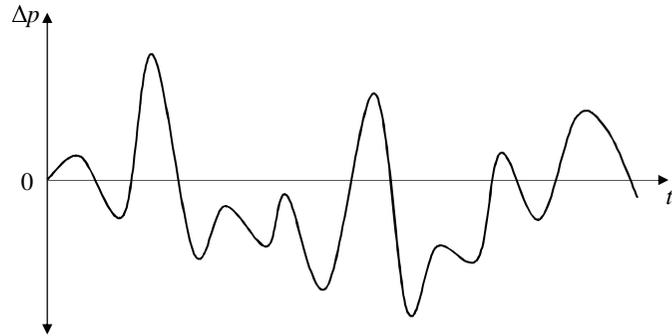


CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UPMC  
UNIVERSITÉ PARIS-UNIVERSITÄT

## Numérisation



Laboratoire d'Informatique de Paris 6

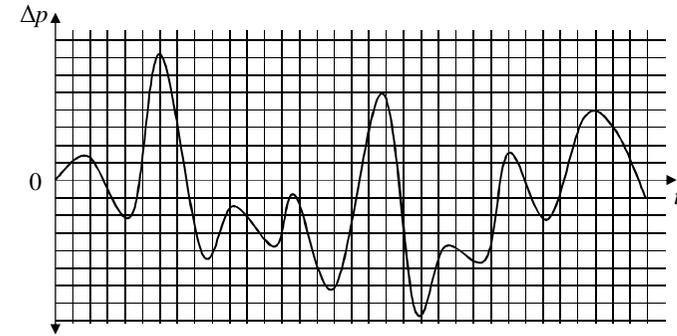


CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UNIVERSITÉ PARIS  
UNIVERSITAS

## Numérisation



Laboratoire d'Informatique de Paris 6

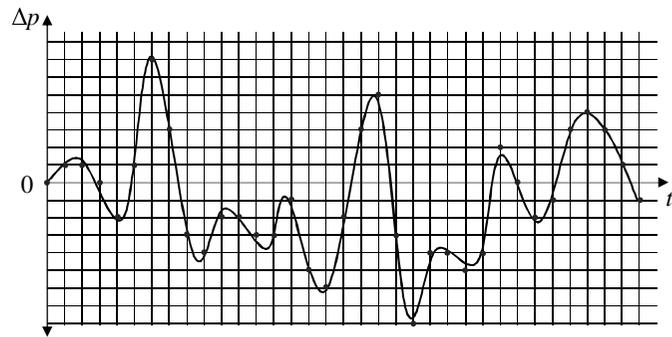


CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UNIVERSITÉ PARIS  
UNIVERSITAS

## Numérisation - échantillonnage



Laboratoire d'Informatique de Paris 6

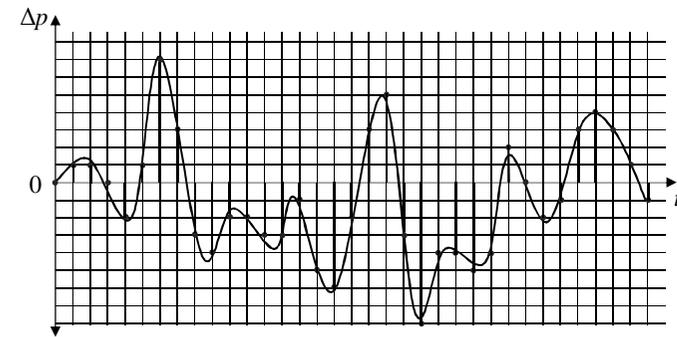


CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UNIVERSITÉ PARIS  
UNIVERSITAS

## Numérisation - quantification



Laboratoire d'Informatique de Paris 6

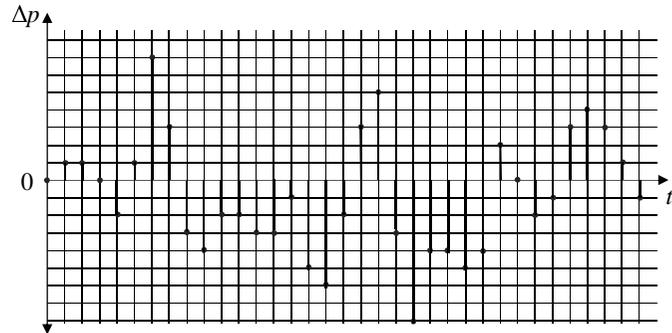


CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UNIVERSITÉ PARIS  
UNIVERSITAS

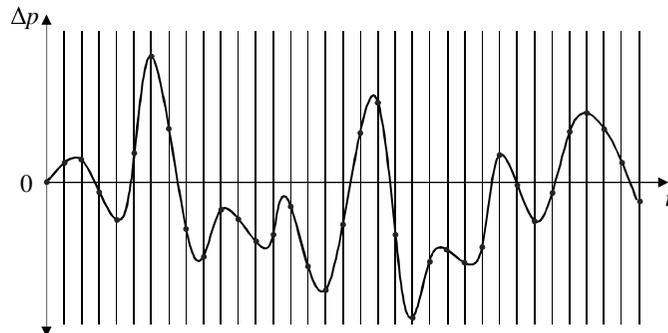
## Numérisation



## Deux étapes

- Echantillonnage
  - sampling en anglais
  - discrétisation dans le temps
- Quantification
  - quantization en anglais
  - discrétisation dans l'amplitude

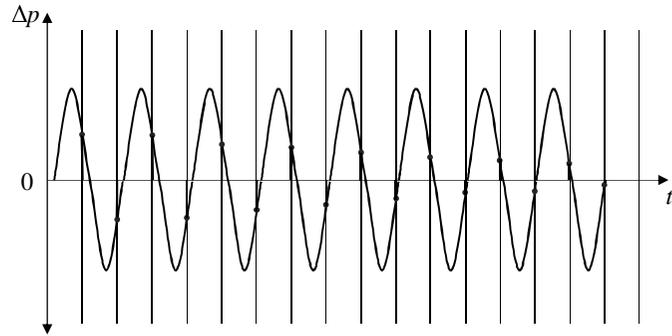
## Echantillonnage



## Th. de Nyquist-Shannon

- Pour pouvoir reconstruire un signal analogique à partir des échantillons numérisés, la fréquence d'échantillonnage doit être au moins deux fois la bande passante du signal original.
- Formulé par Harry Nyquist en 1928
- Prouvé par Claude E. Shannon en 1949

## Fréquence d'échantillonnage



Laboratoire d'Informatique de Paris 6

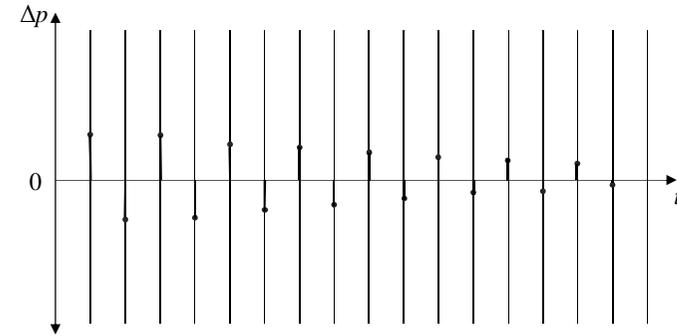


CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UNIVERSITÉ PARIS  
UNIVERSITAS

## Fréquence d'échantillonnage



Laboratoire d'Informatique de Paris 6

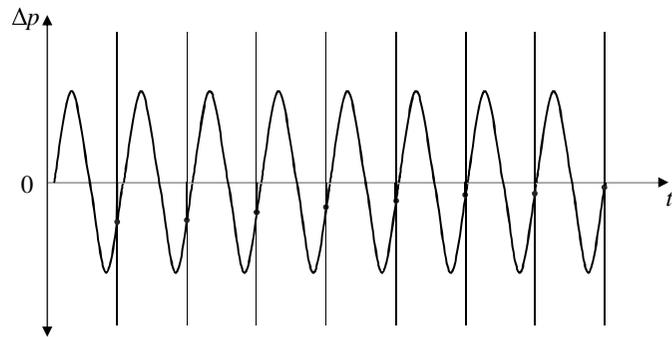


CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UNIVERSITÉ PARIS  
UNIVERSITAS

## Fréquence d'échantillonnage



Laboratoire d'Informatique de Paris 6

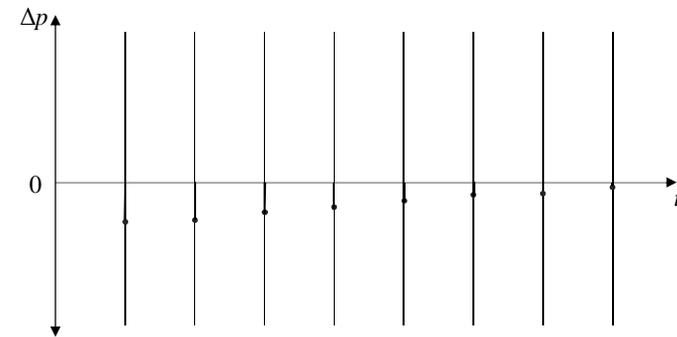


CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UNIVERSITÉ PARIS  
UNIVERSITAS

## Fréquence d'échantillonnage



Laboratoire d'Informatique de Paris 6



CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UNIVERSITÉ PARIS  
UNIVERSITAS

## Fréquences audibles

- Gamme de 20 Hz à 20 kHz (environ) :
  - Dépend des individus et de l'âge
- Large de bande :
  - un peu moins de 20 kHz
- Fréquence d'échantillonnage nécessaire :
  - 40 kHz (1 échantillon chaque 25 µsec)
  - CD audio : 44,1 kHz



Laboratoire d'Informatique de Paris 6



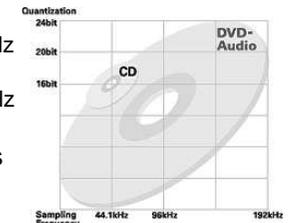
CNRS



UPMC  
UNIVERSITÉ PARIS UNIVERSITÉS

## L'audio CD et les audiophiles

- L'audio CD : d'une qualité suffisante ?
  - fréquence nécessaire : 40,0 kHz
  - marge d'erreur : ???
  - fréquence CD : 44,1 kHz
- L'audio DVD : fréquences plus élevées
  - 48, 96, ou ... 192,0 kHz
  - Meilleure qualité :
    - Hautes fréquences (notes claires)



Laboratoire d'Informatique de Paris 6



CNRS



UPMC  
UNIVERSITÉ PARIS UNIVERSITÉS

## Fréquences de la voix

- gamme de 500 Hz à 2 kHz (environ)
- canal audio CCITT 300 Hz à 3,4 kHz
- bande passante : 3,1 kHz
- fréquence d'échantillonnage nécessaire : 6,2 kHz
- qualité téléphonique : 8 kHz (G.711, GSM)
  - (un échantillon chaque 125 µsec)



Laboratoire d'Informatique de Paris 6



CNRS



UPMC  
UNIVERSITÉ PARIS UNIVERSITÉS

## Plan

- Introduction
- Numérisation
- Etendue dynamique
- Compression et codage



Laboratoire d'Informatique de Paris 6

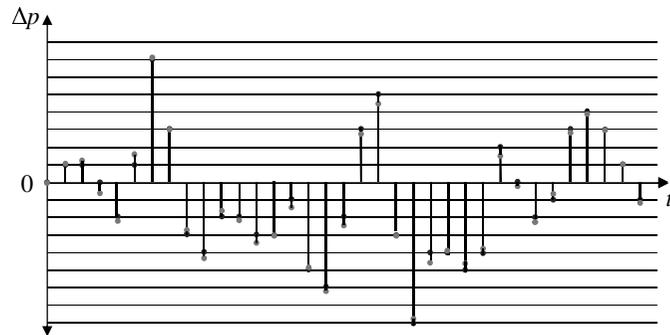


CNRS



UPMC  
UNIVERSITÉ PARIS UNIVERSITÉS

## Quantification



## Etendue dynamique

- le rapport entre deux valeurs :
  - la différence entre le plus haut et le plus bas niveau possible
  - la valeur minimale décelable du signal
- pour une quantification sur m bits :
  - Valeurs entières : différence + 1
  - $2^m : 1$
- typiquement exprimé en décibels (dB)

## Décibels

- Motivations : focaliser sur ...
  - l'intensité du signal au lieu de l'amplitude
  - les rapports au lieu des valeurs absolues
- Audio :
  - l'être humain est sensible aux dédoublement (triplement, etc.), réduction en deux (trois, etc.) de l'intensité du son

## Etendue dynamique humaine

- Pression minimale décelable : 20  $\mu$ Pa
  - la norme pour l'audio : dB re 20  $\mu$ Pa
  - dB SPL (« sound pressure level »)
- Seuil de sensation : 200 Pa
  - au dessus, plus supportable
- Rapport de pression de 10 millions toléré par l'oreille humaine
- Puissance = carré de la pression => rapport de  $10^{14}$ 
  - On utilise une échelle logarithmique
  - $20 \log_{10}(p_1/p_0) = 140$  dB

## Equivalences

rapport intensité	décibels
$x : 1$	$10 \log_{10} x$ dB
1,26	1,00 dB
1,58	1,99 dB
2,00	3,01 dB
4,00	6,02 dB
10,00	10,00 dB
20,00	13,00 dB
100,00	20,00 dB

## Valeurs typiques

- 0 dB : limite audible
- 20 à 30 dB : conversation à voix basse
- 40-50 dB : bibliothèque
- 60-70 dB : télévision
- 70-80 dB : aspirateur
- 90-100 dB : circulation dense
- 110-120 dB : discothèque, concert
- 120 dB : seuil de la douleur
- 120-130 dB : avion au décollage (à 300m)
- 180 dB : fusée Ariane au décollage
  
- Sensibilité variable selon la fréquence :
  - maximum entre 2 et 4kHz

## Etendue dynamique CD

- 16 bits/échantillon
  - 0000 0000 0000 0000 = - 32 768
  - 1000 0000 0000 0000 = 0
  - 1111 1111 1111 1111 = + 32 767
- $p1/p0 = (32\ 767 + 32\ 768 + 1) / 1 = 65\ 536$
- $20 \log_{10}(p1/p0) = 96,33$  dB
  
- environ 96 dB

## Etendue dynamique CD

- L'audio CD : d'une qualité suffisante ?
  - étendue dynamique humaine : 140 dB
  - étendue dynamique CD : 96 dB
  - étendue dynamique d'un orchestre : 90 dB
  
- L'audio DVD : encore plus étendue
  - jusqu'à 24 bits (12, 16, 20, ou 24)
  - étendue dynamique : 144 dB

## Plan

- Introduction
- Numérisation
- Etendue dynamique
- Compression et codage

## Compression avec pertes

- Réduire le débit, sans trop dégrader la qualité sonore
- Utilisation de processeurs de signaux (DSP)
- Compromis entre plusieurs caractéristiques :
  - débit
  - qualité
    - exprimée en MOS (Mean Opinion Score) de 1 à 5
  - retard
    - introduit par la compression (peut varier de 0,125 à 30 ms)
  - complexité de l'algorithme
    - exprimée en MIPS requis pour le DSP

## Plan

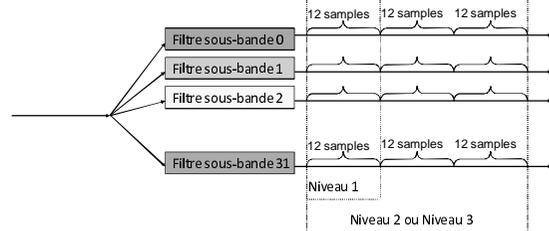
- Introduction
- Numérisation
- Etendue dynamique
- Compression et codage
  - Numérisation du son
  - Numérisation de la voix

## Audio MPEG

- Débit
  - 1,5 Mbit/s pour vidéo et audio
  - environ 0,4 Mbit/s pour deux canaux audio)
- Échantillonnage à 32, 44.1 ou 48 kHz
- Précision équivalant à une quantification à 16 bits par échantillon
- 3 niveaux de compression possibles (par rapport à  $48 \times 16 = 768$  kbit/s), de complexité croissante :
  - niveau 1 : 192 kbit/s (compression 4:1)
  - niveau 2 : 128 kbit/s (compression 6:1)
  - niveau 3 : 64 kbit/s (compression 12:1)

## Traitement par sous-bande

- spectre divisé en 32 sous-bandes :

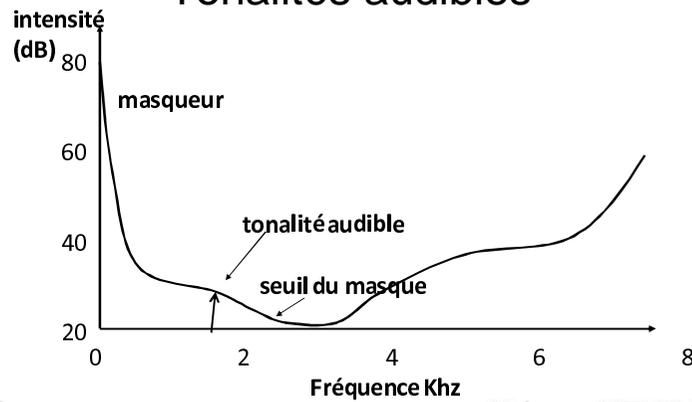


sous-bandes divisées en fenêtres

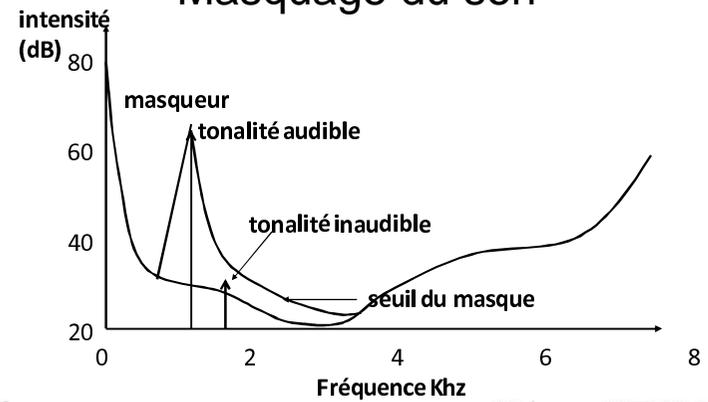
## Le psychoacoustique

- Prise en compte des connaissances sur l'ouïe :
  - On transforme en domaine fréquentiel
  - On supprime ce qu'on ne peut pas entendre

## Tonalités audibles



## Masquage du son



## Plan

- Introduction
- Numérisation
- Etendue dynamique
- Compression et codage
  - Numérisation du son
  - Numérisation de la voix



Laboratoire d'Informatique de Paris 6



CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UPMC  
UNIVERSITÉ PARIS UNIVERSITAS

## La prévisibilité de la voix

- Trois types de vocalisation
  - sonorisée
    - par exemple les voyelles
    - très périodique
  - fricative
    - par exemple s, f, et v
    - corrélations à court terme
  - occlusive
    - par exemple p, t, et k
    - des bruits très courts



Laboratoire d'Informatique de Paris 6



CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UPMC  
UNIVERSITÉ PARIS UNIVERSITAS

## Prévisibilité = compressibilité

- Les vocodeurs
  - modélisent la voix
    - est-ce sonorisée, fricative, occlusive ?
    - quel niveau ?
    - un filtre à appliquer
- Les codecs hybrides
  - envoi de l'information vocoder
  - accompagné par un signal dit de l'excitation
    - à partir d'une liste de « vecteurs »



Laboratoire d'Informatique de Paris 6



CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UPMC  
UNIVERSITÉ PARIS UNIVERSITAS

## Exemple G.728

- Echantillonnage à 8000 Hz
- 5 échantillons (délai de codage < 1ms)
  - quel vecteur ressemble le plus ? (1024 poss.)
  - quel coefficients du filtre ?
  - 10 bits de codage
  - débit : 16 Kbps



Laboratoire d'Informatique de Paris 6



CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UPMC  
UNIVERSITÉ PARIS UNIVERSITAS

## La qualité de la voix

- MOS = « Mean Opinion Score »
  - 5 Excellent
  - 4 Bon
  - 3 Moyen assez bon
  - 2 Médiocre
  - 1 Mauvais
- Basé sur des sondages (peut varier)



Laboratoire d'Informatique de Paris 6



CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UPMC  
UNIVERSITÉ PARIS UNIVERSITAS

## Des codages

- Exemples de codages de la voix :
  - 1016 standard DoD à 4.8 kbps (MOS 3.5)
  - GSM de l'ITU à 13 kbps (MOS 3.6)
  - G 711 de l'ITU à 64 kbps (MOS 4.2)
- Pour la voix sur IP
  - G 723.1 de l'ITU à 5.3 kbps (MOS 3.7)



Laboratoire d'Informatique de Paris 6



CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UPMC  
UNIVERSITÉ PARIS UNIVERSITAS

## Conclusion

- Compression audio :
  - Prise en compte de connaissances sur le fonctionnement de la voix et de l'ouïe pour affiner le codage.
- Différences entre codage du son (large bande de fréquences) et de la voix (bande beaucoup plus faible)



Laboratoire d'Informatique de Paris 6



CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UPMC  
UNIVERSITÉ PARIS UNIVERSITAS

## Compression images/vidéos



Laboratoire d'Informatique de Paris 6



CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UPMC  
UNIVERSITÉ PARIS UNIVERSITAS

## Principe général

- Comme pour la voix, on n'encode pas tout
  - quelles longueurs d'onde ?
  - à quelle finesse ?
- Dépend de l'œil humain



Laboratoire d'Informatique de Paris 6



CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UNIVERSITÉ PARIS  
UNIVERSITAS

## L'œil humain

- Deux éléments
  - les bâtonnets (« rods ») : environ 120 M
    - vision nocturne
    - perception de formes et mouvement
  - les cônes : environ 6,5 M
    - vision diurne
    - perception des couleurs
    - 3 types
      - « bleu » (sensible aux ondes courtes)
      - « vert » (sensible aux ondes moyens)
      - « rouge » (sensible aux ondes longues)
- en cas de manque : « daltonisme »



Laboratoire d'Informatique de Paris 6



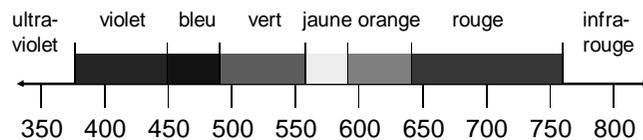
CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UNIVERSITÉ PARIS  
UNIVERSITAS

## Longueurs d'onde

- Entre 380 nm et 760 nm



Laboratoire d'Informatique de Paris 6

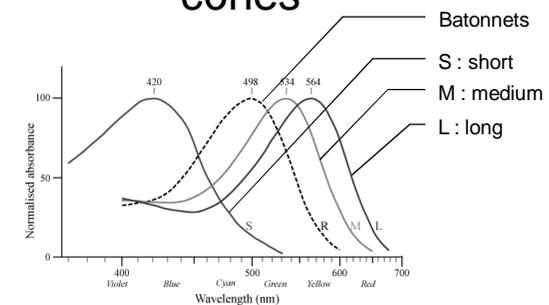


CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UNIVERSITÉ PARIS  
UNIVERSITAS

## Stimulation des bâtonnets et cônes



source : [http://en.wikipedia.org/wiki/Cone\\_cell](http://en.wikipedia.org/wiki/Cone_cell)

d'après Bowmaker J.K. and Dartnall H.J.A., "Visual pigments of rods and cones in a human retina." *J. Physiol.* **298**: pp 501-511 (1980).



Laboratoire d'Informatique de Paris 6



CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UNIVERSITÉ PARIS  
UNIVERSITAS

## Eléments de la couleur

- Luminance (intensité)
  - Energie rayonnant d'une source jusqu'à l'œil
  - (clarté pour la réflexion d'un objet)
  - Claire ou sombre
- Teinte
  - Longueur d'onde dominante
  - Type de couleur (bleu, jaune, rouge)
- Saturation
  - Fonction inverse de la largeur de la bande du spectre émis ou réfléchi
  - Pâle ou vive

## Sensibilité de l'oeil

- Luminance
  - Sensibilité élevée
- Chrominance (variations de couleur)
  - Sensibilité faible
  - 350 000 couleurs différentes
  - 128 teintes
    - sensibilité maximale pour la jaune
  - 20 saturations
    - sensibilité maximale pour la rouge violacé

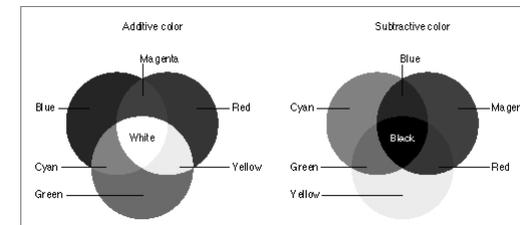
## Couleurs pures et la vision

- Nous n'apercevons pas que des couleurs pures
- Des mélanges de fréquences atteignent nos yeux
- Ces mélanges sont aperçus comme des couleurs en eux mêmes
- La correspondance entre mélange et perception est très complexe
  - Elle admet plusieurs modèles explicatifs

## Modèles RGB et CMY

RGB (RVB) : additif  
(pour les moniteurs)

CMY/CMYK : soustractif  
(pour les imprimantes)



source : <http://developer.apple.com/documentation/mac/ACI/ACI-48.html>

## Synthèse additive : exemple

l'image couleur :



luminance rouge



luminance verte



luminance bleue



Laboratoire d'Informatique de Paris 6



CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UPMC  
UNIVERSITÉ PARIS  
MONTPARNASSE

## Synthèse soustractive

- Absorption pondérée de plusieurs composantes primaires
- Trois types de filtres d'une lumière blanche
  - Cyan, Magenta, Jaune
- Couleurs complémentaires de RVB
  - (0,0,0) Blanc
  - (1,1,0) Bleu
  - (1,0,1) Vert
  - (0,1,1) Rouge
  - (1,1,1) Noir



Laboratoire d'Informatique de Paris 6



CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UPMC  
UNIVERSITÉ PARIS  
MONTPARNASSE

## Nombreux modèles

- Triangle de Maxwell (RGB)
  - Couleurs non représentables
- Autres modèles
  - Différents moyens de représenter les couleurs
  - CIE Yxy
  - XYZ : excitation d'un type de bâtonnet
  - Luv et Lab : perception linéaire des couleurs
  - LCH : contrôle des moniteurs



Laboratoire d'Informatique de Paris 6



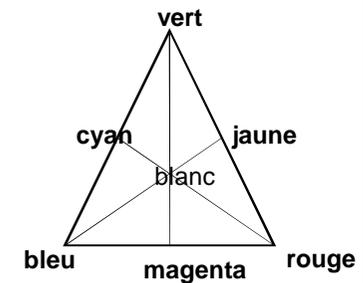
CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UPMC  
UNIVERSITÉ PARIS  
MONTPARNASSE

## Modèle : triangle de Maxwell

- Essentiellement le modèle RGB
- Chaque couleur est composée de bleu, vert, et rouge
- Il existe des couleurs qu'on ne peut pas produire ainsi



Laboratoire d'Informatique de Paris 6



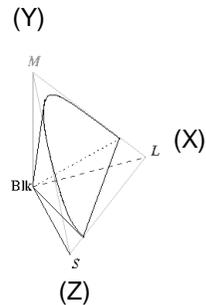
CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UPMC  
UNIVERSITÉ PARIS  
MONTPARNASSE

## Modèle CIE XYZ (tristimulus)

- Coordonnées :
  - X, Y, et Z
  - les nouvelles 'couleurs' primaires (remplacent R, G, B)
- Chacune correspond au niveau d'excitation d'un des types de bâtonnet
- Produit toutes les couleurs possibles



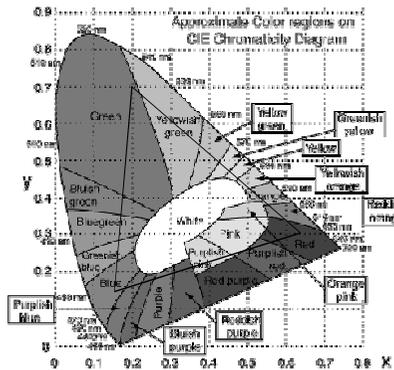
## Traduction entre RVB et XYZ

- X, Y, et Z sont chacun une somme pondérée des composantes primaires

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} 2.77 & 1.75 & 1.13 \\ 1.00 & 4.59 & 0.06 \\ 0.00 & 0.06 & 5.60 \end{bmatrix} \begin{bmatrix} R \\ V \\ B \end{bmatrix}$$

## Modèle CIE Yxy

- Commission Internationale de l'Eclairage (norme de 1931)
- Coordonnés
  - x et y la chromaticité
  - Y (pas montré) la luminance
- Comprend le triangle de Maxwell
- x et y produisent toutes les variantes de couleur possible



## Traduction entre XYZ et Yxy

- Y : la luminance
  - $Y = Y$
- x et y : la chromaticité
  - $x = X / (X+Y+Z)$
  - $y = Y / (X+Y+Z)$

## Modèles CIE Luv et Lab

- Définis par la CIE en 1976
  - le modèle Yxy ne correspond pas linéairement à notre perception de couleurs
- luminance :  $L^*$
- chrominance
  - $u^*$ ,  $v^*$
  - $a^*$  = rouge/vert,  $b^*$  = jaune/bleu



Laboratoire d'Informatique de Paris 6



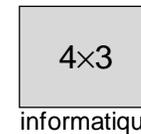
CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UPMC  
UNIVERSITÉ PARIS UNIVERSITÉS

## Image Matricielle

- Composition en termes de « pixels »
  - picture elements
- Rapport de côtés (aspect ratio)



Laboratoire d'Informatique de Paris 6



CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UPMC  
UNIVERSITÉ PARIS UNIVERSITÉS

## Définition d'une image

- Basse définition
  - 76k 320×240
  - 300k 640×480
  - 480k 800×600
- Moyenne définition
  - 780k 1024×768
  - 997k 1152×854
  - 1.3M 1280×1024
- Haute définition
  - 2.1M 1600×1200
  - 3.1M 2048×1536
  - 4.9M 2560×1920
  - ...



Laboratoire d'Informatique de Paris 6



CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UPMC  
UNIVERSITÉ PARIS UNIVERSITÉS

## Résolution d'une image

- Nombre de pixels par unité de surface
- Périphérique de restitution
- Ecran
  - distance entre deux luminophores
  - 0.15 mm à 0.7 mm
- Imprimante
  - taille du point coloré
  - 0.02 mm à 0.08 mm



Laboratoire d'Informatique de Paris 6



CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UPMC  
UNIVERSITÉ PARIS UNIVERSITÉS

## Nombre de bits par pixel

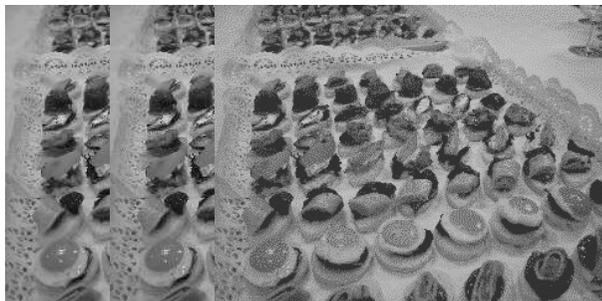
bits	couleurs	genre d'image
4	16	texte ou dessin
8	256	texte ou dessin
16	65 536	image
24	16 M	photo
32	1 G	photo

## Réduction en profondeur

- Possible de passer de plusieurs couleurs à peu de couleurs
- Définition de palettes
  - Par image ou groupe d'images
- Avantage : réduction de la taille utilisée
- Inconvénient : augmentation du temps de calcul et de restitution

## Palette : exemples

16M    256    16



## Codage réversible

- Codage + décodage : fonction identité
- Méthodes de compression classique
  - Fanno-Shannon
  - Huffman
- Taux de compression dépendant fortement de l'image

## Codage non réversible

- Codage : fonction non injective
  - même codage pour deux images différentes
- But : même codage pour deux images perceptivement semblables
- Méthodes :
  - Quantification vectorielle
  - Codage par sous-bandes
  - Transformations linéaires



Laboratoire d'Informatique de Paris 6



CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UPMC  
UNIVERSITÉ PARIS  
SUD

## Quantification vectorielle

- Découpage de l'image en blocs (carrés) de pixels
  - Recherche du plus proche bloc dans la table des blocs
  - Codage du bloc par son numéro de bloc
- Calcul de la table des blocs
  - Corpus d'images représentatives (?)
  - Recherche des classes de blocs et des représentants de classe
    - Algorithmes (nuées dynamiques, LBG)
- Facteur de compression important
  - Répétitions de motif
  - Extension multi résolution (méthodes fractales)



Laboratoire d'Informatique de Paris 6

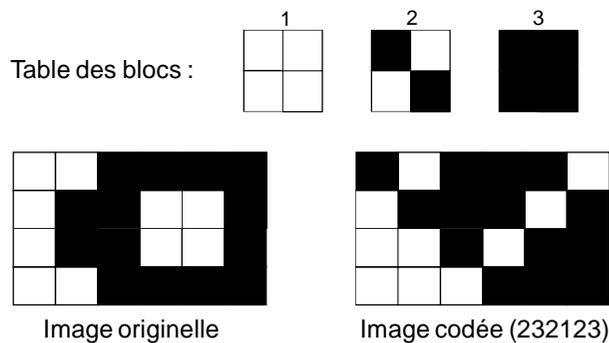


CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UPMC  
UNIVERSITÉ PARIS  
SUD

## Exemple



Laboratoire d'Informatique de Paris 6



CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UPMC  
UNIVERSITÉ PARIS  
SUD

## Caractéristiques des formats

- Entête
  - Codage utilisé
  - Définition et résolution de l'image
  - Représentation de la luminance et des couleurs (palette)
- Méta données
  - Informations décrivant le contenu de l'image, les conditions de son acquisition et les différents traitements qu'elle a subit
- Type de compression
  - Réversible ou non réversible (taux de compression)



Laboratoire d'Informatique de Paris 6



CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UPMC  
UNIVERSITÉ PARIS  
SUD

## Principaux formats

	Méta données	Compression	Multi résolution
BMP	-	-	-
TIFF	dans l'entête	réversible	-
GIF, PNG	-	réversible	-
JFIF	-	non réversible	-
SPIFF	séparé de l'entête	non réversible	-
JTIP	-	non réversible	oui
Flashpix	séparé de l'entête	non réversible	oui
PhotoCD	-	-	oui

## Compression GIF et PNG

- GIF utilise une compression LZW (propriétaire)
  - Palette de 2 à 256 couleurs
  - Utilisation de la transparence
  - Plusieurs images peuvent être stockées dans le même fichier (gif animées)
- PNG utilise une combinaison de Huffman et de LZ77
  - MNG étend PNG pour mettre plusieurs images

## Traitement d'images

- Transformations ponctuelles
  - Pour chaque pixel  $px(a,b)$  d'une image
    - nouvelle valeur du  $px = F(\text{valeur}(a,b))$
    - $F$  quelconque (linéaire, non linéaire)
      - identique pour tous les pixels
- Filtrage spatial
  - Pour chaque pixel  $px(a,b)$  d'une image
    - nouvelle valeur du  $px = G(\text{valeurs pixels au voisinage du } px)$
    - $G$  quelconque (linéaire, non linéaire)
      - identique pour tous les pixels

## Transformations ponctuelles

Exemple : contraste  
écart-type de la luminance des pixels



## Filtrage spatial

Exemple : filtres passe-bas et passe-haut  
écart-type de la luminance des pixels



## Compression images Le format JFIF/PEG

## Pourquoi JPEG

- Compression avec des méthodes sans pertes (Huffman, LZW) pas suffisantes pour les images et la vidéo
- JPEG codage avec pertes (version de base), basé sur deux observations :
  - Le contenu est relativement stable : peu d'altérations dans une petite zone (typiquement 8x8). En mode fréquentiel : hautes fréquences correspondent à des petits détails et du bruit
  - L'œil humain est beaucoup plus sensible aux modifications des basses fréquences que des hautes fréquences.
- Codage avec perte qui vise principalement à supprimer les hautes fréquences et à prendre cette perte en compte pour compresser au maximum.

## Codage JFIF (JPEG)

- Joint Photographic Experts Group
- Choix d'un taux de compression
  - Table normalisée de quantification associée
- Découpage de l'image en blocs de 8x8 pixels
  - Chaque bloc est traité en plusieurs phases
- Performances
  - Taux de compression = 10 pour une image identique à l'original
  - Taux de compression = 100 pour une image reconnaissable

## Codage JFIF : exemples

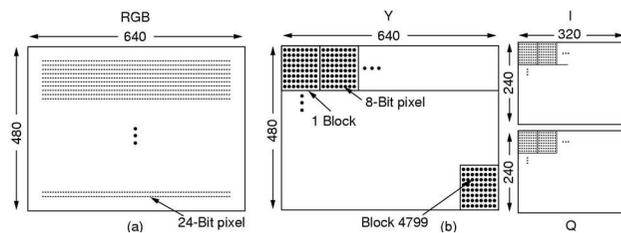
49k    14k    7k    (230k non compressé)



## JPEG - étapes

1. Préparation des blocs
2. DCT (Discrete Cosine Transform)
3. Quantification
4. Compression différentielle
5. Zig-zag et run-length encoding
6. Codage de Huffman

## 1. Préparation des blocs



## 2. DCT

- Transformation Cosinus Discrète
  - Discrete Cosine Transform
- Conversion d'un bloc 8x8 dans le domaine fréquentiel :
  - Exemple : liste de 12 entiers entre 1 et 3
    - (2, 3, 1, 2, 2, 0, 1, 1, 0, 1, 0, 0).
    - Tri + comptage des fréquences : (4,4,3,1).
    - Perte d'information mais fréquences capturées.
  - Fourier, DCT...
- Nécessité de l'étape 1 :
  - DCT nécessite une matrice carrée
  - Faire une DCT sur l'image entière serait coûteux

## DCT unidimensionnelle

- Transformation :

$$F(\omega) = \frac{\alpha(u)}{2} \sum_{n=0}^{N-1} f(n) \cos \frac{(2n+1)\omega\pi}{16}$$

$$\alpha(0) = \frac{1}{\sqrt{2}}$$

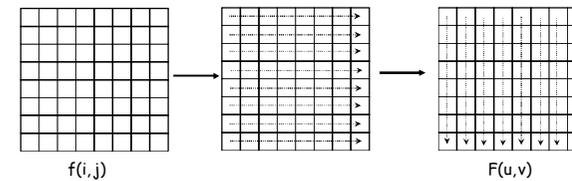
$$\alpha(p) = 1 \quad [p \neq 0]$$

- Transformation inverse

$$f'(n) = \frac{\alpha(u)}{2} \sum_{\omega=0}^{N-1} F(\omega) \cos \frac{(2n+1)\omega\pi}{16}$$

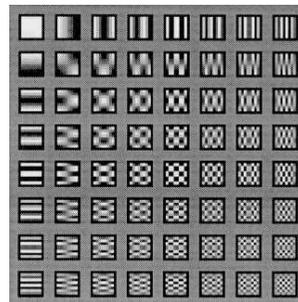
## DCT bidimensionnelle

- Deux transformations unidimensionnelles
  - $F(0,0)$  : composante DC : moyenne du bloc
  - $F(i,j)$  indiquent les puissances spectrales pour chaque fréquence (coefficients AC).



## DCT

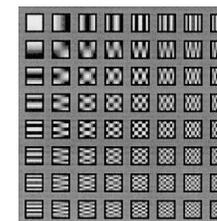
- 64 (8 x 8) fonctions de base
- Superposition de ces fonctions



## DCT

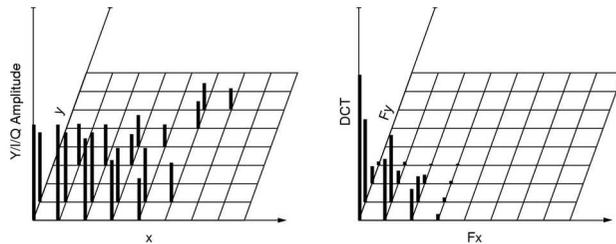


0 – noir  
255 – blanc



70	9	10	0	0	0	0	0
0	0	0	0	0	0	0	0
90	0	0	0	0	0	0	0
-89	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

## Exemple



## 3. Quantification

- Gagner de l'espace : diminuer la précision du stockage :
  - Basses fréquences plus pertinentes => précision plus élevée pour les basses fréquences.
  - Seule partie avec pertes.
- Matrice de quantification :
  - Valeurs de la DCT divisées par celle de la matrice de quantification et arrondies.
  - Coefficients de la matrice de plus en plus grands lorsqu'on s'éloigne de la position (0,0), pour filtrer les hautes fréquences (œil sensible aux basses fréquences)
  - JPEG : deux matrices pour chrominance et luminance.
- Décompression : multiplier par le contenu de la matrice.

Matrice de pixels d'entrée								Matrice DCT quantifiée							
140	144	147	140	140	155	179	175	408	-4	2	-1	2	-1	-1	-1
144	152	140	147	140	148	167	179	4	-5	3	-1	-1	1	1	0
152	155	136	167	163	162	152	172	-1	-3	0	0	-1	0	-1	0
168	145	156	160	152	155	136	160	-1	0	1	-1	0	0	0	0
162	148	156	148	140	136	147	162	0	1	1	0	-1	1	1	1
147	167	140	155	155	140	136	162	0	0	-1	0	0	0	0	0
136	156	123	167	162	144	140	147	1	0	0	0	0	0	0	0
148	155	136	155	152	147	147	136	0	0	0	0	0	0	0	0
Matrice DCT								Matrice DCT déquantifiée (décompression)							
1210	-18	15	-9	23	-9	-14	-19	1209	-20	14	-9	22	-13	-15	-17
21	-34	26	-9	-11	11	14	7	20	-35	27	-11	-13	15	17	0
-10	-24	-2	6	-18	3	-20	-1	-7	-27	0	0	-15	0	-19	0
-8	-5	14	-15	-8	-3	-3	8	-9	0	13	-15	0	0	0	0
-3	10	8	1	-11	18	18	15	0	13	15	0	-19	21	23	25
4	-2	-18	8	8	-4	1	-7	0	0	-17	0	0	0	0	0
9	1	-3	4	-1	-7	-1	-2	15	0	0	0	0	0	0	0
0	-8	-2	2	1	4	-6	0	0	0	0	0	0	0	0	0
Matrice de quantification								Matrice de pixels de sortie (décompression)							
3	5	7	9	11	13	15	17	142	143	154	141	133	153	179	179
5	7	9	11	13	15	17	19	139	152	129	151	144	154	163	181
7	9	11	13	15	17	19	21	150	156	139	166	162	163	154	172
9	11	13	15	17	19	21	23	163	145	160	153	151	153	145	154
11	13	15	17	19	21	23	25	168	150	156	145	140	139	141	159
13	15	17	19	21	23	25	27	148	164	133	164	158	140	136	163
15	17	19	21	23	25	27	29	130	159	123	164	165	140	134	145
17	19	21	23	25	27	29	31	148	156	140	148	159	146	153	141

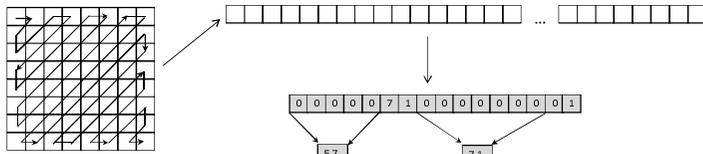
## 4. Compression différentielle

DPCM : Differential Pulse Code Modulation

- Seul le premier bloc est codé.
- Les autres sont codés en différence par rapport à ce premier bloc, en espérant :
  - Que les valeurs soient faibles (moins de bits pour coder).
  - Soient proches de 0 et compressés par les étapes suivantes.
  - Deux étapes pour y arriver.

## 5. Zig-Zag et RLE

- Zig-zag regroupe les hauts coefficients en bout de vecteur
- RLE (Run-length encoding) :
  - Les hautes fréquences sont en principe presque toutes nulles
    - Longues séquences de 0
  - On remplace le vecteur par une suite (skip,value) :
    - Skip = nombre de 0 consécutifs
    - Value= valeur suivant le dernier 0 (nombre de bits pour coder + valeur)



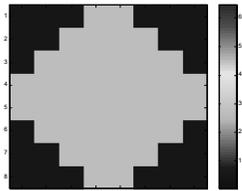
## 6. Codage de Huffman

- Codage standard

## Exemple

D'après [www.gpa.etsmtl.ca/cours/gpa787/Acetates/COURS11GG\\_A2007.ppt](http://www.gpa.etsmtl.ca/cours/gpa787/Acetates/COURS11GG_A2007.ppt)

- Sans compression:  $8 \times 8 \times 8 \text{ bits} = 512 \text{ bits}$
- En plus, si 3 couleurs (RGB)
  - donc 1536 bits.



## Exercice

200	0	-446	0	0	0	-31.7	0	6	9	12	15	18	21	24	27
0	0	0	0	0	0	0	0	9	12	15	18	21	24	27	30
-446	0	-200	0	185	0	0	0	12	15	18	21	24	27	30	33
0	0	0	0	0	0	0	0	15	18	21	24	27	30	33	36
0	0	185	0	200	0	-76.5	0	18	21	24	27	30	33	36	39
0	0	0	0	0	0	0	0	21	24	27	30	33	36	39	42
-31.7	0	0	0	-76.5	0	-200	0	24	27	30	33	36	39	42	45
0	0	0	0	0	0	0	0	27	30	33	36	39	42	45	48

DCT

33	0	-37	0	0	0	-1	0
0	0	0	0	0	0	0	0
-37	0	-11	0	8	0	0	0
0	0	0	0	0	0	0	0
0	0	8	0	7	0	-2	0
0	0	0	0	0	0	0	0
-1	0	0	0	-2	0	-5	0
0	0	0	0	0	0	0	0

Matrice de quantification

Résultat

- Il faut donc coder :
  - (6)(33), (2,6)(-37), (1,6)(-37), (6,4)(-11), ...
- Quelle compression obtient-on ?

## Autres modes JPEG

- (Mode séquentiel)
- Mode sans perte :
  - DCT + quantification est remplacée
- Mode progressif
  - Première passe : AC + quelques DCT
  - Passes suivantes : DCT suivants
- Motion JPEG :
  - Application à chaque image d'une vidéo



Laboratoire d'Informatique de Paris 6



CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UPMC  
UNIVERSITÉ PARIS  
UNIVERSITAS

## Compression Vidéo



Laboratoire d'Informatique de Paris 6



CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UPMC  
UNIVERSITÉ PARIS  
UNIVERSITAS

## Formats analogiques

- NTCS (Standard US) :
  - Porteuse 4.429 MHz ; 30 images/s
  - Entrelacement : 525 lignes (60 Hz)
  - Mauvaise qualité en cas de perturbation
- SECAM :
  - Chrominance envoyée une ligne sur deux
  - Perte de qualité mais non visible à l'œil
  - Zones rouges et bleues mal rendues
  - Disparition en 2010 en France
- PAL :
  - Porteuse 4,43 MHz ; 25 images/s
  - Avantages de SECAM mais plus proche de NTSC
- Remplacement à terme par la TNT.



Laboratoire d'Informatique de Paris 6



CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UPMC  
UNIVERSITÉ PARIS  
UNIVERSITAS

## Formats numériques

- Haute définition
  - Deux fois plus de lignes et colonnes que les systèmes conventionnels
  - Frame rate: 50 or 60 frames per second
- Ratio : 16:9 ou 4:3
- Mode entrelacé ou progressif :
  - Systèmes conventionnels (NTSC, PAL, SECAM) sont entrelacés
  - HDTV, moniteurs sont progressifs



Laboratoire d'Informatique de Paris 6



CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UPMC  
UNIVERSITÉ PARIS  
UNIVERSITAS

D'après wikipedia				
Pixel Resolution (W x H)	Video Format	Pixels (Adapted Megapixels)	Image	Description
1024 x 768	HD Ready, XGA	786,432 (0,8)	16:9	Typically a computer resolution; also exists as a non-standardized "HD-Ready" TV size and on PDP HDTV displays with non-square pixels.
1248 x 702	720p Clean Aperture	876,096 (0,9)	16:9	Used for 750-line video with raster artifact/overscan compensation, as defined in SMPTE 296M.
1280 x 720	720p, WXGA	921,600 (0,9)	16:9	Used for 750-line video, as defined in SMPTE 296M, ATSC A/53, ITU-R BT.1548, Digital television, DLP, LCD and LCOS projection HDTV displays.
1366 x 768	720p/1080i, WXGA, HD Ready	1,049,088 (1,0)	683:384 (Approx 1.69)	Typically a TV resolution; also exists as a standardized "HD-Ready" TV size, HDTV common pixel resolution, that used on LCD/PDP HDTV displays.
1024 x 1080	HD Ready 1080p	1,105,920 (1,1)	128:136 (Approx 4:3)	Non-standardized "HD-Ready" TV size, Used on PDP HDTV displays.
1280 x 1080	HD Ready 1080p	1,382,400 (1,4)	32:27 (Approx 1.69)	Non-standardized "HD-Ready" TV size, Used on PDP HDTV displays.
1440 x 1080	HDCAM/HDV 1080i	1,555,200 (1,6)	4:3	Used for anamorphic 1152-line video in the HDCAM and HDV formats introduced by Sony and defined (also as a luminance subsampling matrix) in SMPTE D11.
1388 x 1062	1080p Clean Aperture	2,001,280 (2,0)	16:9	Used for 1152-line video with raster artifact/overscan compensation, as defined in SMPTE 274M.
1920 x 1080	1080p, Full HD, HD Ready 1080i/p	2,073,600 (2,1)	16:9	Used for 1152-line video, as defined in SMPTE 274M, ATSC A/53, ITU-R BT.709, HDTV common pixel resolution that used on all types of HDTV technologies.
3840 x 2160	2160p	8,294,400 (8,3)	16:9	Quad HDTV, (there is no HD Ready 2160p Quad HDTV format)

## Bande passante

- Exemple :
  - 720,000 pixels par image (frame),
  - Quantification de 8 bits par pixel
  - 60 images seconde
  - Taux de 43,2 Mo par seconde
  - Baisse typique à 4,24 Mo avec une compression MPEG

## Bande passante

- Pour une image :
  - Sans compression : résolution x taille pixel
  - Avec compression : dépend de l'image et de la méthode de compression
  - Images vectorielles : dépend des primitives
- Pour une vidéo :
  - Sans compression : taille image x images/seconde
  - Avec compression : dépend de la méthode de compression et du contenu (typiquement des changements de scènes)

## MPEG

- MPEG-1 :
  - 352x240 pixels à 30 images par seconde en NTSC
  - 352x288 pixels à 25 images par seconde en PAL/SECAM
  - Débits de l'ordre de 1.2 Mbit/s
  - Mode progressif uniquement
- MPEG-2 :
  - Débits de l'ordre de 4 à 6 Mbit/s
  - Redondance temporelle, spatiale et subjective :
    - Deux frames successives sont similaires en général
  - Trois types de frames :
    - I (Intracoded) frames - images codées en JPEG
    - P (Predictive) frames - différence avec la frame précédente
    - B (Bidirectional) frames - différence avec les frames précédentes et suivantes

## MPEG (suite)

- Pas uniquement la vidéo :
  - Partie 1 : système : synchronisation entre vidéo et son
  - Partie 2 : Vidéo
  - Partie 3 : Audio
    - MP1 ou MPEG-1 Part 3 Layer 1
    - MP2 ou MPEG-1 Part 3 Layer 2
    - MP3 ou MPEG-1 Part 3 Layer 3
  - Partie 4 : conformité (méthodes, jeux de test)
  - Partie 5 : simulation (implémentation C)



Laboratoire d'Informatique de Paris 6

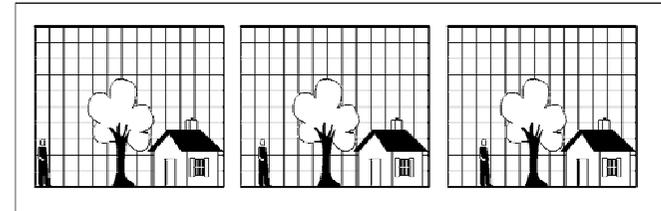


Centre National de la Recherche Scientifique



UPMC Paris Universit s

## MPEG



Frames consecutives



Laboratoire d'Informatique de Paris 6

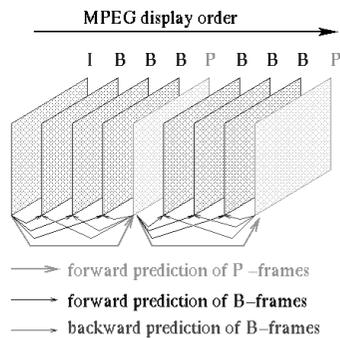


Centre National de la Recherche Scientifique



UPMC Paris Universit s

## Frames MPEG



- I-Frame
  - Frames ind pendantes (avance et retour rapide)
- P-Frame
  - Pr dites   partir des I ou P-frames pr c dentes
- B-Frame
  - Pr dites   partir des I et P-frames pr c dentes et suivantes



Laboratoire d'Informatique de Paris 6



Centre National de la Recherche Scientifique



UPMC Paris Universit s

## Frames MPEG

- Entra ne des pertes
- Codage tr s lent, d codage rapide :
  - Pas utilisable pour de la visioconf rence (plut t H263)
- Frames envoy es dans un ordre diff rent :
  - Affichage : IPPBBBBPIPPBBBBPI
  - Envoi : IPPPBBBBIPPPBBBBPI
  - Utilisation de buffers



Laboratoire d'Informatique de Paris 6



Centre National de la Recherche Scientifique



UPMC Paris Universit s

## I Frames

- Utilisation de JPEG pour le codage :
  - Blocs de 8x8
  - DCT sur les blocs
  - Quantification avec une valeur constante pour tous les coefficients (pas de matrice de quantification)

## MPEG-xxx

- MPEG-3 abandonné rapidement :
  - MPEG-2 pouvait faire la même chose sans beaucoup de modifications
- MPEG-4 :
  - Débuté en 1998, draft en 2003 (MPEG-3 abandonné)
  - Compression vidéo : H-264, divx, ...
  - Codage par objets :
    - Extraction depuis les scènes
    - Codage spécifique pour chaque objet
  - Amélioration de toutes les techniques de MPEG-2 (meilleurs encodeurs notamment)
- MPEG-7 - multimédia :
  - Recherche d'informations
  - Combinaison d'objets pour former une scène
  - Indépendance entre contenu et présentation

## Conclusion

- Format JFIF très utilisé :
  - Bon taux de compression
  - Manque transparence/animations
  - Utilisé dans MPEG
- Formats vidéo :
  - Complexe à mettre en œuvre
  - Compression élevée.

## Multimédia : système et programmation

### Partie 2

## Serveur multimedia

Un exemple typique détaillé : serveur VoD

- Serveurs fonctionnent comme un magnétoscope (lecteur DVD)
- Système VoD ou near-VoD :
  - VoD : lecture à tout moment
  - Near-VoD : films programmés à horaires fixes
- Terminologie :
  - Serveurs avec streaming : push servers
  - Serveurs de fichiers classique : pull servers



Laboratoire d'Informatique de Paris 6



CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UNIVERSITÉ PARIS  
SUD

## OS serveur multimedia

Fortes contraintes temps-réel

- Ordonnement de processus :
  - A qui envoyer les morceaux de vidéos et quand ?
  - Rate Monotonic Scheduling, Earliest Deadline First
- Système de fichiers
  - Comment stocker les fichiers ?
  - Contraintes temps-réel, avance/retour rapide, utilisateurs multiples
  - Placement des fichiers sur le disque, Mise en cache
- Ordonnement du disque
  - Où placer les fichiers sur le disque ?
  - Très prédictible
  - Approche statique ou dynamique



Laboratoire d'Informatique de Paris 6



CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UNIVERSITÉ PARIS  
SUD

## Ordonnement

- Ordonnement classique :
  - Équité
  - Optimisation de l'utilisation de ressources
- Ordonnement temps-réel :
  - Deadline fortes garanties sans adaptation
  - Opérations terminées trop tard inutiles
- Ordonnement multimédia :
  - Deadline souples garanties : adaptation à la charge
  - Opérations terminées trop tard : moins bonne qualité



Laboratoire d'Informatique de Paris 6



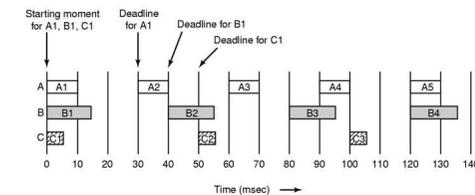
CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UNIVERSITÉ PARIS  
SUD

## Ordonnement multimedia

- Cas simples : taille des fichiers fixée, nombre de fichier fixé



- Cas plus réaliste :
  - Taille variable
  - Résolution variable
  - Nombre variable



Laboratoire d'Informatique de Paris 6



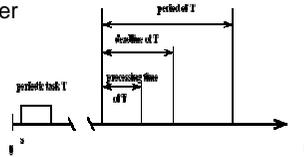
CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UNIVERSITÉ PARIS  
SUD

## Gestion quasi temps-réel

- Taches :
  - Unité de base à ordonnancer
  - Contraintes temporelles :
    - Périodiques, Deadlines
  - Ressource : CPU
  - Indépendantes
  - Préemptive ou pas
- Gestionnaire de taches
  - Contrôle d'admission (nouvel utilisateur accepté ?)
  - Détermine l'ordonnancement
  - Doit fournir des garanties



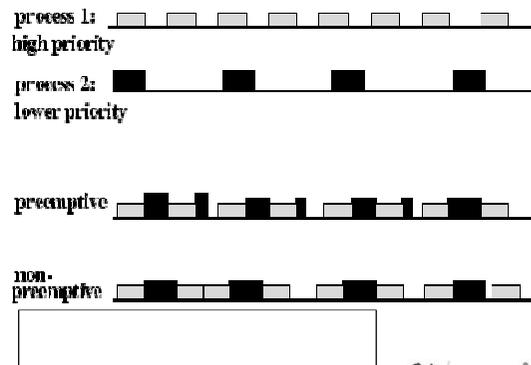
## Ordonnancement

- Si ordonnancement possible :
  - L'ordonnanceur garanti la deadline
- Sinon l'utilisateur est accepté en qualité dégradée ou refusé :
  - L'acceptation ne doit pas dégrader les utilisateurs déjà présents
- Utilisation du processeur :

$$U = \sum_i^m \frac{C_i}{P_i} \leq 1$$

- m : nombre de processus
- C<sub>i</sub> : temps CPU du processus i
- P<sub>i</sub> : période du processus i

## Préemption



## Ordonnancement monotone

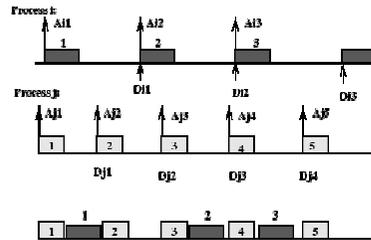
RMS : Rate Monotonic Scheduling

- Contraintes sur les processus :
  - Doivent se terminer dans leur période
  - Indépendants
  - Temps CPU de chaque requête est constant
  - Si non périodiques, pas de deadline
  - Préemption
- Priorité déterminée par le taux d'exécution
  - Exemple d'un processus lancé toutes les 40ms
    - fréquence = priorité = 25Hz
  - Hautes priorités prioritaire sur les basses.

## Ordonnement sur deadline

EDF : Earliest deadline first

- Priorité = première deadline d'abord



## Contrôle d'admission

- RMS

– Ordonnement peut être garanti si :

$$U = \sum_i^m \frac{C_i}{P_i} \leq m(2^{1/m} - 1) \rightarrow \ln(2) = 0.693...$$

- EDF

– Garanti si (utilisation < 100%) :

$$U = \sum_i^m \frac{C_i}{P_i} \leq 1$$

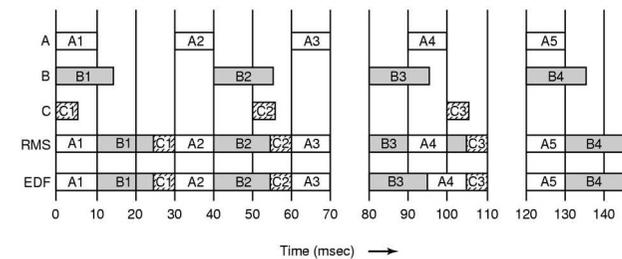
## Ordonnement : exemple

Process	Period (ms)	CPU Time (ms)	Priority	Deadline (ms)
A	30	10	33	30
B	40	15	25	40
C	50	5	20	50

- $U = 10/30 + 15/40 + 5/50 = 0.808$ 
  - ~ 80% du CPU utilisé
  - Borne pour RMS : 0.78 (pas de garantie)
  - Borne pour EDF : 1 (ok)

## Ordonnement : exemple

Process	Period (ms)	CPU Time (ms)	Priority	Deadline (ms)
A	30	10	33	30
B	40	15	25	40
C	50	5	20	50



## Ordonnement : exemple 2

Process	Period (ms)	CPU Time (ms)	Priority	Deadline (ms)
A	30	15	33	30
B	40	15	25	40
C	50	5	20	50

- $U = 15/30 + 15/40 + 5/50 = 0.975$ 
  - ~ 98% du CPU utilisé
  - Peut être ordonné avec EDF



Laboratoire d'Informatique de Paris 6



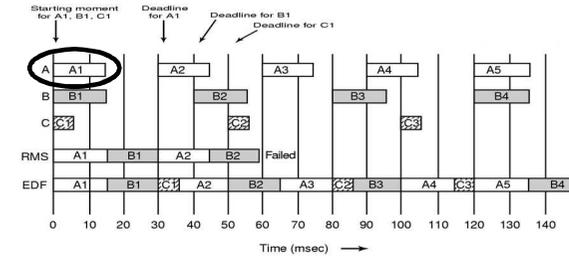
CENTRE NATIONAL DE LA RECHERCHE SCIENTIFIQUE



UNIVERSITÉ PARIS UNIVERSITÉS

## Ordonnement : exemple 2

Process	Period (ms)	CPU Time (ms)	Priority	Deadline (ms)
A	30	15	33	30
B	40	15	25	40
C	50	5	20	50



Laboratoire d'Informatique de Paris 6



CENTRE NATIONAL DE LA RECHERCHE SCIENTIFIQUE



UNIVERSITÉ PARIS UNIVERSITÉS

## Systèmes de fichiers

- Transfert rapide
- Stockage élevé
- Données fournies sans délai :
  - Serveur multimédia (VoD)
  - Difficile si rien n'est planifié
- Pas d'opération de lecture mais uniquement un start
  - Ensuite le serveur envoie au taux fixé
  - L'utilisateur gère le taux d'affichage
  - L'utilisateur peut faire stop ou pause
  - Retour en arrière = start depuis le début



Laboratoire d'Informatique de Paris 6



CENTRE NATIONAL DE LA RECHERCHE SCIENTIFIQUE



UNIVERSITÉ PARIS UNIVERSITÉS

## Avance/retour rapide

- Jouer à une vitesse plus rapide (impossible en général)
- Données non compressées :
  - Jouer une frame sur k
- Données compressées :
  - Si indépendance des frames : indexation pour trouver les k-ièmes frames
  - MPEG : I, P, et B
    - Décompresser et recompresser toutes les k frames comme une I
    - Débit plus élevé
    - Prévion : créer un MPEG avec seulement une frame sur k
  - Séparation entre audio et vidéo



Laboratoire d'Informatique de Paris 6



CENTRE NATIONAL DE LA RECHERCHE SCIENTIFIQUE



UNIVERSITÉ PARIS UNIVERSITÉS

## Utilisateurs multiples

- $k$  utilisateurs regardant le même film =  $k$  utilisateurs regardant des films différents
- Near VoD :
  - Films à des horaires fixés
  - Pas de fonctions avance/retour rapide



Laboratoire d'Informatique de Paris 6



CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UPMC  
PARIS UNIVERSIT S

## Avance/retour rapide

- Utilisation de buffers :
  - Taux de 4Mbps : buffer de 10mn = 300Mo
- En cas d'avance rapide :
  - D passement du buffer : activation d'un flux priv 
  - Remplissage   partir du flux le plus proche
  -   la reprise :
    - Jouer depuis le buffer
    - Supprimer le flux priv 
    - Remplissage du buffer classique



Laboratoire d'Informatique de Paris 6



CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UPMC  
PARIS UNIVERSIT S

## Placement des fichiers

- Fichiers multim dia :
  - Grande taille
  -  crits une fois mais lus souvent
  - Lecture s quentielle
- Sur un disque :
  - Placement contigu
  - Mod le   petits blocs
  - Mod le   grands blocs



2/3/2009

Laboratoire d'Informatique de Paris 6



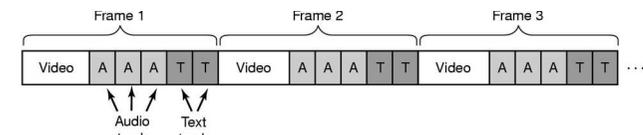
CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UPMC  
PARIS UNIVERSIT S

## Placement contigu

- Simple
- Ok si planifi    l'avance sans changement
- Pose probl me si vid o/audio/texte sont stock s dans des fichiers diff rents
- Acc s al atoire, avance/retour rapide impossible



Laboratoire d'Informatique de Paris 6



CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UPMC  
PARIS UNIVERSIT S

## Modèles à grands/petits blocs

- Stockage non contigu :
  - Remplacement de frames simple, avance/retour rapide ok
- Grands blocs : plusieurs frames par bloc
  - Taille constante
  - Index de blocs
  - Stockage éventuel du numéro de la première frame de chaque bloc
- Petits blocs : une frame (audio/vidéo/texte) par bloc
  - Temps de lecture constant
  - Index de frames



Laboratoire d'Informatique de Paris 6



CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UNIVERSITÉ PARIS  
SUD

## Grands blocs

- Aspects positifs :
  - Peu d'entrées dans l'index
    - Blocs de 256Ko, 16Ko par frame : environ 16 par bloc
    - 216,000 frames au total : 13500 entrées
  - Bonne utilisation du disque
- Aspects négatifs :
  - N frames ne tiennent pas exactement dans un bloc
    - Espace perdu ou frame découpées
  - Double buffering plus difficile
  - Avance/retour rapide : utilisation élevée du disque



Laboratoire d'Informatique de Paris 6



CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UNIVERSITÉ PARIS  
SUD

## Petits blocs

- Aspects positifs :
  - Peu d'espace disque perdu
  - Double buffering facile (frame courante ou suivante)
  - Avance rapide facile (uniquement frames I)
- Aspects négatifs :
  - Index de grande taille
  - Stockage des trous (temps constant) :
    - Index des trous à remplir avec des frames si possible



Laboratoire d'Informatique de Paris 6



CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UNIVERSITÉ PARIS  
SUD

## Plusieurs fichiers

- Near VoD
  - Films de 2h avec 5mn d'intervalle
  - 24 flux : frames stockées ensemble
- Films avec popularité variable
  - Loi de Zipf
    - Probabilité que le prochain client choisisse le k-ème film le plus populaire est  $C/k$
    - $C/1 + C/2 + \dots + C/N = 1$
    - C constante de normalisation
  - Stockage des fichiers populaires au centre du disque



Laboratoire d'Informatique de Paris 6



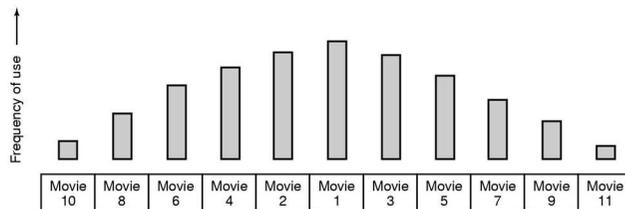
CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UNIVERSITÉ PARIS  
SUD

## Stockage

- $N=1000$ ,  $P(5 \text{ plus regardés}) = 0.307$

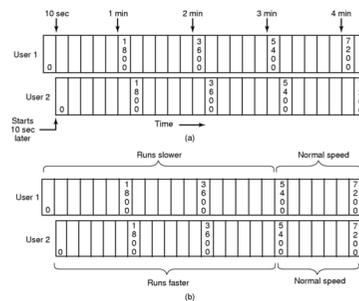


## Mise en cache

- Fichiers multimédia lus du début à la fin
  - LRU ne convient pas
- Deux approches :
  - Mise en cache de blocs
  - Mise en cache de fichiers

## Mise en cache de blocs

- De nombreux usagers regardent simultanément
  - Blocs stockés jusqu'à être vus par tous
  - La désynchronisation entre usagers détermine la taille du cache
- Resynchroniser les flux
  - Accélérer l'un et ralentir l'autre



## Mise en cache de fichiers

- Fichiers très volumineux :
  - Pas tous stockés sur disque
  - DVD, cassettes, etc.
- Transféré sur disque à la demande :
  - Mise en route longue
- Première minutes stockées sur disque :
  - Copie sur disque de manière transparente

## Ordonnancement du disque

- Données multimédia souvent prédictibles
  - lecture séquentielle
- Vision statique : les films ont la même vitesse de lecture
  - Diviser les requêtes en rounds
  - Rounds traités l'un après l'autre
- Vision dynamique : vitesse (débit) variable
  - Deadline sur les requêtes (maintenir le frame rate)
  - Scan-EDF
    - EDF : requête avec la deadline la plus proche
    - SCAN : requêtes sur le trajet de la tête de lecture
    - Requêtes dont la deadline est proche, traitement dans l'ordre du cylindre



Laboratoire d'Informatique de Paris 6



CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UNIVERSITÉ PARIS  
SUD - PARIS  
NORD

## Programmation multimédia



Laboratoire d'Informatique de Paris 6

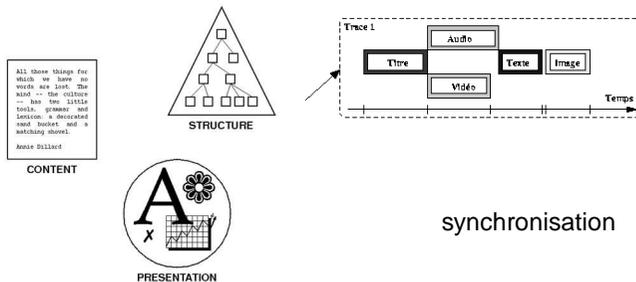


CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UNIVERSITÉ PARIS  
SUD - PARIS  
NORD

## Un document multimédia



Laboratoire d'Informatique de Paris 6



CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UNIVERSITÉ PARIS  
SUD - PARIS  
NORD

## Unités de présentation

- Objets composés d'une séquence d'unités élémentaires
  - Logical Data Units (LDU)
- LDUs structurés :
  - Hiérarchie décrivant le contenu (XML)
  - Hiérarchie décrivant le codage (mpeg)



Laboratoire d'Informatique de Paris 6

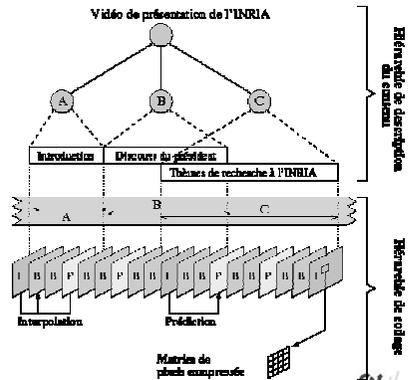


CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UNIVERSITÉ PARIS  
SUD - PARIS  
NORD

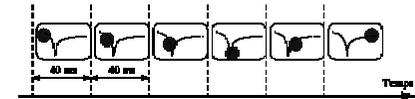
## Un exemple de LDUs



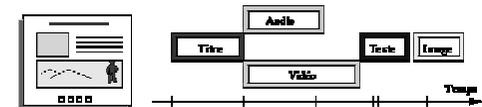
## Synchronisation multimédia

- Intra ou inter-objets

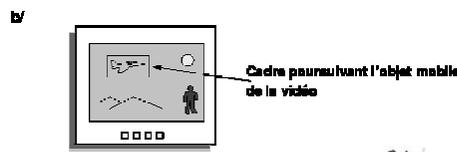
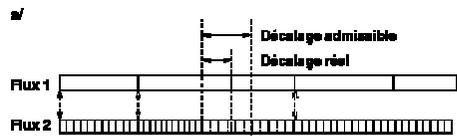
(a) Synchronisation Intra-objet



(b) Synchronisation Inter-objet



## Synchronisation multimédia

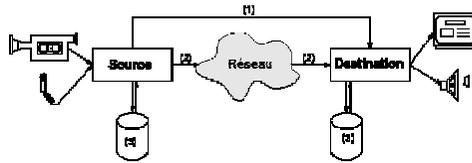


## Valeurs typiques

Média		Couplage	Décalage
vidéo	animation	corrélés	120ms
	audio	synchro des lèvres	80ms
	image texte	annotation superposée annotation non superposée	240ms 500ms
audio	animation	corrélés	80ms
	audio	fort (stéréo)	11us
		faible (plusieurs orateurs)	120ms
		très faible (musique de fond)	500ms
	image	fort (musique avec notes) faible (audio avec diapositives)	5ms 500ms
	texte	annotations textuelles	240ms
curseur graphique	commentaire associé à un élément désigné	500ms	

## Sur un réseau

- Synchronisation naturelle (intrinsèque aux média) :



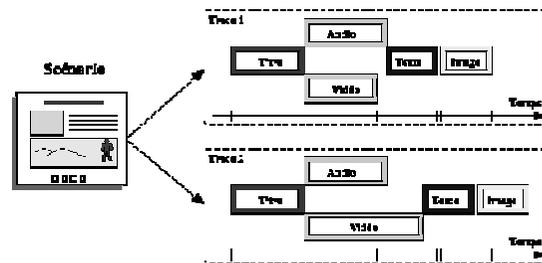
- Source : un fichier, capture vidéo, audio, ...
- Canal réseau : gigue, pertes, engorgement
- Destination : ressources limitées

## Notion de scénario temporel

- Un scénario temporel :
  - objets média qui s'enchaînent dans le temps
  - plusieurs traces d'exécution qui respectent ou pas une spécification de la synchronisation

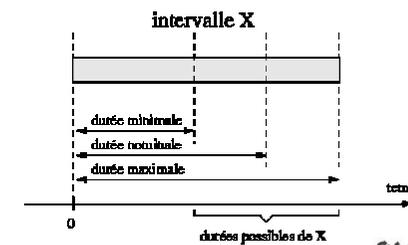
## Notion de scénario temporel

- Scénario déterministes ou pas



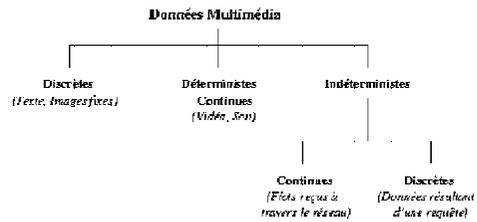
## Information temporelle

- Synchronisés grâce à :
  - Instants de début et de fin
  - Durée de l'objet



# Information temporelle

- Instants vs. intervalles

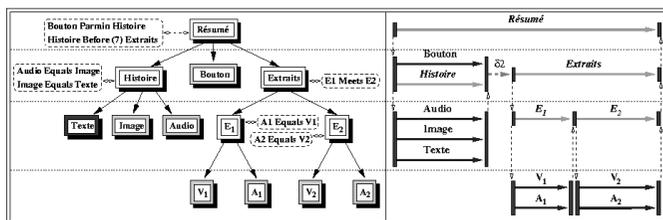


# Modèle synchrone

Relations	Représentation Graphique	Contraintes Numériques
A égale B		$dur(A) = dur(B)$
A before B		---
A dans [t1, t2] B		$t < dur(B) - dur(A)$ $\& t > 0 \ \& \ t > 0$
A overlaps B		$t > dur(A) - dur(B)$ $\& t > 0 \ \& \ t > 0$
A inside B		---
A starts B		$dur(A) < dur(B)$
A finishes B		$dur(A) < dur(B)$

# Scénario temporel

- STP (Simple temporal problems) :  
– passage d'intervalles à instants



# SMIL

## Synchronized Multimedia Integration Language

## Introduction

- Tendance actuelle :
  - Diversité des formats et des plates formes :
    - pas de format d'échange, pérennité, ...
  - Multimédia et web se développent en parallèle
- Multimédia sur le web - problèmes d'intégration :
  - Médias entre eux (mp3, vidéo, texte, ..)
  - Avec le web (insérer vidéo, permettre de la synchro)
- Cadre du travail W3C
  - SYMM Working Group (SMIL 1.0 et 2.0)

## Objectifs

- Format texte avec intégration d'objets média
- Basé sur des technologies classiques :
  - XML, etc.
- Notion de documents temporisés et de synchronisation
- Neutralité vis-à-vis des protocoles d'accès réseaux et formats des médias RTP, RTSP, Mpeg,...
- Rassembler les industriels du multimédia et du web autour d'un format ouvert (le défi)

## Organisations impliquées

- Oratrix, Real Networks, Microsoft, IBM, Macromedia, Intel, Philips, Panasonic, Nokia Produits
- Institution publiques: INRIA, CWI, NIST, WGBH ...Syst. Expérimentaux
- Points forts :
  - Version 1.0 est un succès relatif ...
  - Langage très simple
  - Plus en plus d'intégration avec les autres standards du web



## Structure d'un document SMIL

- Composants accessibles via des urls :
  - En dehors du fichier SMIL
  - Ces composants peuvent avoir des types de médias différents : audio, vidéo, texte, image, etc.
- Méthode de synchronisation
- Interactions des utilisateurs : CAT (Global) et liens spatio-temporels, changements dynamiques dans la présentation (événements)



Laboratoire d'Informatique de Paris 6



CENTRE NATIONAL DE LA RECHERCHE SCIENTIFIQUE



UNIVERSITÉ PARIS UNIVERSITÉS

## Organisation d'un document

- Deux parties :
  - Entête : contient des informations du niveau du document
  - Corps : contient le scénario temporel, les animations, les transitions et les objets media utilisés



Laboratoire d'Informatique de Paris 6



CENTRE NATIONAL DE LA RECHERCHE SCIENTIFIQUE



UNIVERSITÉ PARIS UNIVERSITÉS

```

<smil xmlns="http://www.w3.org/2000/SMIL20/Language">
<head>
  <layout type="text/smil-basic">
    <region id="left-video" left="20" top="50" z-index="1"/>
    <region id="left-text" left="20" top="120" z-index="1"/>
    <region id="right-text" left="150" top="120" z-index="1"/>
  </layout>
</head>

<body>
<par>
  <seq>
    
    <text src="graph-text" region="left-text"/>
  </seq>
  <par>
    <a href="http://www.w3.org/People/Berners-Lee">
      <video src="tim-video" region="left-video"/>
    </a>
    <text src="tim-text" region="right-text"/>
  </par>
  <seq>
    <audio src="joe-audio"/>
    <video id="jv" src="joe-video" region="right-video"/>
  </seq>
</par>
</body>
</smil>
    
```

Entête

Corps



Laboratoire d'Informatique de Paris 6



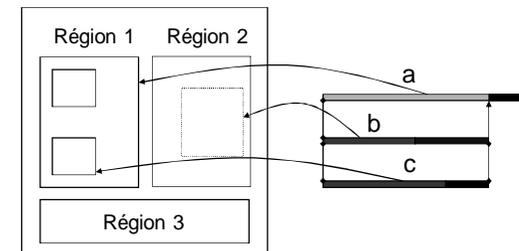
CENTRE NATIONAL DE LA RECHERCHE SCIENTIFIQUE



UNIVERSITÉ PARIS UNIVERSITÉS

## Entête : aspect spatial

- Régions et sous-régions (hiérarchie)



Laboratoire d'Informatique de Paris 6



CENTRE NATIONAL DE LA RECHERCHE SCIENTIFIQUE



UNIVERSITÉ PARIS UNIVERSITÉS

## Corps du document

- Scénario temporel (synchronisation)
  - défini récursivement : Schedule elements
- Schedule = Parallel | Seq | Excl
  - | Media object
  - | ancres (dep/arr)
  - | Switch
  - | priorityClass
  - | Prefetch



Laboratoire d'Informatique de Paris 6



CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UPMC  
UNIVERSITÉ PARIS  
UNIVERSITAS

## Objets

- Marqués avec les balises :
  - Audio, Video, Text, Img, Textstream, Animation, Ref, Param, et ... Prefetch
- Attributs :
  - Src : localisation du fichier (URL)
    - `rtsp://rtsp.example.org/video.mpg`
  - Type : type mime (par exemple video/mpeg)
  - Region : identifiant d'une surface d'affichage
  - Dur : durée de l'objet média



Laboratoire d'Informatique de Paris 6



CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UPMC  
UNIVERSITÉ PARIS  
UNIVERSITAS

## Synchronisation

- L'attribut dur (duration)
  - "intrinsic": durée du média (durée du fichier externe).
  - "explicit": durée spécifiée (dur="15s")

- L'attribut repeat
  - RepeatCount="3": durée du média



- RepeatDur="12 s":



Laboratoire d'Informatique de Paris 6



CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UPMC  
UNIVERSITÉ PARIS  
UNIVERSITAS

## Synchronisation

- Attribut begin, end
  - Valeur = décalage par rapport à l'élément père
    - begin= "13 s"
  - Autre horloge
    - begin= "e2.end + 5 s"
  - Temps absolu
    - begin= "wallclock(2001-01-01Z)"
  - Événement asynchrone (interactivité)
    - begin = "bouton.click"



Laboratoire d'Informatique de Paris 6



CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE

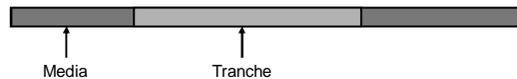


UPMC  
UNIVERSITÉ PARIS  
UNIVERSITAS

## Découpage des media

- Clipping spatial effectué à travers les régions et sous-régions
- Clipping temporel effectué avec les attributs clip-begin et clip-end (médias externes)

```
<video id="a" src="attendre.mpg"
clip-begin="smpte=00:01:45"
clip-end="smpte=00:01:55"
...
/>
```



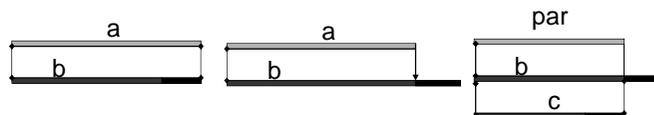
## L'élément séquentiel : seq

- Jouer en séquence un ensemble d'objets
- Attributs
  - Fill : utilisé pour la « persistance » sur l'écran
  - Remove : effacer de l'écran dès la terminaison
  - Freeze : garder la dernière image après terminaison

```
<seq>
<image id="a" regionName="x" src="attendre.gif"
fill="freeze"/>
<video id="b" regionName="x" src="video.au dur="20 s" />
</seq>
```

## L'élément parallèle : par

- Jouer en parallèle un ensemble d'objets
  - Terminaison fixée
  - Attributs :



## L'élément switch

- Un élément à choisir parmi un ensemble d'éléments alternatifs
- Choix est basé sur des valeurs d'attributs
  - language, screen size, depth, bitrate, systemRequired
  - ...et des préférences de l'utilisateur

```
...
<switch>
<audio src="joe-audio-better-quality" language="fr"/>
<audio src="joe-audio" language="en"/>
</switch>
...
```

## Transitions

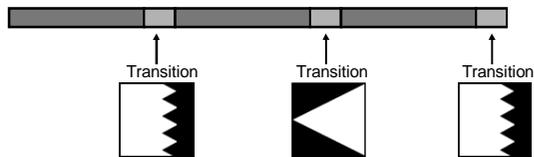
```

...
<transition id="wipe1" type="zigZagWipe" subtype="leftToRight" dur="1s"/>
<transition id="wipe2" type="veeWipe" subtype="leftToRight" dur="1s"/>
...
<seq>



</seq>

```

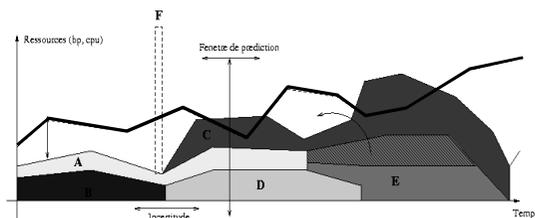


## Liens hypermédia temporisés

- Extension de la sémantique des URLs
  - `http://foo.com/path.smil#ancre(begin(id(ancre)))`
  - deux types (a: totalité, area: partie)
  - sauts dans l'espace et dans le temps
- Attribut show
  - Replace (valeur par défaut)
  - New (fork)
  - Pause (Appel de procédure)

## Préchargement

- Réduire les délais de chargement :
  - Lier ordonnancement et gestion de ressources (BP, CPU)
  - Scénario pour la gestion de ressources à partir du scénario temporel



## Exemple

- Préchargement d'une image (affichage juste après la vidéo) :

```

<smil:lns="http://www.w3.org/2001/SMIL20/CR/Language">
<body>
<seq>
<par>
<prefetch id="endimage" src="tes.gif"/>
<text id="interlude" src="pleasewait.html" fill="freeze"/>
</par>
<video id="main-event" src="rtsp://www.example.org/video.mpg"/>

</seq>
</body>
</smil>

```

## Conclusions

- Grande simplicité d'utilisation :
  - Proche du html, xml
  - Format ouvert
- Impact grandissant mais pas idéal :
  - Browsers HTML pas engagés
  - Lecture avec Real One Player notamment
  - Conception avec LimSee II



Laboratoire d'Informatique de Paris 6



CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UPMC  
UNIVERSITÉ PARIS UNIVERSITAS

## Localisation des médias Systèmes Pair-à-pair

### Partie 3



Laboratoire d'Informatique de Paris 6



CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UPMC  
UNIVERSITÉ PARIS UNIVERSITAS

## Deux problèmes de localisation

- Où placer le média ?
  - les CDNs
- Comment trouver le média ?
  - les DHTs



Laboratoire d'Informatique de Paris 6



CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UPMC  
UNIVERSITÉ PARIS UNIVERSITAS

## Où placer le média ?

- Gros fichiers et forte demande
  - Surcharge pour un serveur centralisé
- Effets d'un réseau moindre effort
  - Délais, gigue, pertes quand le serveur est loin du récepteur
- Solutions : Réplication, proximité vers l'utilisateur
- Implémentation : les CDNs
  - « Content Distribution Networks »
  - Réseaux de diffusion de contenu
  - Exemple : Akamai



Laboratoire d'Informatique de Paris 6



CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UPMC  
UNIVERSITÉ PARIS UNIVERSITAS

## Comment trouver le média ?

- Emplacement dynamique
  - Exemple : réseaux poste à poste
    - Multiples copies d'un fichier stockées
    - Postes qui se connectent, se déconnectent, se déplacent
  - Indexation classique par moteur de recherche impossible
- Solution : localisation dynamique
- Implémentation : les DHTs (systèmes P2P structurés)
  - « Distributed Hash Tables »
  - Tables de hachage distribuées
  - Exemples : Tapestry, Pastry, Chord, CAN



Laboratoire d'Informatique de Paris 6



CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UNIVERSITÉ PARIS  
SUD 11 PARIS UNIVERSITÉS

## Content distribution networks



Laboratoire d'Informatique de Paris 6

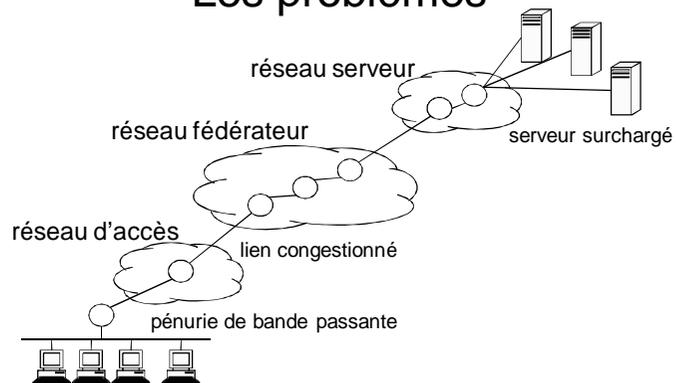


CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UNIVERSITÉ PARIS  
SUD 11 PARIS UNIVERSITÉS

## Les problèmes



Laboratoire d'Informatique de Paris 6

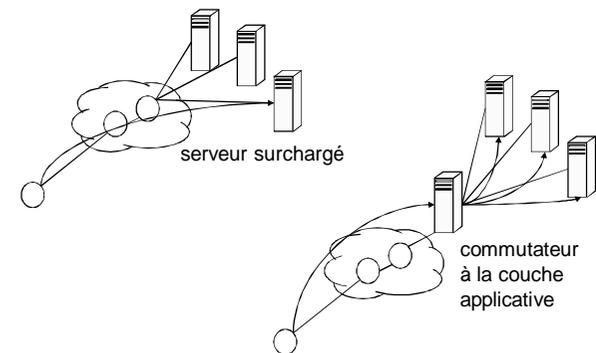


CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UNIVERSITÉ PARIS  
SUD 11 PARIS UNIVERSITÉS

## Équilibrage de charge



Laboratoire d'Informatique de Paris 6



CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UNIVERSITÉ PARIS  
SUD 11 PARIS UNIVERSITÉS

## Équilibrage de charge

- Avantages
  - Décharge un serveur surchargé
- Désavantages
  - N'aide pas au niveau de délai ou gigue
  - Ne résoud pas la congestion dans le réseau



Laboratoire d'Informatique de Paris 6

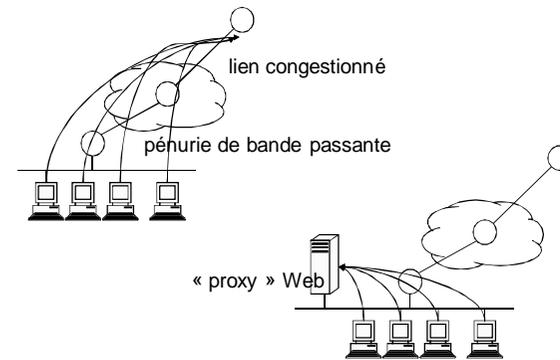


CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UNIVERSITÉ PARIS  
UNIVERSITAS

## Serveurs mandataires



Laboratoire d'Informatique de Paris 6



CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UNIVERSITÉ PARIS  
UNIVERSITAS

## Serveurs mandataires

- Avantages
  - Délais plus courts
  - Enlève la congestion
  - Décharge le serveur principal
- Désavantages
  - N'aide pas les clients sans proxy
  - Hors contrôle du serveur principal
    - copies à jour ?
    - facturation ?



Laboratoire d'Informatique de Paris 6

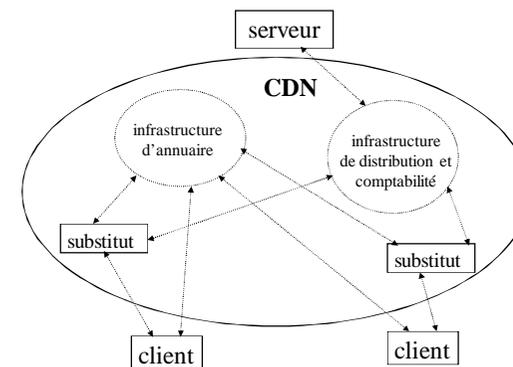


CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UNIVERSITÉ PARIS  
UNIVERSITAS

## CDN : architecture



Laboratoire d'Informatique de Paris 6



CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UNIVERSITÉ PARIS  
UNIVERSITAS

## CDN : Composants

- Les substituts (mise en cache)
  - distribution de données aux clients
- Infrastructure d'annuaire
  - dirige le client vers le substitut
- Infrastructure de distribution
  - distribution de données serveur → substituts
- Infrastructure de comptabilité
  - récolte d'informations clients
  - tarification de serveur



Laboratoire d'Informatique de Paris 6



CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE

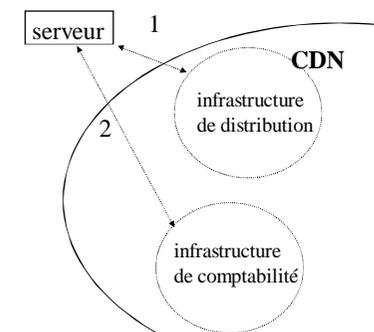


UNIVERSITÉ PARIS  
UNIVERSITAS

## Interaction serveur – CDN

1. Le serveur pousse les données vers le CDN  
*OU*  
Le CDN tire les données du serveur
2. Le serveur demande de l'information et de la comptabilité du CDN  
*OU*  
Le CDN fournit de l'information et de la comptabilité au serveur

*fr.news.yahoo.com*



Laboratoire d'Informatique de Paris 6



CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UNIVERSITÉ PARIS  
UNIVERSITAS

## Fonctionnement

- Seul du contenu statique est utilisable par le système
- Le nom modifié contient le nom de fichier original et l'identifiant du fournisseur :
  - <http://a73.g.akamaitech.net/7/23/>
  - [cnn.com/af/x.gif](http://cnn.com/af/x.gif)
- En cas de requête sur Akamai :
  - Vérification de l'existence du fichier dans le cache
  - Si absent, requête sur le fournisseur réel et mise en cache



Laboratoire d'Informatique de Paris 6



CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UNIVERSITÉ PARIS  
UNIVERSITAS

## Infrastructure d'annuaire

- Basé sur le DNS
  - infrastructure déjà existante
- Ajout d'un serveur DNS spécialisé
  - Choix en fonction de :
    - Mesures de délai, latence, gigue, etc.
    - Politiques



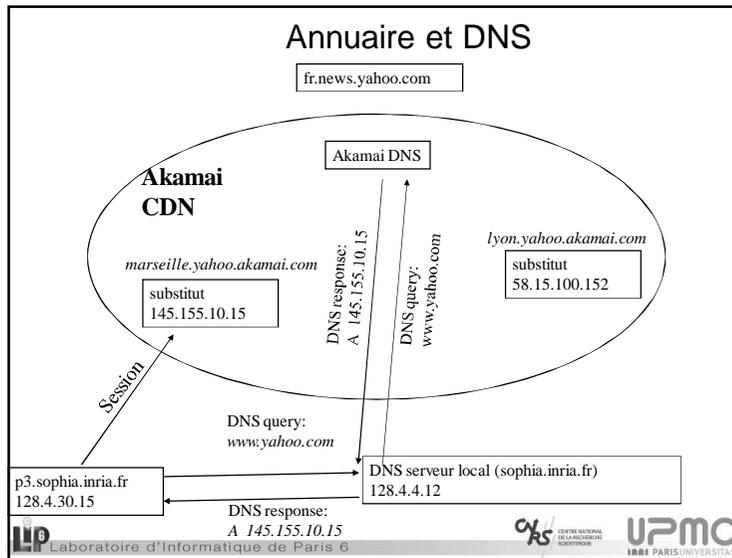
Laboratoire d'Informatique de Paris 6



CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UNIVERSITÉ PARIS  
UNIVERSITAS



## Choix du substitut

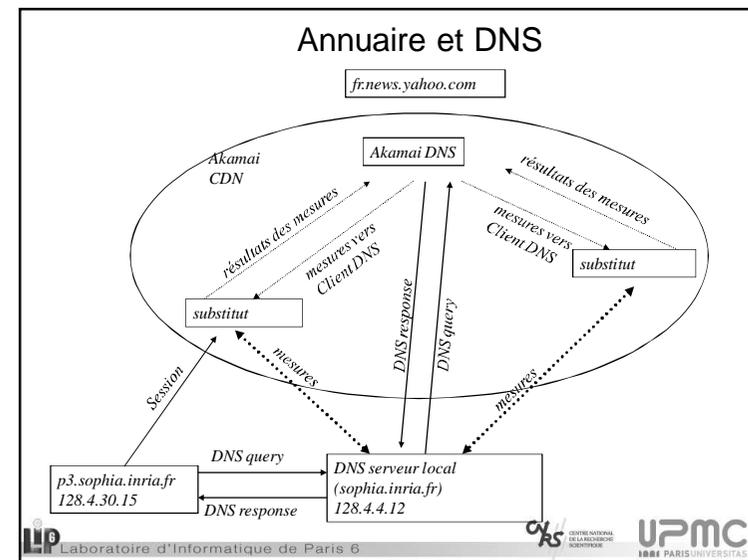
- Soit le document XYZ
  - Numérotation des serveurs de 0 à n
  - Le document XYZ est placé sur le serveur (XYZ mod n)
  
- Problème en cas de panne : n n-1
  - Renumérotation de tous les fichiers
  - Fichiers plus demandés : serveurs surchargés
  - Hachage consistant (charge bien répartie, résistant aux pannes, etc.)

Laboratoire d'Informatique de Paris 6

## Proximité géographique

- Comment choisir le plus proche ?
  - Même ISP, même AS ?
  - Utiliser BGP pour avoir une connaissance du réseau
  - Triangulation des clients avec ping
  - Mesure QoS (gigue, délai, débit)

Laboratoire d'Informatique de Paris 6



## Réseaux pair-à-pair

### Localisation de contenus et DHTs



Laboratoire d'Informatique de Paris 6



CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UPMC  
UNIVERSITÉ PARIS UNIVERSITAS

## Applications du P2P

- Partage de fichiers (musique, films, etc.)
- Calcul distribué
- VoIP (Skype)
- TV, VoD
- Travail collaboratif



Laboratoire d'Informatique de Paris 6



CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UPMC  
UNIVERSITÉ PARIS UNIVERSITAS

## Caractéristiques

- Large échelle : millions de noeuds
- Dynamique : arrivées et départs fréquents
- Peu ou pas d'infrastructure
  - Aucun serveur central
  - Éventuellement des serveurs de connexion
- Symétrique : tous les nœuds sont équivalents
  - Parfois une hiérarchie est introduite
- Communication possible entre tous les pairs
  - Pas toujours vrai : routeurs NAT, firewall



Laboratoire d'Informatique de Paris 6



CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UPMC  
UNIVERSITÉ PARIS UNIVERSITAS

## Attractivité

- Coût faible :
  - Les utilisateurs partagent leurs ressources
  - Peu ou pas de serveurs d'infrastructure
- Disponibilité forte
  - Moyens de calcul :
    - Nombreuses machines connectées simultanément
  - Moyens de stockage :
    - Si les fichiers sont suffisamment répliqués, ils sont toujours disponibles chez de nombreux usagers.



Laboratoire d'Informatique de Paris 6



CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UPMC  
UNIVERSITÉ PARIS UNIVERSITAS

## Services offerts

- Entrer dans le réseau
- Quitter le réseau
- Partage de fichiers :
  - Publication :
    - insérer un nouveau fichier, offrir des ressources
  - Recherche :
    - trouver les pairs fournissant des ressources ou un fichier de nom X (efficacité, fiabilité, passage à l'échelle)
  - Récupération :
    - obtenir une copie d'un fichier
- Similaire pour les autres applications



Laboratoire d'Informatique de Paris 6



CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UNIVERSITÉ PARIS  
SUD - PARIS  
UNIVERSITAS

## Overlay P2P

- Réseau
  - de niveau applicatif
  - virtuel :
    - La topologie sous-jacente est transparente pour l'utilisateur
    - Les liens sont des connexions TCP ou des entrées dans une table de voisins
- Le réseau doit être maintenu en permanence :
  - mes voisins sont-ils toujours actifs ?



Laboratoire d'Informatique de Paris 6



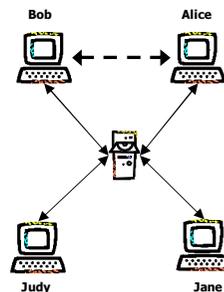
CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UNIVERSITÉ PARIS  
SUD - PARIS  
UNIVERSITAS

## Systèmes centralisés

- Type Napster
- Avantages :
  - Recherche efficace
  - Bande passante limitée
  - Pas d'information sur les nœuds
- Inconvénients :
  - Nœud central sensible
  - Passage à l'échelle limité



Laboratoire d'Informatique de Paris 6



CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UNIVERSITÉ PARIS  
SUD - PARIS  
UNIVERSITAS

## Overlays non structurés

- Les overlays sont des structures génériques, on peut les utiliser comme on veut avec des opérations d'insertion de nœuds, de suppression, de recherche de contenu...
- Avantages
  - Flexibilité : pas de structure => arrivées et départs simples
  - Requêtes complexes
- Inconvénients :
  - Recherche de contenus difficile – inondation
    - horizon limité
    - Tradeoff entre l'overhead et la taille de l'overlay accessible



Laboratoire d'Informatique de Paris 6



CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE

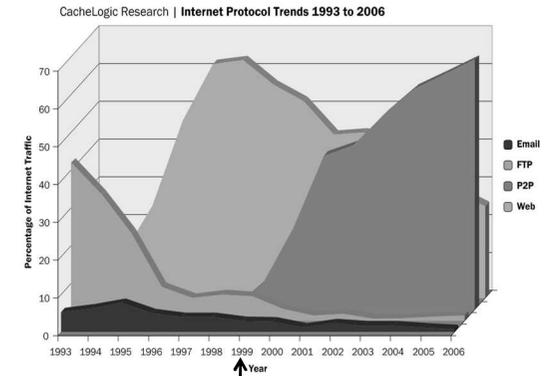


UNIVERSITÉ PARIS  
SUD - PARIS  
UNIVERSITAS

## Problèmes du non structuré

- Inefficace
  - Les requêtes sont propagées par inondation
  - Même avec un routage “intelligent”, le pire cas est toujours en  $O(n)$  (messages envoyés)
- Incomplet
  - En l’absence de résultat, est-ce parce qu’il n’y en a pas ?
- Améliorations possibles mais malgré tout...

## Évolution

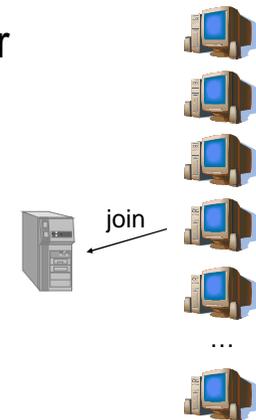


## Les origines : Napster (1999)

- Objectif : application P2P pour le partage de fichiers mp3
  - Chaque utilisateur peut contribuer avec ses propres fichiers
- Fonctionnement :
  - Serveur central maintenant une liste des pairs actifs et de leurs fichiers.
- Stockage et téléchargement distribué :
  - Les clients font office de serveurs de fichiers
  - Tout le contenu téléchargé est partagé

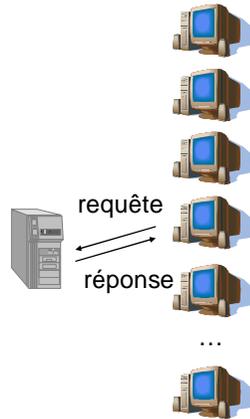
## Napster

- Connexion :
  - Connexion au serveur
  - Transmission de la liste de fichiers partagés
- Recherche :
  - Envoi de requêtes au serveur
  - Recherche dans la base et envoi d’une liste de pairs
- Récupération :
  - Contact direct du/des pair ayant le fichier
  - Prise en compte du taux de transfert



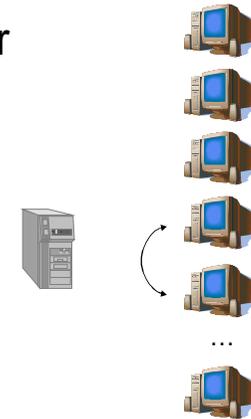
## Napster

- Connexion :
  - Connexion au serveur
  - Transmission de la liste de fichiers partagés
- Recherche :
  - Envoi de requêtes au serveur
  - Recherche dans la base et envoi d'une liste de pairs
- Récupération :
  - Contact direct du/des pair ayant le fichier
  - Prise en compte du taux de transfert



## Napster

- Connexion :
  - Connexion au serveur
  - Transmission de la liste de fichiers partagés
- Recherche :
  - Envoi de requêtes au serveur
  - Recherche dans la base et envoi d'une liste de pairs
- Récupération :
  - Contact direct du/des pair ayant le fichier
  - Prise en compte du taux de transfert



## Napster

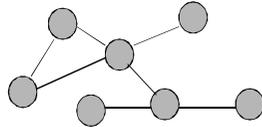
- Avantages :
  - Peu d'overhead
  - Succès garanti (si le fichier est publié, il sera trouvé)
- Inconvénients
  - Serveur central (critique)
  - Passe mal à l'échelle

## Après Napster

- Protocoles de téléchargement de fichiers :
  - 1999 : Napster
  - 2000 : Gnutella, eDonkey
  - 2001 : Kazaa
  - 2002 : eMule, BitTorrent
  - 2003 : Skype
  - 2004 : PPLive
  - TV : TVKoo, TVAnts, PPStream, SopCast
  - A venir : VoD, jeux applications distribuées
- Tentative de décentralisation à plus ou moins forte échelle

## Overlays non structurés

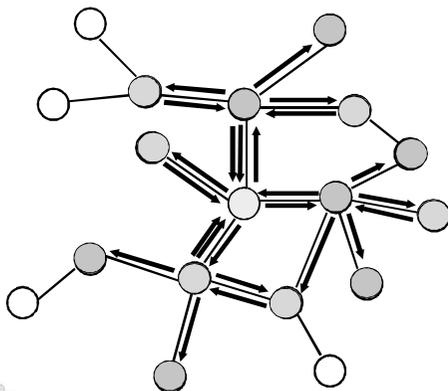
- Pas de structure fixée a priori
  - Organisation des pairs en une structure aléatoire
    - Ex : chaque pair choisi 3 voisins au hasard
    - Structure plate (tous égaux) ou hiérarchique
  - Le service P2P est construit sur ce graphe
- Approches classiques :
  - Gnutella
  - KaZaA/FastTrack
  - BitTorrent



## Gnutella

- Application décentralisée, chaque pair :
  - Stocke des fichiers
  - Route les requêtes (recherche de fichiers) de/vers ses voisins
  - Répond au requêtes s'il possède les fichiers
- Principe = recherche par inondation :
  - Si le pair n'a pas le fichier il transmet à 7 de ses voisins.
  - Idem pour les voisins avec une profondeur 10 maximum.
  - Pas de boucles mais les paquets peuvent être reçus plusieurs fois.

## Inondation



- = source
- ↑ = envoi requête
- = traitement
- = résultat
- ↑ = envoi réponse

## Gnutella

- Avantages :
  - Recherche rapide
  - Arrivée et départ simples
  - Les fichiers populaires sont trouvés avec un petit TTL
    - Permet de récupérer depuis plusieurs sources.
- Inconvénients
  - TTL limité => pas de garantie de trouver un fichier disponible
  - Overhead élevé pour la recherche
    - Passage à l'échelle peu aisé
  - Charge répartie de manière non uniforme

## FastTrack, KaZaA, eDonkey

- Résolution du problème d'échelle avec une hiérarchie à 2 niveaux
- Clients : nœuds standards
- Super-pairs avec plus de ressources :
  - Un client peut (ou pas) décider d'être un super pair à la connexion
  - Acceptent des connexions de clients et peuvent les mettre en contact
  - Gèrent les recherches dans un overlay de SP



Laboratoire d'Informatique de Paris 6



CENTRE NATIONAL DE LA RECHERCHE SCIENTIFIQUE



UPMC PARIS UNIVERSIT S

## FastTrack, KaZaA, eDonkey

- Avantages
  - Plus stable
  - Plus complet (les recherches dans l'overlay les super-nœuds permettent d'explorer plus complètement)
  - Passe mieux à l'échelle
- Inconvénients
  - Pas réellement P2P (super nœuds plus sensibles)
    - eDonkey : les super-pairs sont de serveurs
  - Overhead important



Laboratoire d'Informatique de Paris 6

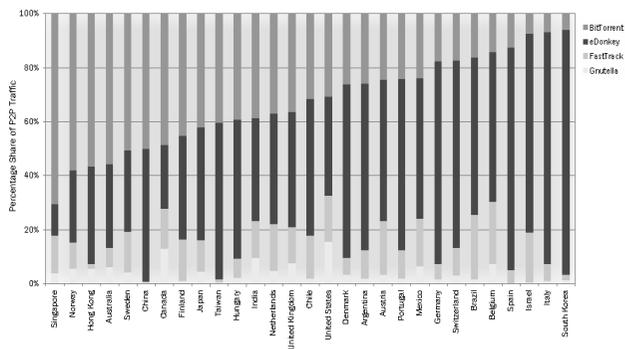


CENTRE NATIONAL DE LA RECHERCHE SCIENTIFIQUE



UPMC PARIS UNIVERSIT S

##  tat en 2004



Laboratoire d'Informatique de Paris 6



CENTRE NATIONAL DE LA RECHERCHE SCIENTIFIQUE



UPMC PARIS UNIVERSIT S

## Un exemple suppl mentaire

### Bittorrent



Laboratoire d'Informatique de Paris 6



CENTRE NATIONAL DE LA RECHERCHE SCIENTIFIQUE



UPMC PARIS UNIVERSIT S

## Bittorrent - principes

- Défauts des systèmes classiques :
  - Vision client/serveur
    - Ne passe pas à l'échelle
  - Free-riding dans les réseaux P2P
    - Téléchargent sans contribuer
- Bittorrent : système P2P avec mécanismes d'incitation au partage.



Laboratoire d'Informatique de Paris 6



CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UPMC  
UNIVERSITÉ PARIS UNIVERSITAS

## Eléments de base

- Serveurs web :
  - Fournissent les fichiers d'infos (.torrent)
  - Contiennent : nom, taille, checksum, @IP du tracker
- Trackers :
  - Nœuds ne partageant pas de contenu
  - Liste des pairs partageant le contenu
  - Partie critique
- Utilisateurs :
  - Downloaders : pairs qui téléchargent
  - Seeders : pairs qui possèdent le fichier entier mais continuent de le fournir



Laboratoire d'Informatique de Paris 6



CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UPMC  
UNIVERSITÉ PARIS UNIVERSITAS

## Fichiers

- Les fichiers sont découpés en morceaux de taille fixe (typiquement 256Ko)
- Chaque downloader annonce aux autres pairs les morceaux qu'il possède
- Des codes de hachage sont utilisés pour tous les morceaux afin de garantir leur intégrité.
- Téléchargement des morceaux :
  - « Rarest order first » (avec randomisation pour que tout le monde ne veuille pas le même)
  - Méthode pour accélérer la fin du téléchargement
  - Principe « un prêté pour un rendu »



Laboratoire d'Informatique de Paris 6



CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UPMC  
UNIVERSITÉ PARIS UNIVERSITAS

## Mécanismes « incitatifs »

- Au plus 4 uploads simultanés
  - Éviter d'avoir trop de connexions TCP ouvertes simultanément
    - Contrôle de congestion difficile avec beaucoup de connexions partageant la BP.
  - Réévalué toutes les 10 secondes
  - Basé sur le taux de téléchargement :
    - 4 pairs intéressés avec le meilleur taux de téléchargement (réciprocité)



Laboratoire d'Informatique de Paris 6



CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UPMC  
UNIVERSITÉ PARIS UNIVERSITAS

## Mécanismes « incitatifs »

- Optimistic Unchoking
  - Un pair est « unchoked » toutes les 30s
  - Indépendant du taux de téléchargement
  - Permet de tester des connexions non utilisées
  - Rend possible le free-riding



Laboratoire d'Informatique de Paris 6



CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UPMC  
UNIVERSITÉ PARIS UNIVERSITAS

## Mécanismes « incitatifs »

- Anti-snubbing
  - Un pair peut être « chocked » par tous les autres
  - Pose problème tant qu'il ne trouve pas de voisin avec l' « Optimistic Unchoking »
  - Après 60s sans téléchargement avec un voisin il peut le mettre en « opportunistic » de manière forcée.



Laboratoire d'Informatique de Paris 6



CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UPMC  
UNIVERSITÉ PARIS UNIVERSITAS

## Les approches DHT

### Overlays structurés



Laboratoire d'Informatique de Paris 6



CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UPMC  
UNIVERSITÉ PARIS UNIVERSITAS

## Deuxième génération

- Problème du non structuré :
  - Pas de réponse garantie ou à un coût élevé (inondation complète du réseau)
- Objectifs de base :
  - Éviter l'inondation pour la recherche.
  - Garantir une réponse avec un nombre borné de sauts.
  - Fournir une réponse complète.



Laboratoire d'Informatique de Paris 6



CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UPMC  
UNIVERSITÉ PARIS UNIVERSITAS

## Caractéristiques idéales

- Coopération
- Arrivées et départs simples (mise à jour rapide, peu d'overhead)
- Tolérant aux fautes (départ brusque)
- Anonyme
- Sécurité
- Auto-organisation
- Trafic bien réparti

## Distributed Hash Tables

- Idée globale :
  - Un identifiant est associé à chaque objet
  - Exemple : ID = hachage du contenu
- Une opération de base, la recherche :
  - Lookup(ID)
  - Input : un entier ID
  - Output : un pair avec des infos sur ID

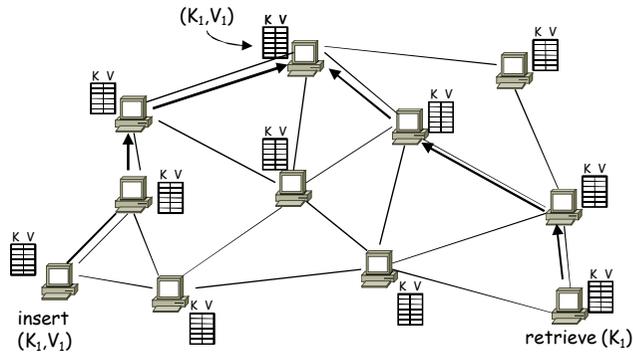
## DHTs

- Le hachage idéal
  - $x$  = nom d'une chanson
  - $H(x)$  = @IP/port d'une machine qui stocke la chanson
  - Impossible en pratique
- Les IDs sont stockés par les pairs :
  - Chaque pair est responsable d'un certain nombre d'IDs
  - Hachage permet de trouver le pair ayant les informations

## DHTs

- Le réseau structuré :
  - Noeuds = machines/utilisateurs
  - Lien (A,B) si A connaît l'@IP de B dans sa table de routage
  - Donnée  $d$   $h(d)$  = clef
  - Chaque machine a une table d'association des clefs vers les @IP des fournisseurs des données pour les clefs sous sa responsabilité

## Idée générale



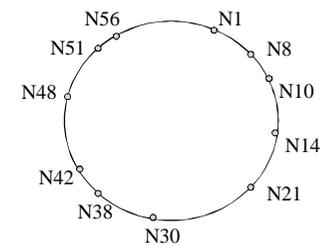
## Exemples de DHT Chord - CAN

## Chord - Protocole

- Fonction de hachage : la clef est calculée avec SHA-1 sur  $m$  bits.
- Les @IP sont hachées de la même manière.
- Les identifiants sont ordonnés sur un cercle anneau.

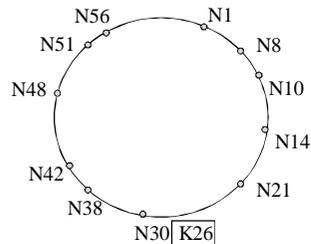
## Chord - Structure logique

- Espace d'adressage  $m$ -bit ( $2^m$  IDs)  
– typiquement  $m=160$
- Nœuds organisés sur un anneau



## Chord - Assignment des clefs

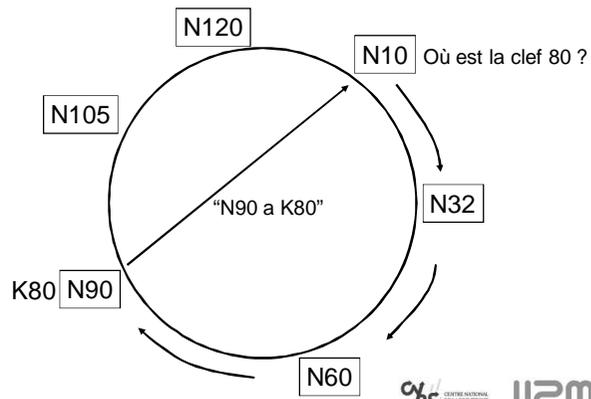
- Clefs : SHA-1 du contenu
- ID des nœuds : SHA-1 de l'@IP
- Clef k donnée au premier nœud d'ID  $\geq k$



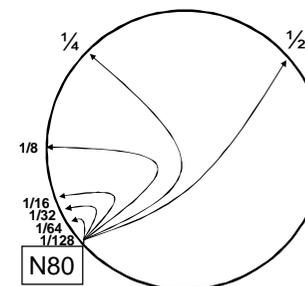
## Chord - Routage sur l'anneau

- Objectif : qui possède la clef k ?
- Plusieurs approches :
  - Un nœud ne connaît que son successeur :
    - Routage sur l'anneau
    - Routage en  $O(N)$
    - Complexité spatiale en  $O(1)$ , successeur
  - Chaque nœud connaît tous les autres :
    - Routage en  $O(1)$
    - Complexité spatiale en  $O(N)$
    - Coût supplémentaire ?
  - Autres approches ?

## Chord - Routage sur l'anneau

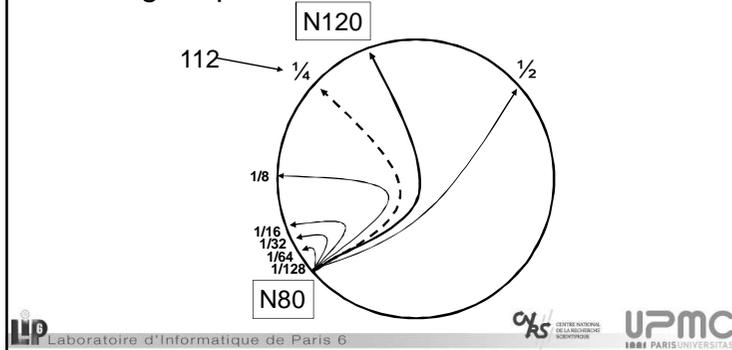


## Chord - « Finger table »



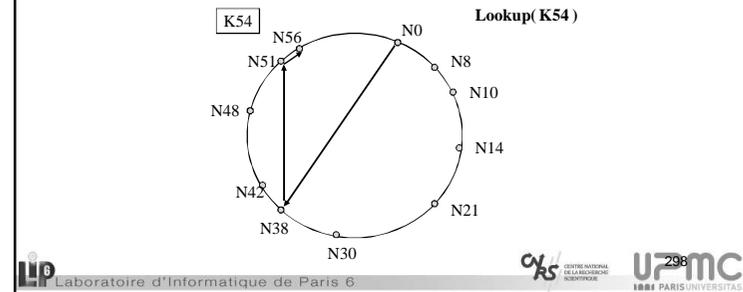
## Chord - « Finger table »

- Finger  $i$  pointe vers  $n+2^i$  :  $m$  voisins



## Chord - Routage des requêtes

- La requête pour  $k$  est transférée au finger maximal plus petit que  $k$
- $m$  étapes maximum



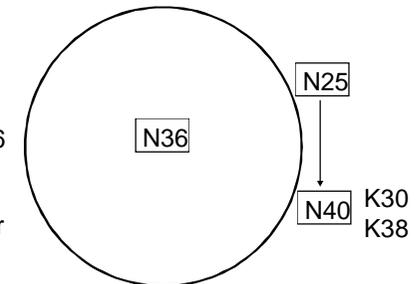
## Chord - Routage

- Chaque nœud transmet une requête sur au moins la moitié de la distance le séparant de la cible.
- Théorème : avec forte probabilité, le nombre de nœuds avant d'arriver à destination est  $O(\log N)$



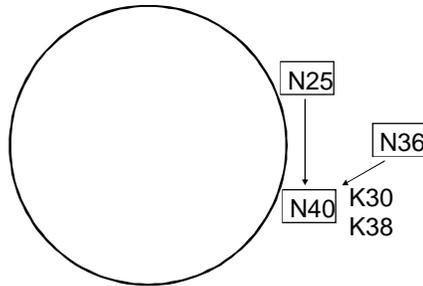
## Chord - Arrivée

- Recherche(36)
- $N36$  affecte son successeur
- Les clefs de 26 à 36 sont attribuées à  $N36$
- $N25$  affecte son successeur
- Les tables finger sont mises à jour plus tard (complexe)



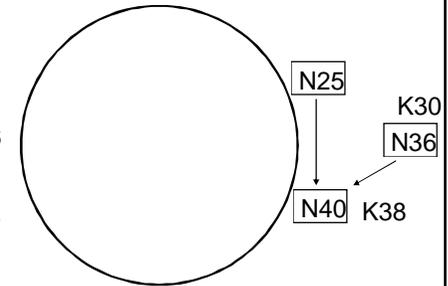
## Chord - Arrivée

1. Recherche(36)
2. N36 affecte son successeur
3. Les clefs de 26 à 36 sont attribuées à N36
4. N25 affecte son successeur
5. Les tables finger sont mises à jour plus tard (complexe)



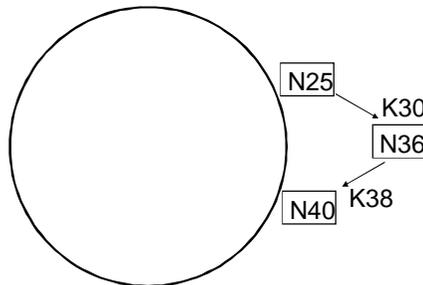
## Chord - Arrivée

1. Recherche(36)
2. N36 affecte son successeur
3. Les clefs de 26 à 36 sont attribuées à N36
4. N25 affecte son successeur
5. Les tables finger sont mises à jour plus tard (complexe)



## Chord - Arrivée

1. Recherche(36)
2. N36 affecte son successeur
3. Les clefs de 26 à 36 sont attribuées à N36
4. N25 affecte son successeur
5. Les tables finger sont mises à jour plus tard (complexe)



## Chord - Arrivées

- Recherche
  - Si les tables finger sont à jour (ou presque) :  $O(\log N)$
  - Successeur ok, tables finger erronées : recherche plus lente
  - Successeur incorrect : pas de garantie
- Performance :
  - Si le système est stabilisé :  $O(\log N)$
  - Sinon
    - Certaines tables finger peuvent être incomplètes (impact faible)
    - De nouveaux nœuds peuvent se trouver entre la destination et son prédécesseur
      - La recherche doit être propagée par les nouveaux nœuds un par un

## Chord - Pannes/départs

- Le protocole fonctionne si chacun connaît son successeur
- Pour rendre le protocole plus sûr :
  - Chacun connaît plusieurs successeurs
  - Stabilisation différente
- Départs volontaires :
  - Peuvent être traités comme une panne
  - Le nœud peut transférer ses clefs à son successeur et informer son prédécesseur et son successeur de leurs existences réciproques
- Nœuds malicieux ?



Laboratoire d'Informatique de Paris 6



CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UNIVERSITÉ  
PARIS  
UNIVERSITAS

## Deuxième exemple : CAN

- Content Adressable Network
- Un modèle simple et intuitif
  - Espace euclidien
- Mais pas le plus déployé
  - Tapestry et Pastry ont plusieurs applications basées dessus :
    - Stockage distribué, filtre spam distribué, ...



Laboratoire d'Informatique de Paris 6



CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UNIVERSITÉ  
PARIS  
UNIVERSITAS

## CAN : solution

- Système virtuel de coordonnées cartésiennes
  - Espace des clefs
- Espace partitionnée parmi tous les nœuds
  - chaque nœud « possède » un morceau d'espace
- Pour chaque point de l'espace :
  - Stockage de données en ce point
  - Routage d'un point à un autre
  - Fonctions assurées par le nœud auquel appartient le point de l'espace



Laboratoire d'Informatique de Paris 6

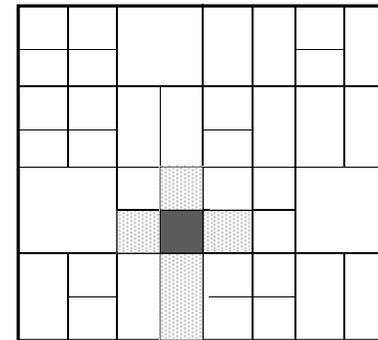


CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UNIVERSITÉ  
PARIS  
UNIVERSITAS

## CAN - routage



Laboratoire d'Informatique de Paris 6

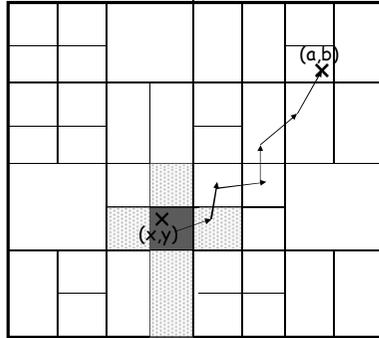


CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UNIVERSITÉ  
PARIS  
UNIVERSITAS

## CAN - routage



Laboratoire d'Informatique de Paris 6



CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UNIVERSITÉ PARIS  
UNIVERSITAS

## CAN - routage

- L'état d'un nœud
  - Ses propres données
  - Information à propos de ses voisins
  - rien d'autre



Laboratoire d'Informatique de Paris 6



CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE

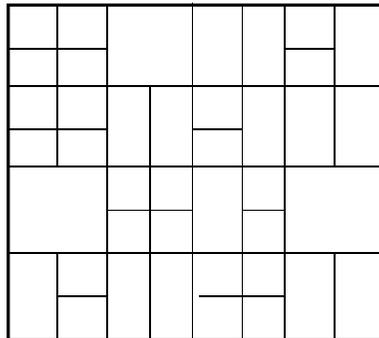


UNIVERSITÉ PARIS  
UNIVERSITAS

## CAN - insertion d'un nœud

- 1) Trouver un nœud "I" déjà dans le CAN
- 2) Choisir une position aléatoire
- 3) I route vers (p,q), et découvre J
- 4) L'espace de J est divisé en 2

●  
nouveau  
nœud



Laboratoire d'Informatique de Paris 6



CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE

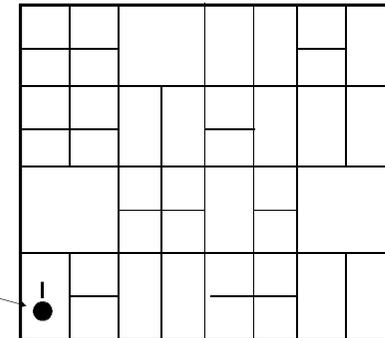


UNIVERSITÉ PARIS  
UNIVERSITAS

## CAN - insertion d'un nœud

- 1) Trouver un nœud "I" déjà dans le CAN
- 2) Choisir une position aléatoire
- 3) I route vers (p,q), et découvre J
- 4) L'espace de J est divisé en 2

●  
nouveau  
nœud



Laboratoire d'Informatique de Paris 6



CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE

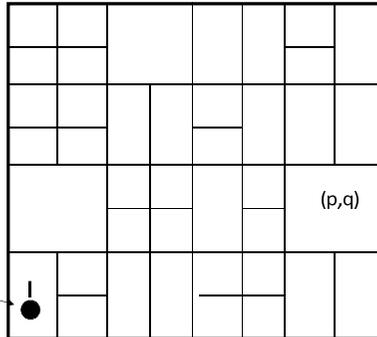


UNIVERSITÉ PARIS  
UNIVERSITAS

## CAN - insertion d'un nœud

- 1) Trouver un nœud "I" déjà dans le CAN
- 2) Choisir une position aléatoire
- 3) I route vers (p,q), et découvre J
- 4) L'espace de J est divisé en 2

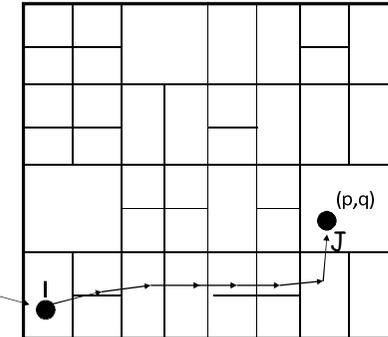
nouveau nœud



## CAN - insertion d'un nœud

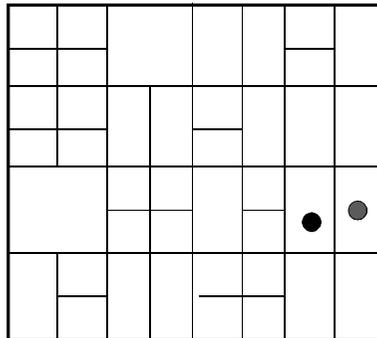
- 1) Trouver un nœud "I" déjà dans le CAN
- 2) Choisir une position aléatoire
- 3) I route vers (p,q), et découvre J
- 4) L'espace de J est divisé en 2

nouveau nœud



## CAN - insertion d'un nœud

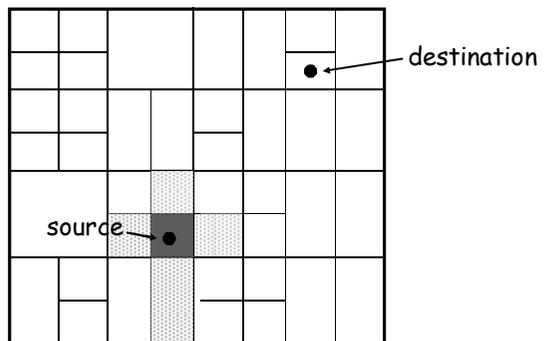
- 1) Trouver un nœud "I" déjà dans le CAN
- 2) Choisir une position aléatoire
- 3) I route vers (p,q), et découvre J
- 4) L'espace de J est divisé en 2



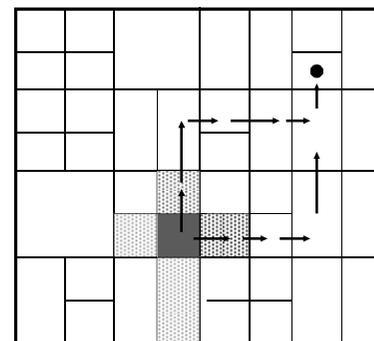
## CAN - Défaillance d'un nœud

- L'espace doit être réparée
  - les données du nœud défaillant
    - mises à jour périodiques
    - de la redondance dans les voisins
  - routage
    - les voisins assument la responsabilité
  - seuls les voisins sont impliqués

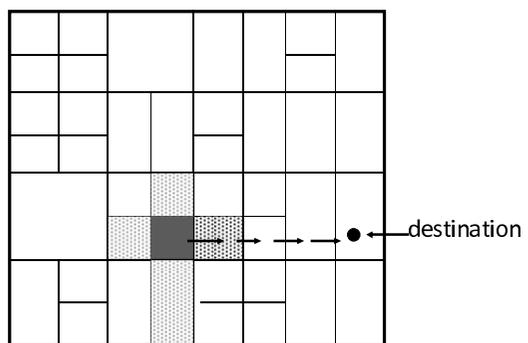
## CAN - Routage résistant



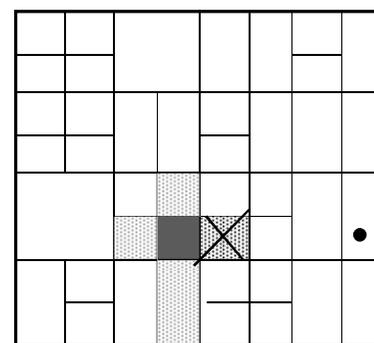
## CAN - Routage résistant



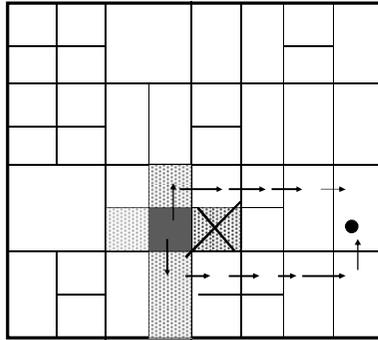
## CAN - Routage résistant



## CAN - Routage résistant



## CAN - Routage résistant



Laboratoire d'Informatique de Paris 6



CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UPMC  
UNIVERSITÉ PARIS  
UNIVERSITAS

## CAN - facteur d'échelle

- Soit
  - $n$  le nombre de nœuds dans le CAN
  - $d$  le nombre de dimensions de l'espace virtuel
- Voisins
  - le nombre par nœud est  $2d$
- Chemins entre les nœuds
  - $(dn1/d)/4$  sauts, en moyenne
- Tapestry et Chord
  - $\log(n)$  voisins par nœud
  - $\log(n)$  sauts sur un chemin



Laboratoire d'Informatique de Paris 6



CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UPMC  
UNIVERSITÉ PARIS  
UNIVERSITAS

## DHT - Conclusions

- Limiter les nombres de contacts
  - Peu d'informations à stocker sur chaque sommet
  - Mises à jour plus rapides
- Limiter le diamètre
  - Inondation rapide si besoin
  - Recherche en temps borné garanti
- Nombreuses approches récentes :
  - Optimisation de ces paramètres
  - Topologies sous-jacentes de plus en plus complexes.



Laboratoire d'Informatique de Paris 6



CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UPMC  
UNIVERSITÉ PARIS  
UNIVERSITAS

## DHT - Problèmes

- Localisation DHT Vs. Localisation réelle
  - A est le voisin Chord/CAN de B
  - A est à Paris, B à Tokyo
- Défaillance de plusieurs nœuds ?
- Concentrations de trafic
  - Nœuds responsables pour des fichiers fréquemment réclamés



Laboratoire d'Informatique de Paris 6



CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



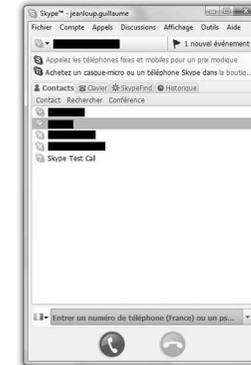
UPMC  
UNIVERSITÉ PARIS  
UNIVERSITAS

# Skype

## VoIP en P2P/DHT

# Skype

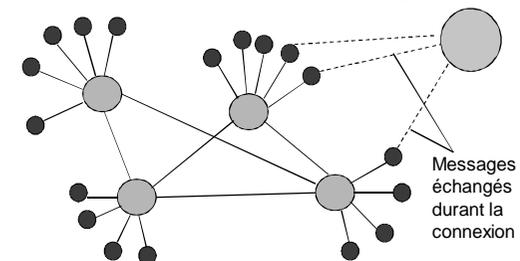
- Protocole pas complètement connu
  - Propriétaire
  - Contrôle et contenu crypté



# Skype Overlay

- Nœuds ordinaires (ON)
  - Connectés aux super nœuds
- Super Nœuds (SN)
  - Connectés entre eux
  - Overlay non structuré (similaire à Gnutella ?)
  - Agissent comme directory servers (similaire aux index serveur de Napster)
- Serveur de connexion Skype
  - Seul composant central
  - Stocke et vérifie login/pass
  - Stocke la liste de contacts

# Skype Overlay



## Construction de l'overlay

- Nœuds ordinaires :
  - Listes d'@ IP de SN
  - Jusqu'à 200 entrées
- Super nœuds :
  - Certaines @IP sont fixes
  - Fournis par Skype
- Serveur de login :
  - Utilisés à la connexion pour trouver des SN



Laboratoire d'Informatique de Paris 6



CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UNIVERSITÉ PARIS  
SUD  
PARIS  
UNIVERSITAS

## Construction de l'overlay

- Contact serveur de login
- Signaler sa présence :
  - Contacter un SN
  - Contacter ses voisins (à travers le SN) et leur signaler sa présence



Laboratoire d'Informatique de Paris 6



CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UNIVERSITÉ PARIS  
SUD  
PARIS  
UNIVERSITAS

## SN – Serveurs d'Index

- Super Nœuds = index
  - Indexation des utilisateurs connectés et de leur @IP
- Si un contact n'est pas trouvé dans l'index local du SN :
  - La recherche est diffusée aux super nœuds voisins
  - Implémentation pas complètement comprise
    - Inondation similaire à Gnutella



Laboratoire d'Informatique de Paris 6



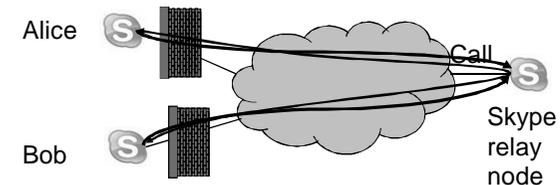
CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UNIVERSITÉ PARIS  
SUD  
PARIS  
UNIVERSITAS

## SN – Relai

- SN peuvent relayer des informations :
  - Traverser des serveurs NAT
  - Eviter des chemins congestionnés



Laboratoire d'Informatique de Paris 6



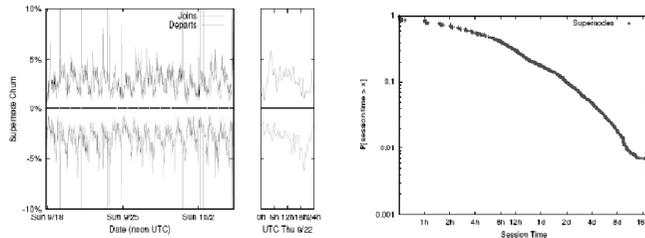
CENTRE NATIONAL  
DE LA RECHERCHE  
SCIENTIFIQUE



UNIVERSITÉ PARIS  
SUD  
PARIS  
UNIVERSITAS

## Devenir un SN

- Bande passante élevée, @IP publique
  - Détails pas clairs (on ne choisit pas)
  - Très variable ( churn, durée des sessions)



## Devenir un SN

- Carte des SN au niveau mondial



## Skype - Conclusion

- VoIP a d'autres prérequis que le simple transfert de fichiers :
  - Délai, gigue, etc.
- Skype semble gérer ces contraintes malgré le churn élevé
- Protocole pas entièrement compris

## Conclusions

## Clefs du succès

- Coût faible :
  - peu ou pas d'infrastructure
  - Serveurs de connexion
- Chacun peu introduire les contenus
  - Fait-maison, illégal, légal, ...
- Forte disponibilité des contenus :
  - accessibles presque en permanence
- Applications à succès :
  - Téléchargement de fichiers, VoIP, TV sur IP, VoD, ...



Laboratoire d'Informatique de Paris 6



Centre National  
de la Recherche  
Scientifique



Université Paris  
Montpellier

