



Transport de contenu multimédia

Jean Le Feuvre
Cyril Concolato

SI350

lefeuvre, concolato @telecom-paristech.fr



- **Définitions et problèmes généraux**
- **Étude détaillée de certains modes de transport**
 - Streaming sur IP
 - Broadcasting sur MPEG-2
 - Diffusion de fichiers



Définitions

■ Distribution de contenu multimédia

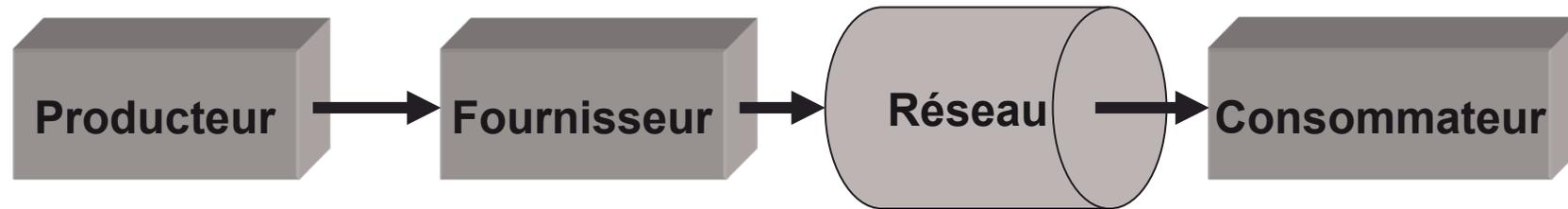
- Ensemble de moyens permettant d'acheminer du contenu jusqu'à une personne au moyen d'un réseau

■ Contenu Multimédia

- Contenu numérique audio-visuel comprenant plusieurs types de media et des informations de synchronisation, de présentation, d'interactivité
- Flux élémentaire = succession temporelle de données média d'un même type
 - Ex : Audio, Vidéo, Image fixe, Graphique 2D/3D, Texte, méta données



Les acteurs d'une chaîne de distribution



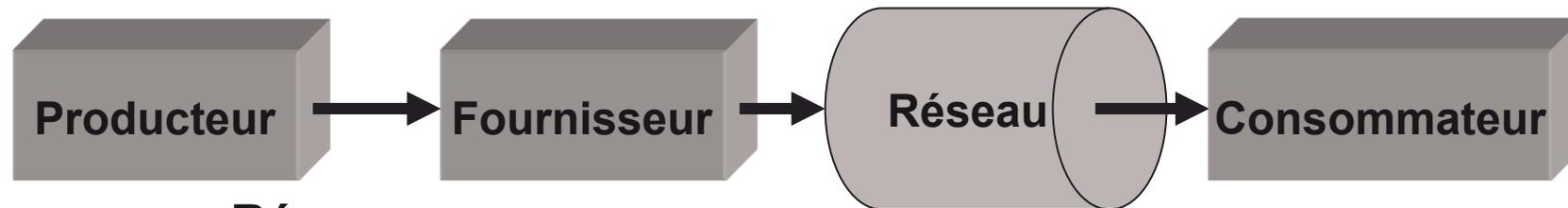
■ Producteur

- Rôle : Création, encodage, préparation à la distribution de contenu MM
- Ex : Professionnels, Personnels (« User Generated Content (UGC) »)

■ Fournisseur

- Rôle : Exploiter un réseau pour distribuer le contenu
- Peut également faire l'encodage et la préparation
- Ex : Professionnels (TDF, Orange), Particulier (Web Vidéo, Peer-to-Peer)

Les acteurs d'une chaîne de distribution (2)



■ Réseaux

- Rôle : Transporter le contenu avec une qualité de service donnée (débit, délais, taux d'erreur ...)
- Moyen physique de transport
 - Ex : Satellite, Hertzien, Cellulaire, filaire...
- Protocoles (MPEG-2, IP, DAB ...)

■ Consommateur

- Rôle : Recevoir et visualiser le contenu
- Nécessite un récepteur adapté au réseau
- Nécessite un lecteur adapté au contenu



Classification des modes de distribution

■ Distribution centralisée

- 1 fournisseur / N consommateurs, $N \gg 1$
- Distribution à la demande
 - Ressources systèmes importantes au niveau du fournisseur
 - Stockage, puissance de calcul, bande passante
- Distribution de masse « Broadcast », « Multicast »
 - Ressources systèmes mieux gérées
 - Infrastructure spécialisée

■ Distribution décentralisée

- 1 fournisseur / 1 consommateur
 - Répartition des ressources
 - Communication bidirectionnelle ou monodirectionnelle
- Exemples:
 - Réseaux P2P
 - Vision Conférence Full-duplex



Distribution centralisée à la demande

- **« On Demand »**
- **Caractéristiques**
 - Obtention du contenu sur requête
 - Utilisation des ressources réseau à partir de la requête
 - Mode connecté: une connexion par consommateur
 - Utilisation d'une voie de retour possible
 - Contenu adaptable au consommateur (profil utilisateur)
- **Exemples**
 - VOD: « Video on Demand »
 - Téléchargement de fichiers



Distribution centralisée de masse

- **« Broadcast »**
- **Caractéristiques**
 - Distribution simultanée du même contenu à tous les consommateurs
 - Utilisation permanente du réseau (ex: satellite)
 - Mode non-connecté: Le consommateur rejoint une session
- **Exemples**
 - Session « Multicast » sur réseau IP
 - Télé, Radio: Broadcast de programmes TV, Radio
 - « Near – Video on Demand »
 - rediffusion toutes les N minutes de P programmes en parallèle



Problématiques de distribution de contenu

■ Gestion du temps

- Les données multimédia sont à caractère temporel
- Respect des contraintes temps-réel, de la synchronisation intra-flux

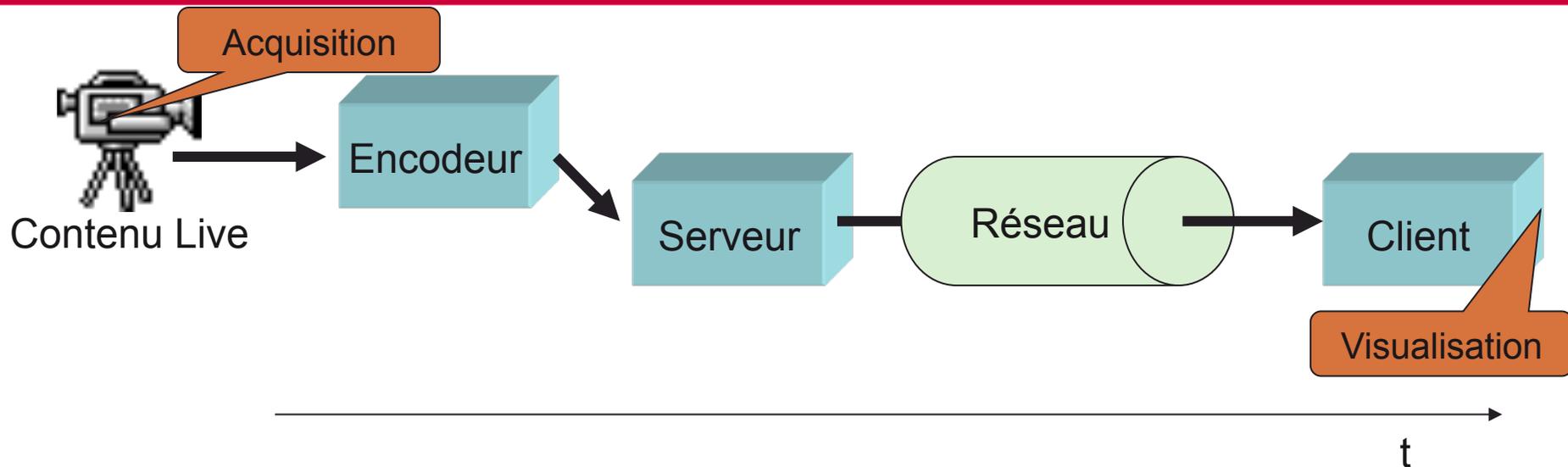
⇒ Besoin de protocoles adaptés (signalisation)

■ Gestion du nombre de flux

■ Gestion des contraintes réseaux

■ Gestion du mode de connexion

Distribution Temps Réel



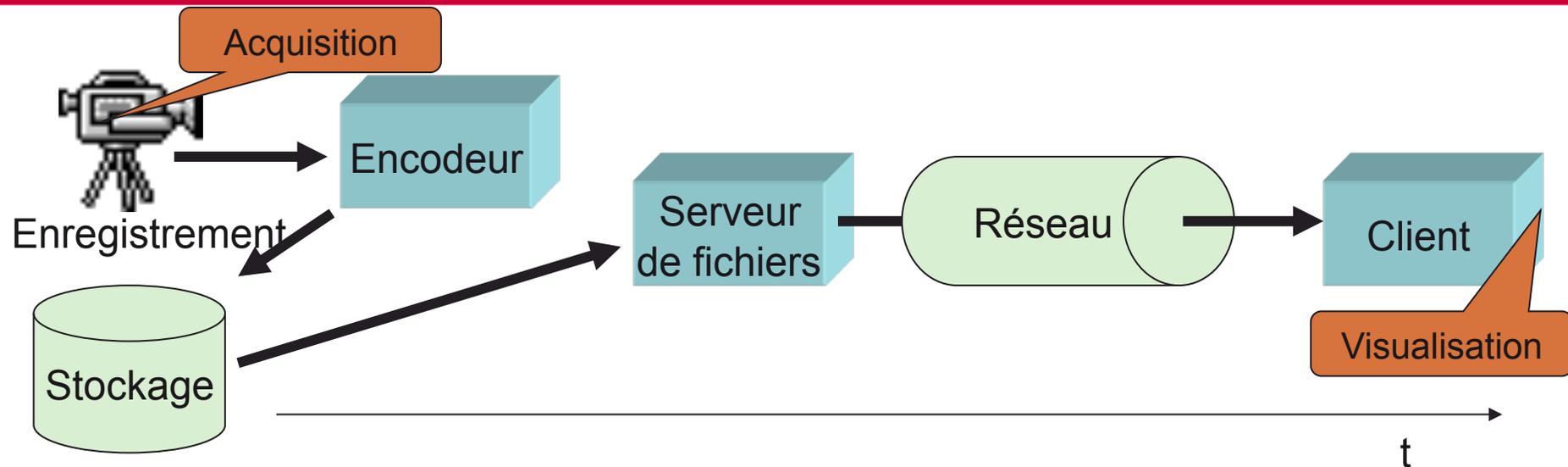
■ Contrainte

- Respect d'un délai constant entre l'acquisition d'un média et sa visualisation

■ Problèmes

- Nombreux délais: Encodage, Serveur, **Réseau**, Décodage
- Gigue (« jitter »): Variation du délai de bout en bout

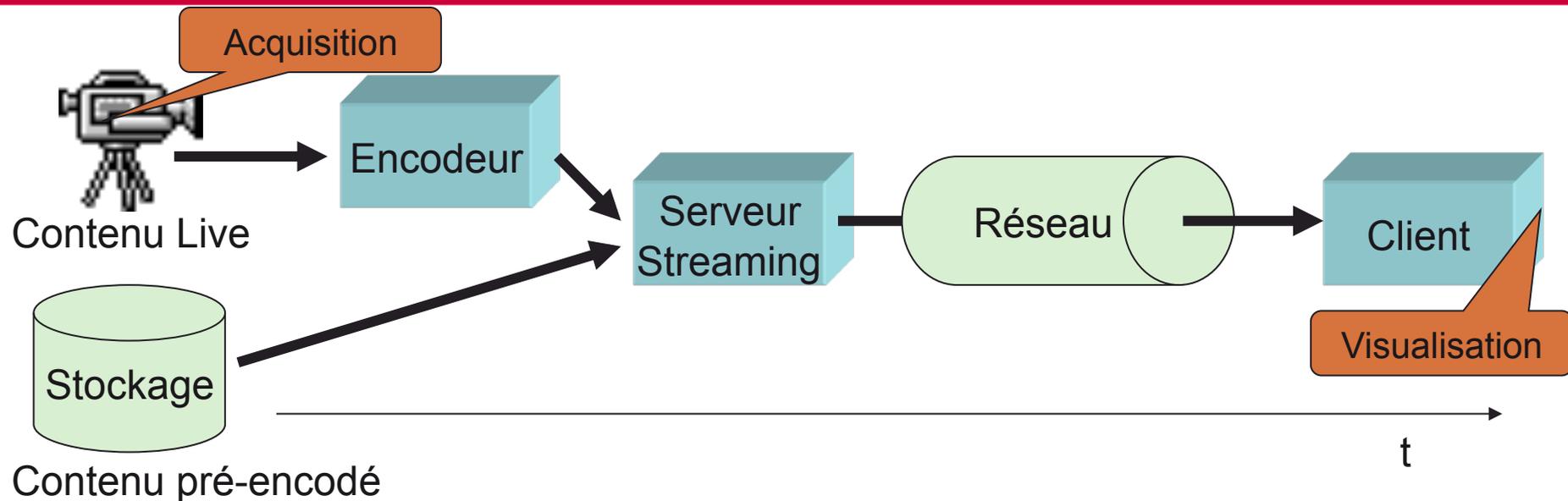
Envoi de fichier



■ Mode « Download & Play »

- Enregistrement complet dans un fichier puis envoi du contenu
- Insensible à la variation des délais
- Très sensible au débit du réseau et à la taille du contenu (délai initial)

Envoi Temps Réel de flux



■ Mode « Live » ou « Pré-encodé »

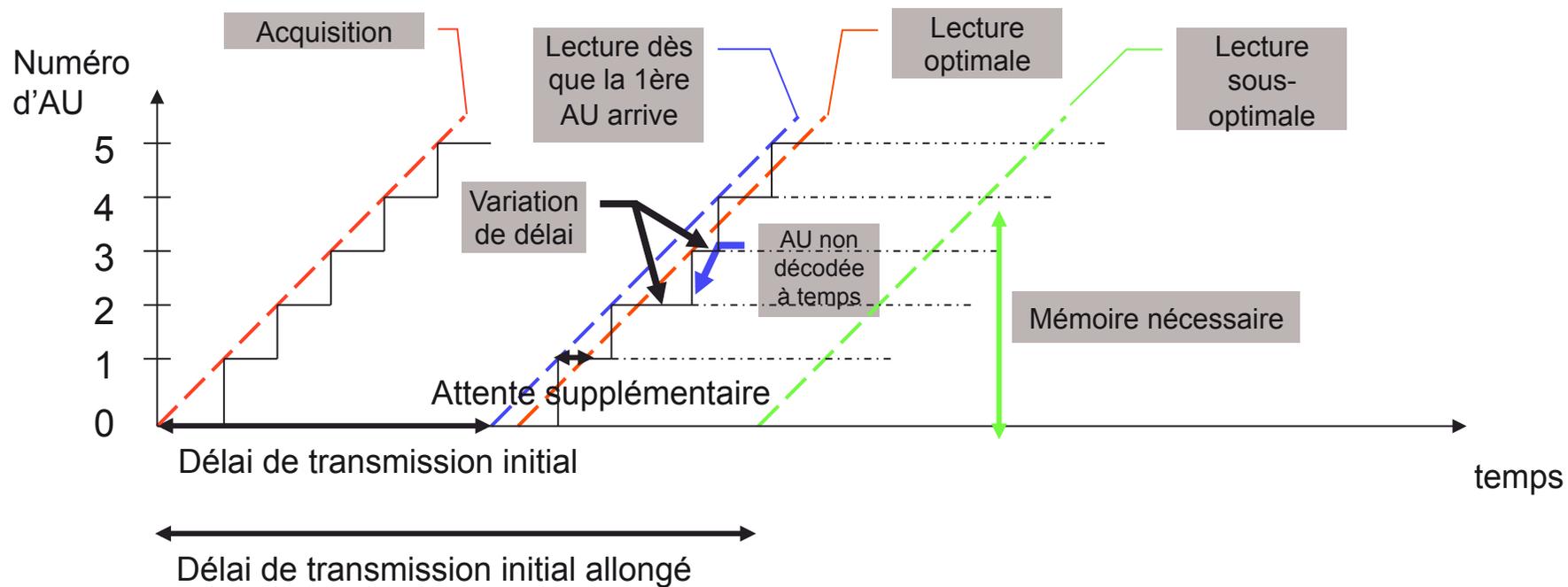
- Envois à intervalle régulier de fragments
 - Au fil de l'acquisition
 - Au fil de la lecture dans le fichier
- Respect des échéances de chaque fragment
 - Sensibilité aux variations du délai



Quelles échéances ?

- **Decoding Time Stamp (DTS)**
 - Instant indiquant quand la donnée doit être décodée
- **Presentation/Composition Time Stamp (PTS / CTS)**
 - Instant à partir duquel la donnée peut être présentée
- **Relation DTS / CTS**
 - $CTS = DTS$ la plupart du temps
 - $CTS \neq DTS$ pour les codages bidirectionnels
 - Exemple: Images I, P et B

Contraintes Temps Réel





Contraintes Temps Réel

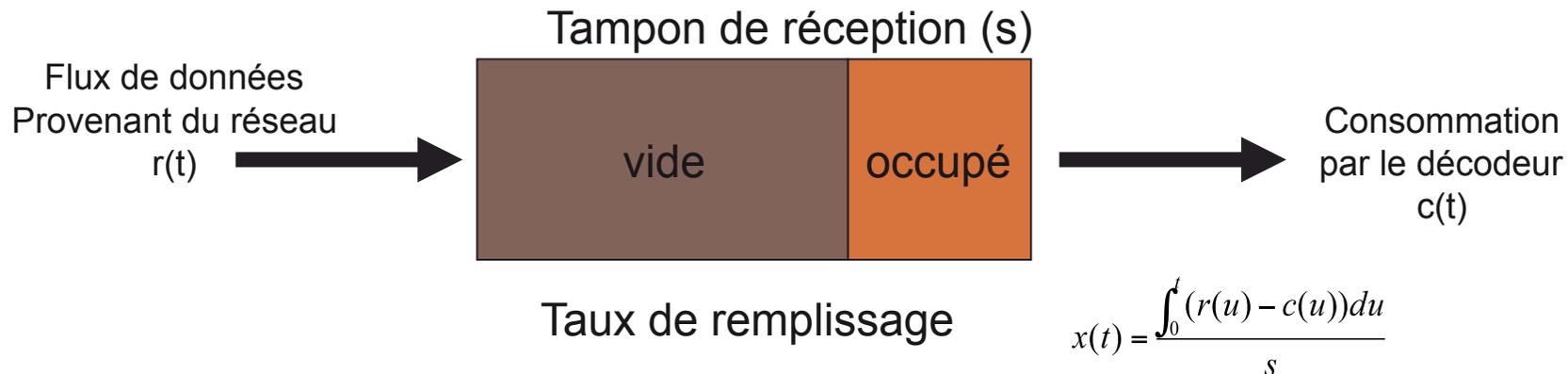
■ Délai de chargement initial

- Dépendant de l'application
 - Ex: téléphonie (duplex)
 - Délai idéal < 150 ms
 - Délai tolérable < 400 ms
 - Délai intolérable > 400 ms
 - Ex: vidéo
 - Délai tolérable de quelques secondes
- Lié à la capacité de stockage disponible dans le tampon de réception

■ Variation du délai

- Peut entraîner le non respect des échéances d'AU
- Nécessité d'un tampon de réception (pre-buffering)
 - Prébuffering trop court => sensibilité par rapport à la gigue
 - Prébuffering long => surconsommation de ressources (terminal, réseau), gêne pour l'utilisateur

Gestion des tampons de réception



■ Contraintes à respecter

- $x(t) \geq 0$
 - $x(t) = 0$ pose problème si le décodeur attend des données
 - Famine ou « buffer underflow »
 - $x(t) > 0$ laisse une marge de manœuvre pour palier au problème de gigue
- $x(t) \leq 1$
 - $x(t) = 1$ pose problème si d'autres données arrivent
 - Débordement ou « buffer overflow »
 - Surconsommation du réseau
 - $x(t) < 1$



Récapitulatif

- **Nécessité de l'utilisation de tampon de réception**
 - Problème de délai initial
 - Problème de gigue
 - Adaptation aux variations de débit
 - Attente des paquets dans le tampon
 - Problème de changement d'ordre dans les paquets
 - Pas de garantie d'arrivée dans l'ordre
 - Nécessité de numérotter les paquets
 - Attente des paquets dans le tampon



Fragmentation et Aggregation

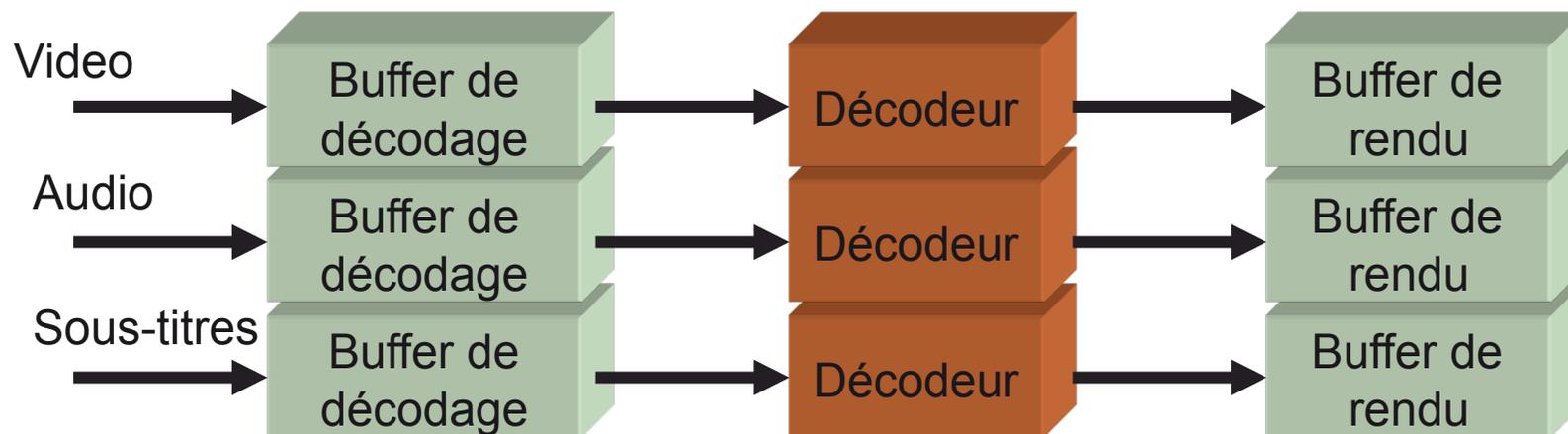
■ Fragmentation

- Nécessaire quand les AU média sont plus grosses que les paquets réseaux

■ Aggrégation

- Utile pour économiser de la bande passante quand les paquets réseaux sont plus gros que les AU média
- On groupe plusieurs AU dans un même paquet

Distribution de plusieurs flux



■ Problèmes individuels à chaque flux

■ Problèmes liés aux différences entre flux

- Echelle de temps différentes
 - Ex: Video = 25 images/s (1 image/40ms), Audio = 1 échantillon/20 ms, Contenus non périodiques (données textuelles ou graphiques)
- Débits différents (variable et/ou constant)
 - Ex: Video 20 Kbps -> 8 Mbps, Audio 8 kbps -> 500 kbps
- Tolérances aux délais différentes
- Tolérances aux pertes/corruptions différentes
 - Mécanismes de protection/correction différents



Distribution de plusieurs flux (2)

■ Problèmes inter-flux

- Respect des contraintes de synchronisation entre flux
 - Définition d'une **horloge** commune à tous les flux
 - Exemple: horloge absolue (NTP) et horloge relative au début de la session (envoi du temps d'horloge par le serveur)
- Priorités des flux les uns par rapport aux autres
 - Ex: Contenu A/V
 - Perte d'une image vidéo tolérable
 - Perte du son trop longue non acceptable
 - Ex: Présentation Transparents + Audio
 - Perte d'un transparent non acceptable

■ Besoin de signalisation

- Un protocole de distribution de contenu multimédia doit fournir un minimum de signalisation



Signalisation Multimédia

■ But

- Indiquer les informations nécessaires à la diffusion temps-réel d'un ou plusieurs flux en parallèle

■ Informations à transmettre

- Par flux
 - Type de flux
 - Mécanisme de protection aux erreurs
 - Base temporelle pour exprimer les CTS/DTS
 - Dépendances d'horloge
 - Priorités
- Par paquet de données
 - Informations de fragmentation (début, fin d'AU)
 - Permet de savoir si une AU a été reçue en entier ou pas
 - DTS/CTS: respect des échéances de temps de chaque AU
 - Information d'accès aléatoire (RAP)
 - Information de détection des pertes (numéro de paquet ou décalage dans les données)
 - Permet de mettre en place des mécanismes correctifs

■ Contrainte du mécanisme de signalisation

- Complet: toutes les infos nécessaires par rapport à la qualité de service du réseau
 - Efficace : utilisation optimale de la bande passante
 - Ex: échantillons audio = quelques octets
 - Ex: téléphone mobile GPRS: MTU = 100 octets, Paquet RTP = 12 octets de headers => 12 %
- => Utilisation de techniques de multiplexage



Problématiques de distribution de contenu

- Gestion du temps
- **Gestion du nombre de flux**
 - Les données multimédia sont réparties en plusieurs flux parallèles
 - Efficacité de la distribution
 - Conservation de la synchronisation inter-flux
- ⇒ Besoin de techniques de multiplexage
- Gestion des contraintes réseaux
- Gestion du mode de connexion



Le multiplexage

■ Buts

- Maitriser la variation de délai entre flux
- Réduire le nombre de connexions réseau
- Factoriser les informations de signalisation

■ Principes

- Envoi groupé sur un même canal de données issues de plusieurs flux

■ Types de multiplexage

- Temporel vs. Applicatif
- Débit constant vs. Débit variable
 - Multiplexage statistique



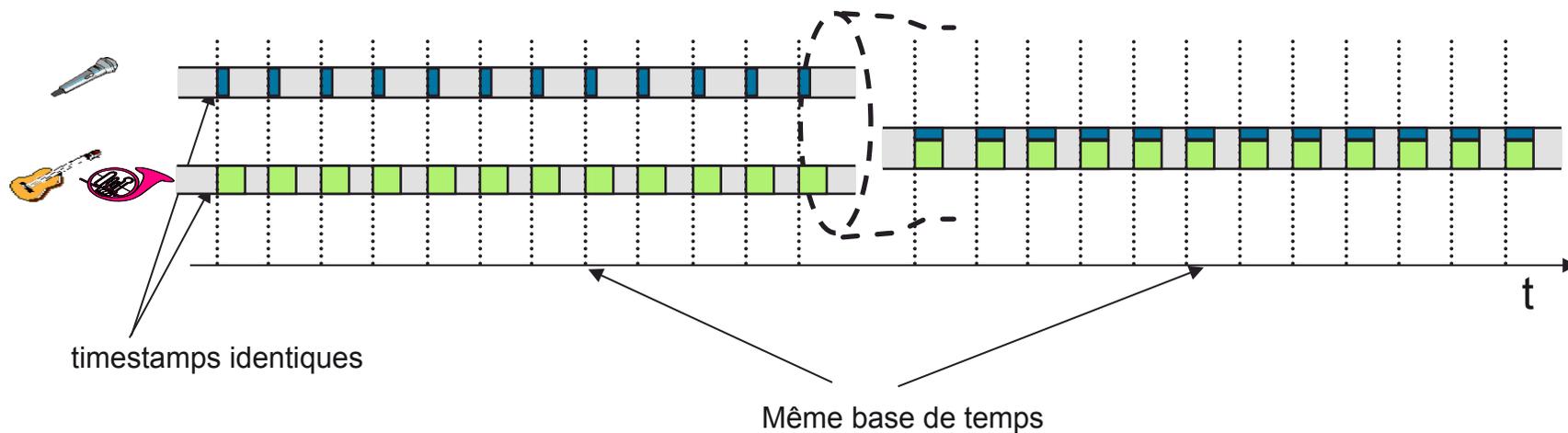
Multiplexage (1)

■ 2 flux audio:

- même échelle temporelle : 1 AU = 20 ms
- débits constants : ex: 20 kbps et 60 kbps

■ Résultat

- Débit constant
- Base de temps inchangée





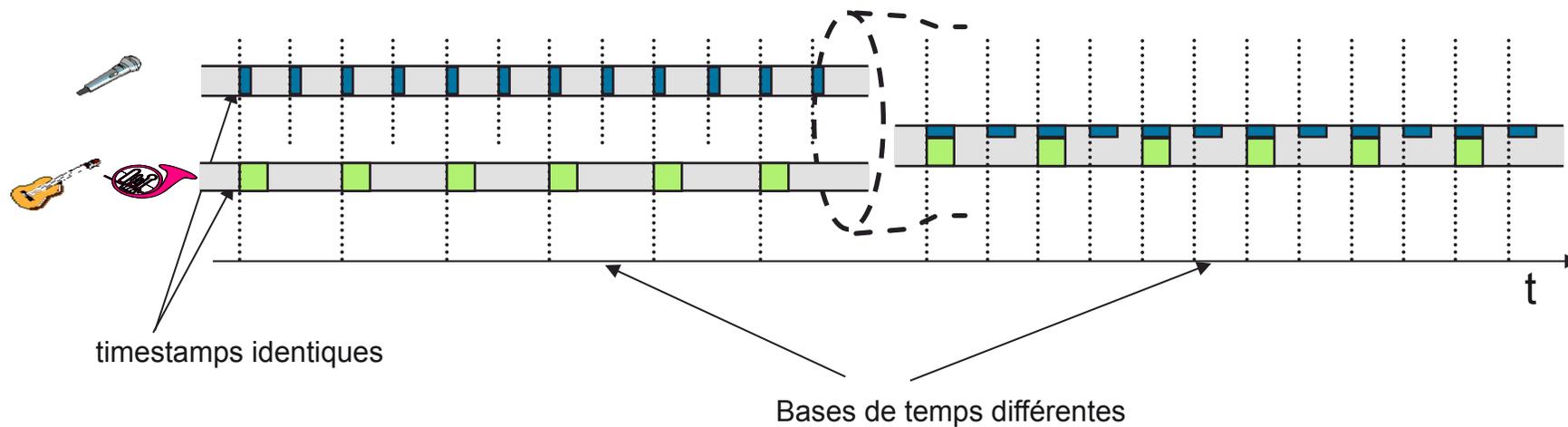
Multiplexage (2)

- **2 flux audio:**
 - échelles temporelles différentes
 - 1 AU = 20ms
 - 1 AU = 40ms
 - Débits constants : ex: 20 kbps et 60 kbps
- **2 possibilités**
 - Débit variable vs débit constant
 - Echelle de temps vs. Délai



Multiplexage (3)

- Débit variable
- Fréquence la plus rapide
 - Délai d'attente court
 - Signalisation peu efficace

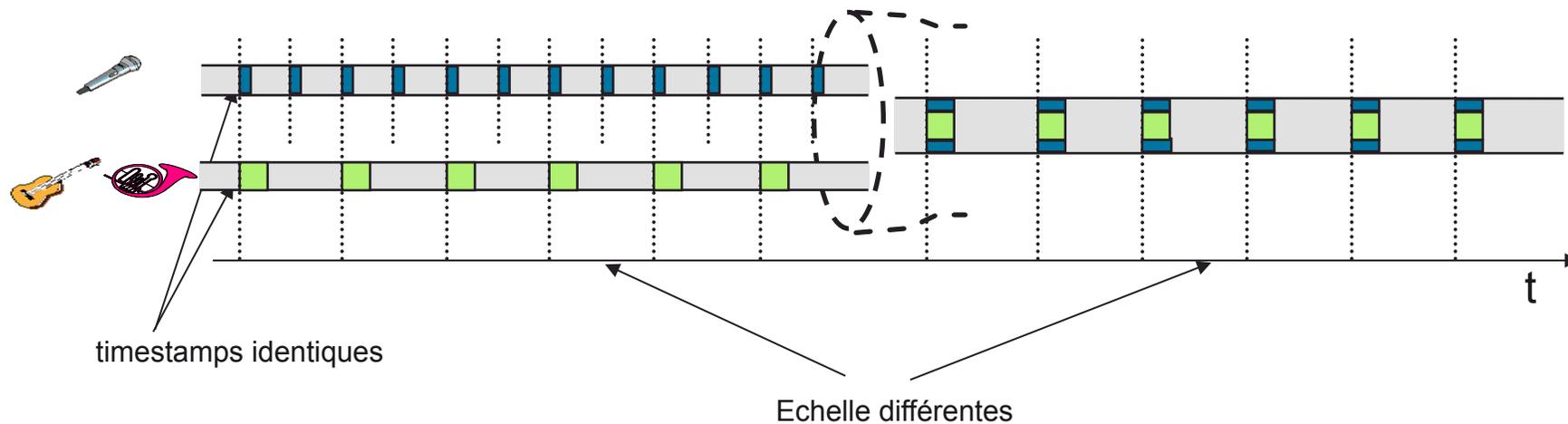




Multiplexage (4)

■ Utilisation d'une fenêtre de multiplexage

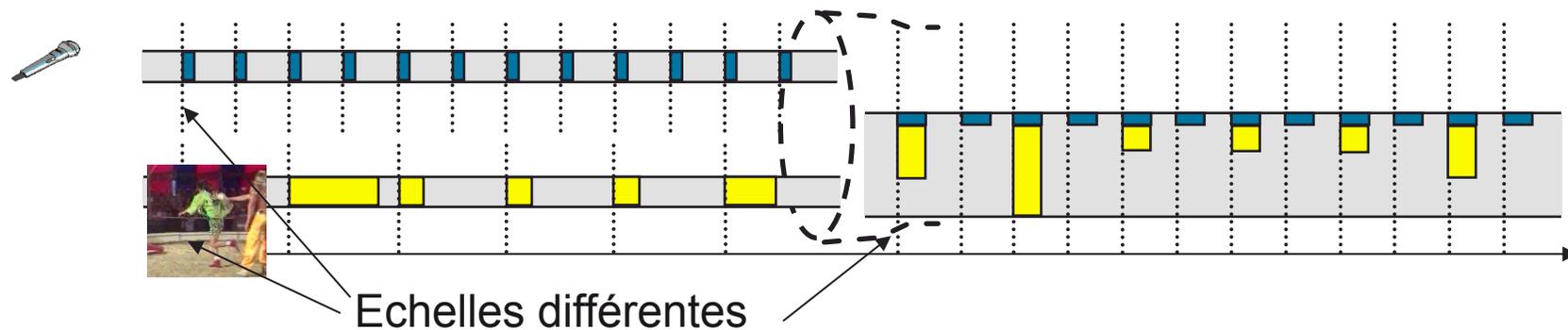
- Débit constant
- Nombre de paquets réduit (meilleure efficacité)
- Introduction d'un délai supplémentaire





Multiplexage de sources à débit variable

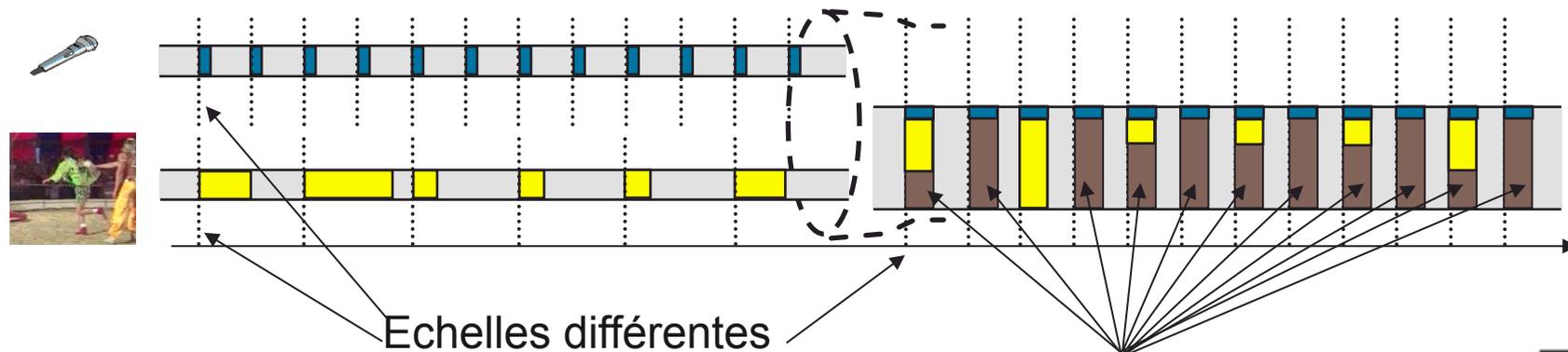
- Résultat: Multiplexe à débit variable
 - Complexité démodulation, démultiplexage
 - + Utilisation optimale de la bande passante
 - + Réactivité plus grande à l'ajout d'un nouveau flux





Multiplexage de sources à débit constant

- Résultat: Multiplexe à débit constant
 - + Facilité pour la modulation du canal
 - + Facilité de démultiplexage (parallélisme)
 - Perte de bande passante (bourrage)
 - Ajout de nouveau flux compliqué
- Ex: 1 Flux Audio + 1 Flux Vidéo (débits variables)
- Utilisé sur les réseaux à commutation de circuit, réseaux satellitaires
- Ex: MPEG-2 TS





Problématiques de distribution de contenu

- Gestion du temps
- Gestion du nombre de flux
- **Gestion des contraintes réseaux**
 - Les données multimédia peuvent être tolérantes aux pertes ou aux erreurs
 - ⇒ Besoin de techniques de formation des paquets, de récupération d'erreurs
- Gestion du mode de connexion

Environnements de diffusion non fiables

■ 3 types de problèmes

- Perte de données
 - Ex: Réseaux UDP
 - Cause :
 - problèmes de congestion du réseau
 - Saturations des buffers des routeurs
- Corruption aléatoire de bits de données
 - Se traduit le plus souvent par de la perte de paquet
 - Cause :
 - Chemins multiples qui créent des interférences (réseaux sans fils)
- Perte d'ordre dans les données
 - Chemins différents utilisés par des paquets consécutifs

■ Conséquences

- Impact sur les flux audio-visuels 'purs'
 - Ex: Flux Audio/Vidéo => perte d'une trame ou d'un échantillon audio
 - Ex: Image fixe => corruption d'une valeur de chrominance
- Impact important sur les flux à redondance faible
 - Ex: flux de description de scène => perte de la structuration de la scène
 - Ex: Flux de méta données MPEG-7 => perte d'un descripteur guide de programme

Problème de perte de données – Solutions techniques liées au réseau

■ Retransmission des données

- Totale: Protocole fiable type TCP
- Sélective: RTP + RTCP + politique de retransmission
- Problèmes
 - Risque de congestion des réseaux
 - Risque de famine au niveau du client « buffer underflow »
 - Possible perte de la synchronisation entre flux (Audio/Vidéo)

■ Ajout de redondance

- « Forward Error Correction »
- Problème : Augmentation du débit
 - Compromis entre quantité d'information redondante et la capacité à récupérer les données perdues ou corrompues

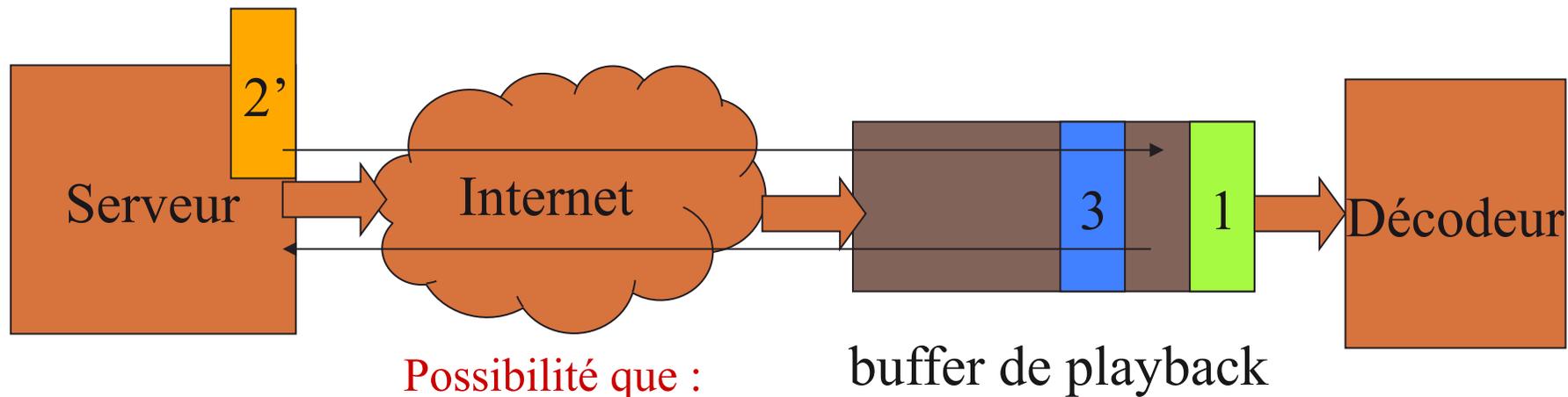
■ Règles de formation des paquets de données

- Limiter l'impact d'une perte ou d'une erreur
- Règles de bonne paquetisation

Retransmissions sélectives

■ Retransmissions automatiques (tel ARQ dans TCP): délais importants

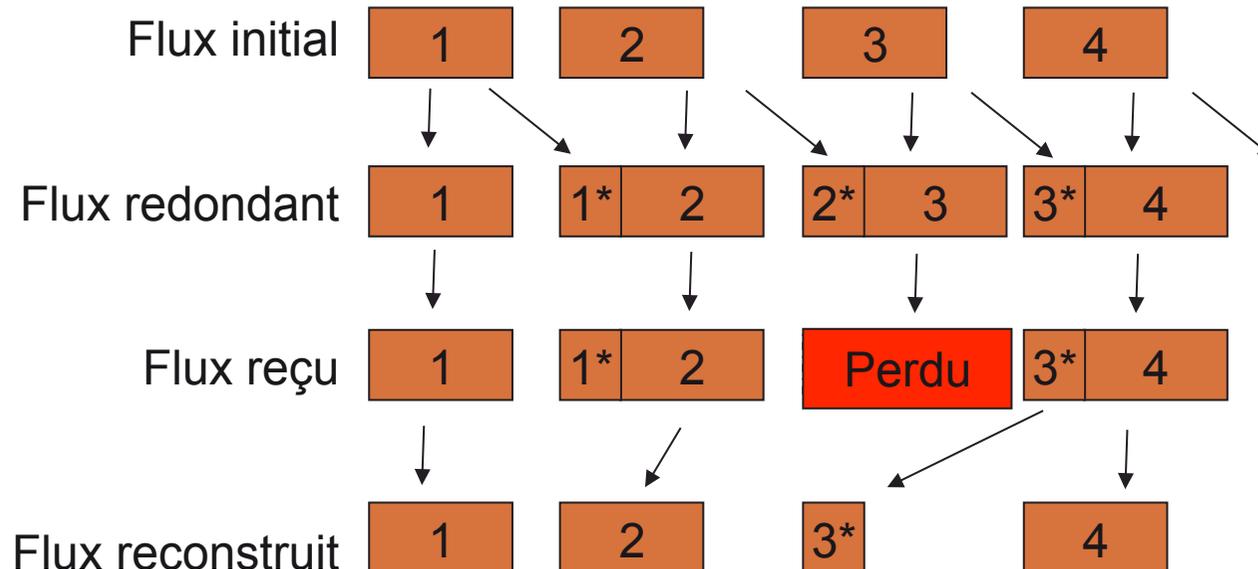
- délais total =
temps de détection de la perte de paquet
+ temps de réception par le serveur de la demande de retransmission
+ temps de transmission du nouveau packet



Possibilité que :
délais de retransmission
> temps avant échéance du paquet



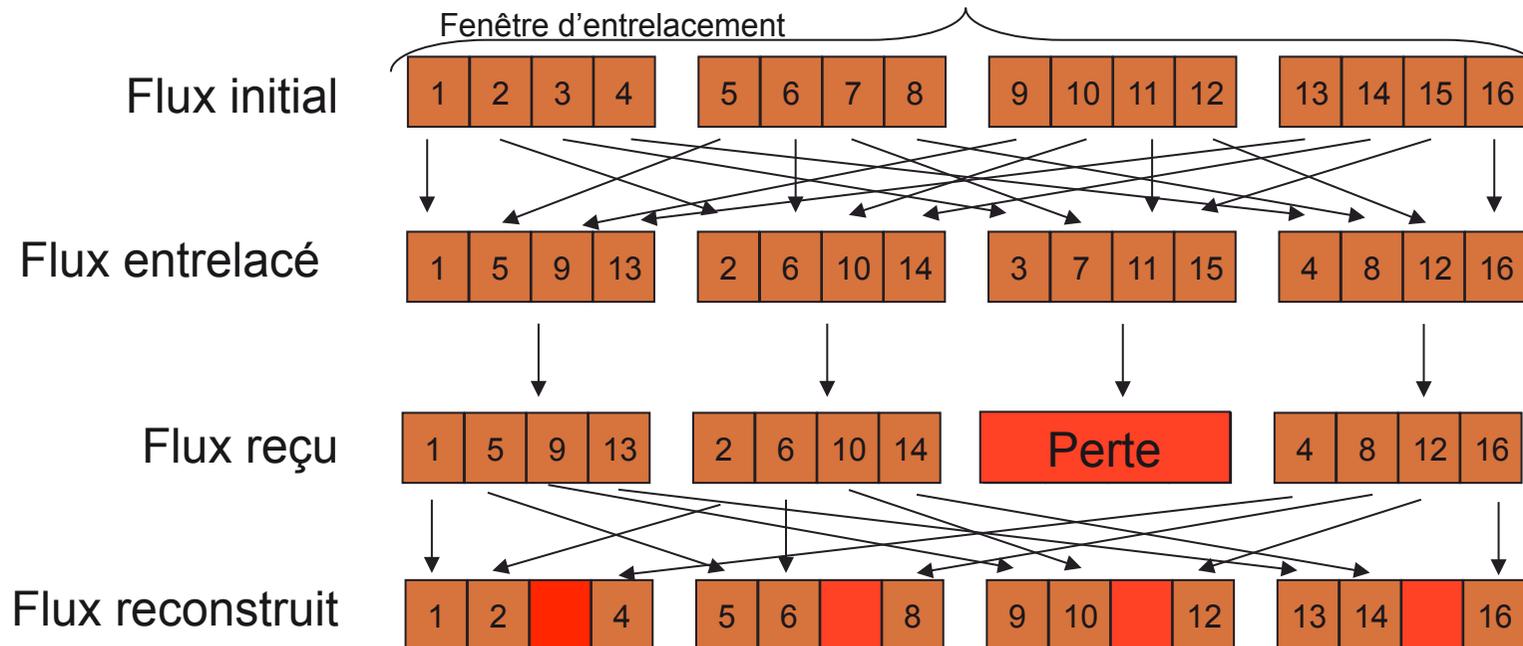
Exemple de FEC simple



- **Ajout de redondance**
 - 2 types: codes correcteurs d'erreur ou envoi d'information en double
 - Compromis taille des données ajoutées vs capacité à reconstruire
- **Exemple d'utilisation**
 - Codes Correcteurs: Reed-Solomon (télévision numérique)
 - Information en double:
 - Flux scalable: renvoi de la couche de base
 - Flux ré-encodé avec une qualité réduite

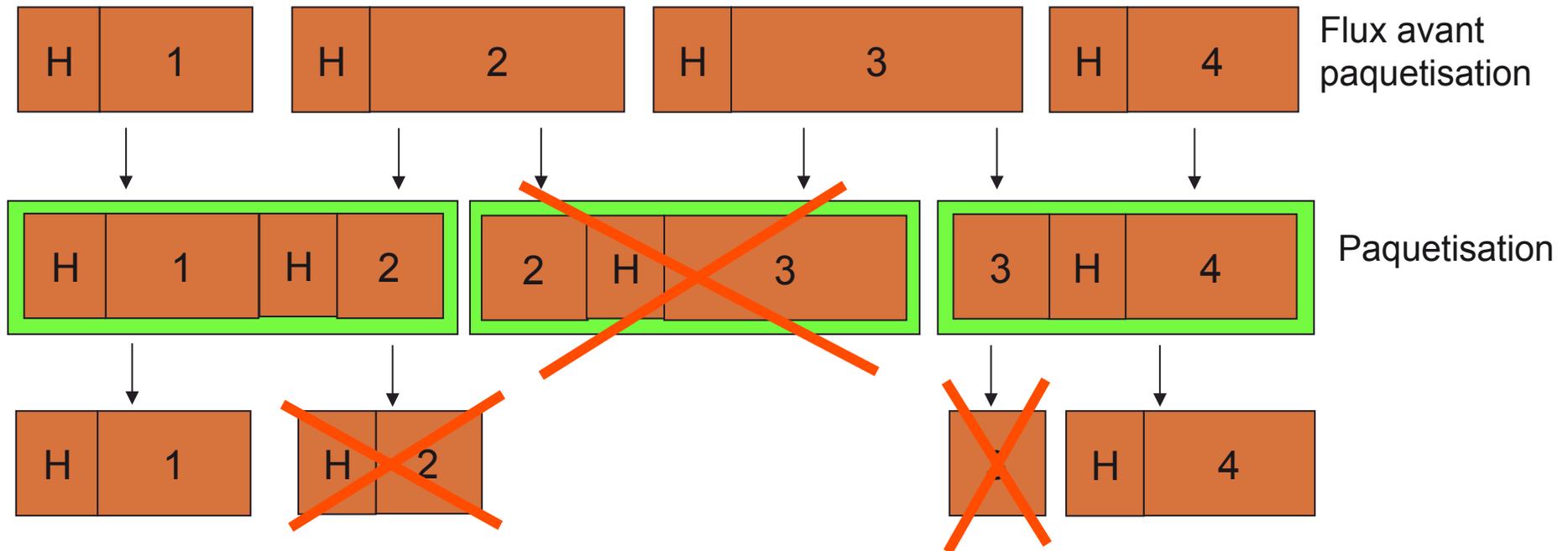


Exemple de FEC: entrelacement audio



- Amélioration de la qualité perçue en cas de perte
- Faible augmentation de débit (table d'entrelacement)
- Augmentation de la latence
 - Attente de réception de 4 paquets pour réordonner

Problème de perte de données – Importance de la paquetisation



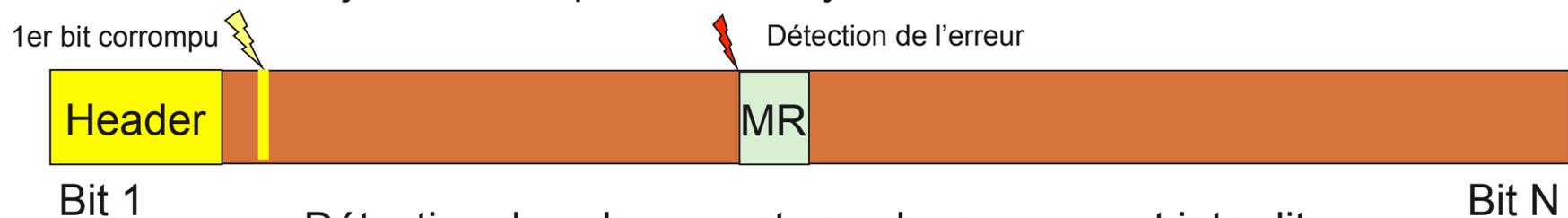
■ Règles de base

- Un paquet doit contenir un nombre entier d'AU
- Si une AU est trop grosse, elle est répartie sur plusieurs paquets
- Les fragments d'AU ne doivent pas être mélangés dans un même paquet

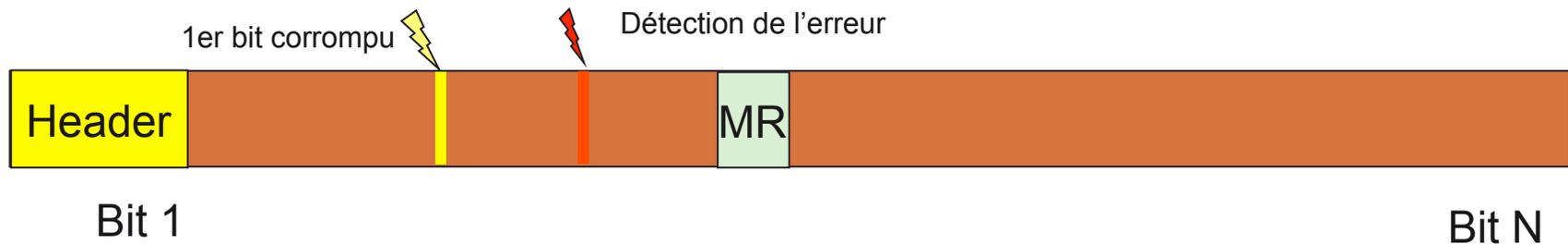
Problème de corruption de données – Techniques de codage robuste

■ Problème 1: Détecter les erreurs

- Se rendre compte qu'il y a eu une ou des erreurs
- Isoler la zone du flux binaire corrompue
- Exemples: Codage vidéo
 - Ajout de marqueurs de resynchronisation



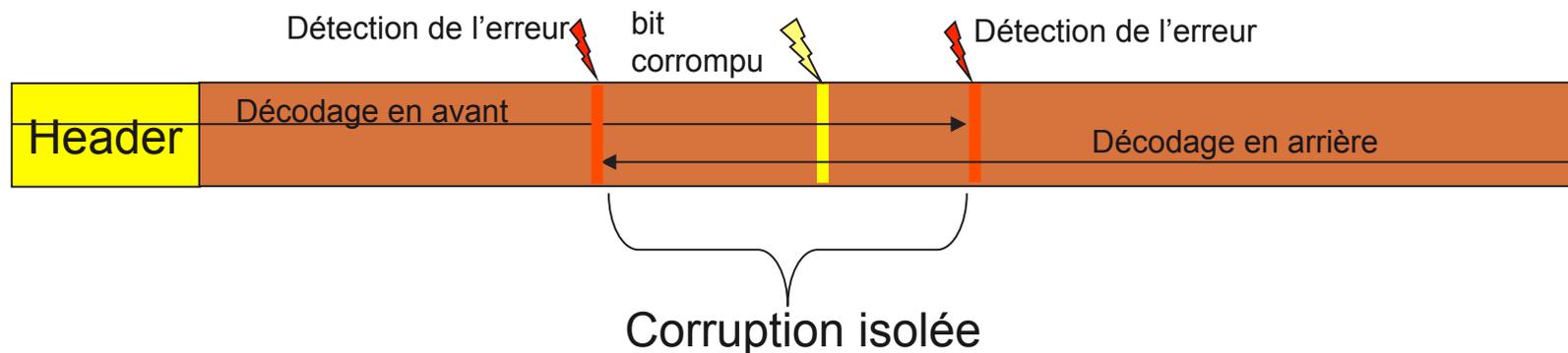
- Détection de valeurs vecteurs de mouvement interdites



Problème de corruption de données – Techniques de codage robuste (2)

■ Problème 2: Isoler les erreurs

- Technique de codage réversible
 - Problème de performance
 - Détermination des mots de resynchronisation





Problème de corruption de données – Techniques de codage robuste (3)

■ Problème 3: Dissimulation des erreurs

- Masquer les erreurs
- Exemple:
 - Technique audio: utilisation d'échantillon de silence
 - Technique vidéo: utilisation des éléments précédents (macro block ou image complète)



Problématiques de distribution de contenu

- **Gestion du temps**
- **Gestion du nombre de flux**
- **Gestion des contraintes réseaux**
- **Gestion du mode de connexion**
 - Mode Connecté: Négociation, Adaptation du contenu
 - Mode Broadcast: Le mécanisme de Carrousel



Problème de connexion en Broadcast

■ Particularité

- Connexion à n'importe quel moment

■ Problème

- Données transmises non décodables directement
 - Ex : Image P, Mises à jour de scène ...
- Délai d'attente de données décodables (ex: image I)

■ Solution: mécanisme de *Carousel*

- Principes:
 - Données envoyées cycliquement
 - ex: Object Carousel, Data Carousel, MPEG-2 DSM-CC, FLUTE
 - Utilisation d'un numéro de version pour détecter les nouvelles versions
- Utilisé pour la synchronisation en Broadcast
 - Attente des données carousel pour démarrer une session
- Utilisé comme mécanisme de correction d'erreur en environnement non fiable
 - Données importantes réenvoyées plus régulièrement