

Réseaux

Licence L3 Informatique

Ehoud Ahronovitz

UM2 - LIRMM

Université Publique Laïque

1 Introduction

- Quelle sauce
- Quel Plat
- Accompagnement
- Quelques bases
- Caractéristiques
- Architecture

Déroulement

- Cours 16.5h, TD 16.5h, TP 18h ;
- Contrôle : examen écrit (30), note TP ou projet (20) ;
- Prérequis : système : appel système, programmation C/C++ ;
- Conseils : voir et faire les examens précédents.

Attention : il **fa**ut approfondir le cours ⇒ voir la bibliographie, consulter, poser des questions. En un mot, apprendre.

Objectifs - Programme

Objectifs : Connaître les principes et mécanismes des réseaux. Répondre aux questions

- Qu'est-ce qu'un réseau ?
- Comment mettre en œuvre et utiliser des applications ?
- Qu'est-ce qui relève du travail d'informaticien dans les réseaux ?

Programme

- 1 Introduction générale : éléments de base, architecture.
- 2 Monde Internet : caractéristiques.
- 3 Mise en œuvre d'applications dans l'Internet. Client-Serveur.
- 4 Types de serveurs. Modes de connexion, protocoles sous-jacents.
- 5 Routage. Algorithmes et protocoles.
- 6 Configuration de réseaux et sous-réseaux.
- 7 Protocoles d'adressage, d'erreurs et contrôles.
- 8 Grandes applications : Messagerie électronique, W3, systèmes de fichiers distribués,...

bibliographie

- Andrew Tanenbaum, *Réseaux*, 4^{ème} édition, Pearson Éducation, 2003
Computer Networks, 4th edition, Prentice Hall, 2003.
- Guy Pujolle, *Les réseaux*, 3^{ème} édition, Eyrolles, 2000.
- Douglas Comer, *TCP/IP : Architecture, protocole et applications*, Pearson Éducation, 5^{ème} édition, 2006,
Internetworking with TCP/IP, Pearson Prentice Hall, 5th edition, 2006.
- J.Kurose, K. Ross, *Analyse structurée des réseaux*, 2^{ème} édition, Pearson Éducation, 2003.
- Cizault *IPV6 : Théorie et Pratique*, 3^{ème} édition, O'Reilly France, 2002.
- Christian Huitema, *Le routage dans l'internet*, Eyrolles, 1995 (2^{ème} édition en anglais); *IPV6* P. Hall, 1997; ...*Et dieu créa l'Internet...*
- W. Stallings *Computer Networking with Internet Protocols and Technology*, Pearson-P. Hall, 2004.

Réponses Technologiques

- Des liaisons physiques : câbles, fibres optiques, ondes, ... Chaque support a ses propres caractéristiques, essentiellement une distance liée à un débit, ainsi qu'une distance maximale. On verra que des caractéristiques qui semblent identiques ne donnent pas les mêmes résultats effectifs.
- Des protocoles : "méthodes" communes ; accords sur des règles permettant aux entités communicantes de se comprendre ; on parlera de protocoles d'application lorsqu'on considère les applications communicantes, de protocoles de plus bas niveau au fur et à mesure que l'on se rapproche du fonctionnement du matériel.
- Comme tout système informatique, une construction par couches successives, tant **matérielles** que **logicielles**. On parlera d'architecture des réseaux puis des applications.

Présentation du problème

Besoins des **utilisateurs** : des applications qui *communiquent*. Pour *communiquer entre eux* et *partager des ressources communes*. Ceci implique dans tous les cas un échange de données entre des applications.

Hypothèses :

- les applications sont sur des ordinateurs,
- les ordinateurs fonctionnent de façon autonome,
- les ordinateurs disposent d'un accès au *périphérique* réseau ; en terme de système d'exploitation, il y a un contrôleur (une carte réseau) et un pilote permettant de lire et écrire sur le périphérique. La particularité de ce périphérique est qu'il est partagé.

Exemple de protocole

Le terme *protocole* est associé à des notions très différentes, en fonction du domaine d'utilisation.

Du point de vue des réseaux, c'est une suite de règles que doivent respecter les entités (au moins deux, mais aussi plus!) qui communiquent. Si une entité ne le respecte pas, la communication échoue.

Une conversation téléphonique, **après la phase d'appel**. Le protocole de la conversation commence lorsque la personne jointe décroche :

- 1 la personne jointe doit dire quelque chose :
 - *allo*
 - *bonjour*
 - *ne raccrochez pas*
 - *pour continuer composer le 76549231876 terminez par 655219088*
- 2 à l'une des deux premières réponses, la personne appelante répond pour démarrer la conversation,

Exemple de protocole suite

- 3 dans les autres cas, elle continue à patienter (retour au début du protocole),
- 4 lorsqu'enfin les deux personnes peuvent discuter, le protocole impose de ne pas parler les deux à la fois,
- 5 pour terminer la conversation une des deux personnes doit annoncer à l'autre sa volonté de finir,
- 6 l'autre personne peut refuser mais un accord mutuel est nécessaire,
- 7 la conversation se termine lorsque l'une des deux a raccroché.

Remarque : Si une personne a raccroché sans avertir, elle n'a pas respecté le protocole.

Attention : Ce protocole est « souple » car les entités communicantes remplacent les défaillances par une certaine intelligence, absente dans les réseaux. . .

Attention

- Un **protocole** ne s'exprime pas toujours sous forme d'un algorithme ;
- Il y a toujours négociation entre **plusieurs** (au moins deux) entités, souvent sous forme :
 - de questions : chaîne de caractères reconnue expédiée par un demandeur, test portant sur un élément commun, . . .
 - et de réponses : chaîne de caractères reconnue expédiée par le répondeur, réponse au test, . . .
 permettant de réaliser, retarder ou interdire une action.

Exercice

- Écrire le protocole d'une conversation à trois personnes.
- Difficile ? Laisser tomber et revenir aux protocoles des réseaux.
- Exemple **simplifié** d'un protocole entre deux applications :

fini=faux;

tant que (*non fini*) **faire**

 lireUnMessage() ;

si (*premierCaractère == 'F'*) **alors**

 fini=vrai;

sinon

si (*premierCaractère == 'B'*) **alors**

 repondreBonjour(nomDemandeur,monNom);

 dialoguer();

sinon

 expedierErreur();

Un peu de vocabulaire : les grands classiques

Structure : des communications : on utilise ce terme pour désigner la forme logique sous laquelle les entités communiquent ; on parlera de communications *point à point* (seules deux entités concernées à la fois), *multipoints* (plusieurs entités concernées, dérangées (. . .)), *diffusion* par exemple).

Topologie : forme géométrique de la connexion physique : étoile, bus, anneau, arbre, maillage régulier, Noter qu'on peut traverser plusieurs réseaux de topologies différentes et on parlera alors d'interconnexion.

Exemples classiques :

- anneau à jeton avec une topologie d'anneau et une communication point à point,
- bus ethernet, avec une topologie de bus et une communication par diffusion.

Vocabulaire encore : Architecture

Architecture : ensemble (empilement, hiérarchie) de protocoles. On parle de *Modèles en couches*.

Domaine : extension géographique

- RLE (LAN); unité ≈ 1 km,
- MAN; unité $\approx 10^3$ km,
- WAN au delà

caractéristiques fondamentales : quel débit sur quelle distance.

Vrai problème : distinguer réseau physique (média connectant physiquement plusieurs ordinateurs) et réseau logique (*virtuel*, interconnexion de plusieurs réseaux physiques).

Interconnexion

Dans la figure précédente, toutes les flèches étaient à *l'intérieur* d'un réseau physique !

L'**interconnexion** est assurée par des ordinateurs simples ou des machines spécialisées : concentrateurs (*hub*), commutateurs (*switch*), routeurs (*router*), passerelles (*gateway*) en fonction de la spécialisation atteinte.

⇒ Leur problème essentiel est l'efficacité.

⇒ Leur rôle est d'assurer la **commutation**, c'est-à-dire le transfert de l'information entre un point d'entrée et un point de sortie. On parle de :

- commutation de circuits (vieille technologie du monde de la vieille téléphonie, mais pouvant refaire surface),
- ou de commutation de paquets, chaque entité d'interconnexion stocke la donnée (un paquet), détermine le prochain destinataire et lui fait suivre la donnée (*store & forward*).

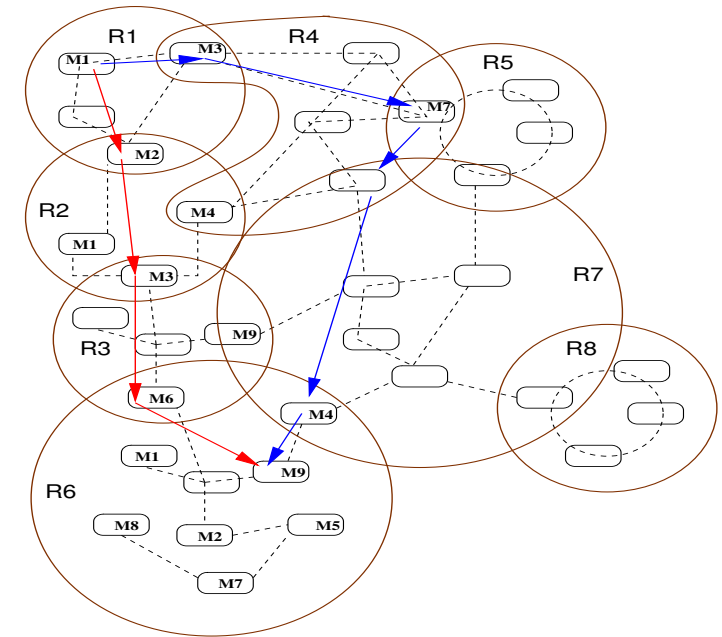
Ces transferts s'opèrent dans un **mode sans connexion** ou **connecté** (on parle alors de circuit virtuel).

Une toile, pas une tuile

Réseaux divers à architectures différentes interconnectés

Quel sens pour le *meilleur* chemin ?

Vaste programme



Tartes à la crème

Plein de normes existent, utiles, certes, mais il est inutile de se laisser submerger, du moins pour ce cours.

normalisation	v24 v28 v35 v90	modems
	x21 x25 x29 x400	ccitt
	802.2 802.3 802.4	ieee
	802.11 802.11g	ieee, sans fil
	8802/2 8802/3 8802/4	iso

- modèle OSI de l'ISO (cf. architecture, dans la suite de ce chapitre)
- Obsolètes : Architectures SNA (ibm), DSA (bull), DECNET (dec), etc
- Bien sûr : Internet ; on le verra, il s'agit d'un ensemble de protocoles et d'applications.

Caractéristiques des supports

Unité utilisée : X bits par seconde. Notation : $X\text{bit}.s^{-1}$

Exemples :

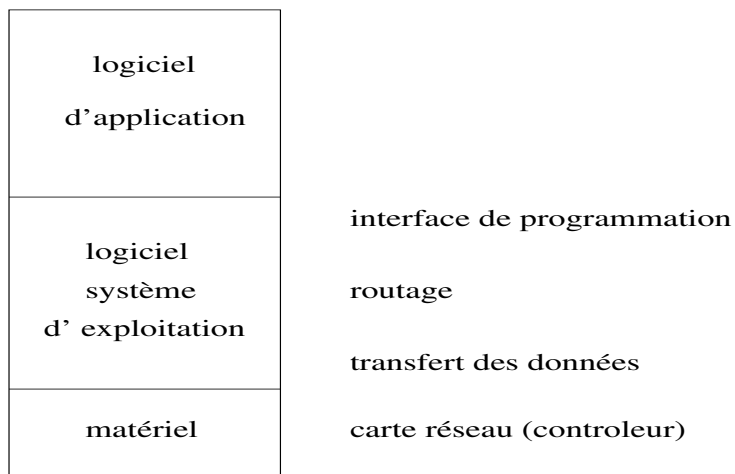
- $55\text{Kbit}.s^{-1}$ est un "bon" débit lorsqu'on utilise une ligne téléphonique avec un modem classique (appareil permettant la connexion d'un ordinateur personnel avec un réseau d'un opérateur, utilisant sa propre ligne téléphonique) ;
- $256\text{Kbit}.s^{-1}$ à $10\text{Mbit}.s^{-1}$ pour un modem adsl (modem spécialisé) ;
- $100\text{Mbit}.s^{-1}$ sur le réseau local de l'ufr, lui-même connecté au monde extérieur par une ligne offrant $8\text{Mbit}.s^{-1}$.

Exercice

Calculer le débit minimal nécessaire pour effectuer une vidéo transmission (25 images par seconde) avec des images de 1024×1024 points (pixels) et un codage de la couleur sur 32 bits sans compression ?

Caractéristiques des logiciels

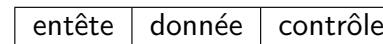
Offrir des services aux applications : stockage, acheminement, accès au matériel, interface de programmation ...



Caractéristiques des données

Unité fondamentale, le **paquet**, suite d'octets. Autres noms, en fonction de la couche étudiée ou de la spécificité du réseau : trame, cellule.

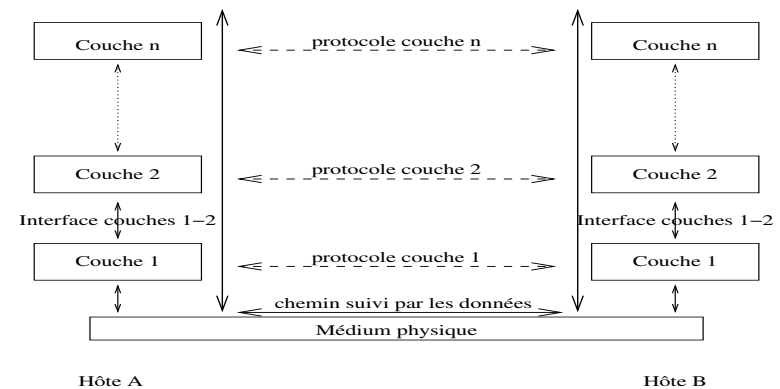
Chaque paquet contient outre les données à transmettre, une partie (entête) contenant des adresses permettant d'acheminer le paquet et une partie de contrôle, permettant de contrôler la validité des données.



On peut prévoir dans l'entête les adresses source et destination.

Question : quelle destination ? locale ? suivante ? finale ?

Architecture : construction en couches



protocole : règles et conventions utilisées entre couches homologues (sur deux hôtes) .

interface : règles et conventions utilisées entre couches voisines (sur un même hôte).

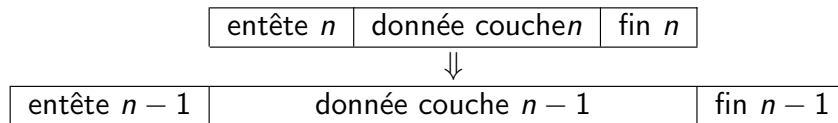
Encapsulation - Un Début

Principe : Chaque couche a ses propres impératifs liés à

- l'adressage (forme et codage de l'adresse),
- la taille (maximale et minimale) du paquet,
- la méthode de réalisation du contrôle.

Elle est donc amenée lors de l'**expédition** à :

- envelopper (on dit encapsuler) le paquet transmis par la couche au dessus dans la partie *donnée* de son propre paquet,



2 – couche liaison

Il faut dire *liaison de données*. Caractéristiques et fonctions :

- **Unité** traitée : un paquet ;
- elle transfère des paquets de source à destination ; on parle de *trame*, de *cellule*. . . selon les propriétés et données contenues dans les paquets.
- correction d'erreurs,
- règles de partage du support,
- qualité de service.

remarque : jusque là il s'agit d'une liaison directe entre deux hôtes, sans changement de support physique.

Exemple : *ethernet* est un ensemble (matériel et logiciel) permettant de réaliser les impératifs de cette couche. Il est aujourd'hui intégré dans les cartes réseau *ethernet*.

Architecture : problème de couches

Architecture OSI : un modèle, permettant d'illustrer la pile de problèmes à résoudre.

On s'intéresse dans cette partie surtout à la **diversité des problèmes**.

1 – couche physique

- moyens de transmission d'éléments binaires
- modems, multiplexeurs,
- capacités électroniques
- **unité** traitée : un bit, au mieux un octet.

Ses caractéristiques induisent des performances en termes de débit (on dit aussi *bande passante*). On ne l'étudie pas dans ce cours.

Protocole ethernet

- La partie matérielle d'*ethernet* permet de détecter si le support est libre ou occupé par une transmission ; mais si deux hôtes détectent simultanément que le support est libre ?
- Heureusement elle permet aussi de détecter une collision.

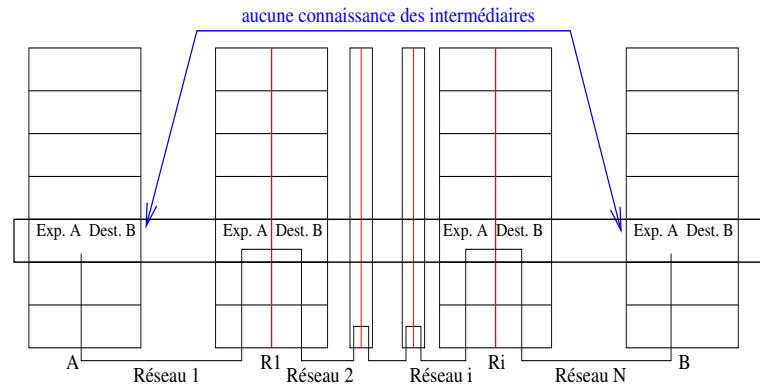
Schéma algorithmique d'accès et partage du support d'*ethernet* :

```

paquetExpédié = faux ;
tant que (non paquetExpédié) faire
  tant que (non supportLibre) faire
    ⊥ attendre ;
  expédier paquet ;
  si collision alors
    ⊥ tirer délai aléatoire;
  sinon
    ⊥ paquetExpédié = vrai ;
  
```

3 – Couche réseau

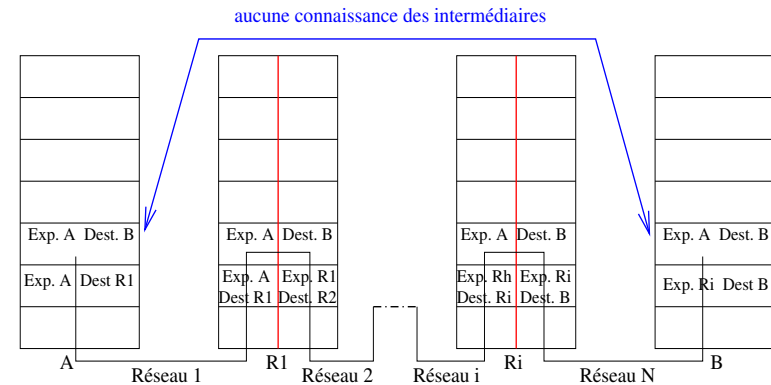
Changement important : Le destinataire final peut se situer dans un réseau distant, différent par la structure et la topologie de celui de l'expéditeur. Il faut passer par des intermédiaires (routeurs). Il faut interconnecter des matériels et des réseaux hétérogènes.



Couche réseau - un peu plus

- contrôle de flux : baisser, augmenter la cadence en fonction de l'état des espaces tampon,
- routage : trouver un chemin adéquat ou au moins le prochain nœud,
- adressage : quelle forme, comment passer de l'adresse réseau à l'adresse physique (adresse physique et adresse liaison sont souvent utilisés comme synonymes)
- mode connecté (exemples : X25, certains réseaux publics) ou
- mode sans connexion (exemple : IP protocole de l'*Internet*)
- Attention : taille des paquets \neq taille des trames ; donc découpage possible et besoin de réassembler les morceaux.

Et pourtant il circule



4 – Couche transport

On passe du niveau d'un paquet à celui d'une suite de paquets. À nouveau, des adresses source et destination seront associées, internes à chaque hôte, pour assurer :

- fiabilité ; que faire si un paquet est perdu ? mais qu'est-ce que la fiabilité ?
- ordre des paquets ; est-ce que deux paquets peuvent suivre des chemins différents ?
- mode connecté ou sans connexion (encore!!! Tout dépend du service offert par le voisin en dessous)
- qualité de service ; notion importante, dépendante du service rendu par les trois premières, mais difficile à exprimer.
- exemples :
protocoles TP0 à TP4 dans le monde ISO
protocoles TCP, UDP, ... dans le monde Internet

5 – Couche Session

- établissement d'une connexion, maintien, libération ; penser à une session de travail ;
- points de reprise, synchronisation ; que faire lorsqu'une communication s'est interrompue inopinément ?

Exercice : Trouver au moins une application dans laquelle une session avec points de reprise est indispensable pour un fonctionnement correct. Trouver une autre dans laquelle on peut s'en passer sans dommage.

7 – Application

Ce n'est pas un fourre-tout pour autant. Penser aux

- protocoles de messagerie (acheminement, transcription des adresses),
- protocoles de la toile : http, https,
- protocoles de transfert de fichiers (multi-fichier, compression),
- codage des images (type de codage, compression),
- synchronisation d'horloges...

En quelque sorte, le début des problèmes lorsque la partie transmission sur les réseaux *fonctionne* (?)

Prévoir un protocole par nouvelle application.

6 – Couche Présentation

- syntaxe des données
- problèmes de représentation car environnement hétérogène ; exemple : gros et petit boutiste pour le codage des entiers. Pour mémoire :

	octet1	octet2	octet3	octet4
gros boutiste	poids fort $2^{31} \dots 2^{24}$	$2^{23} \dots 2^{16}$	$2^{15} \dots 2^8$	poids faible $2^7 \dots 2^0$
petit boutiste	poids faible $2^7 \dots 2^0$	$2^{15} \dots 2^8$	$2^{23} \dots 2^{16}$	poids fort $2^{31} \dots 2^{24}$

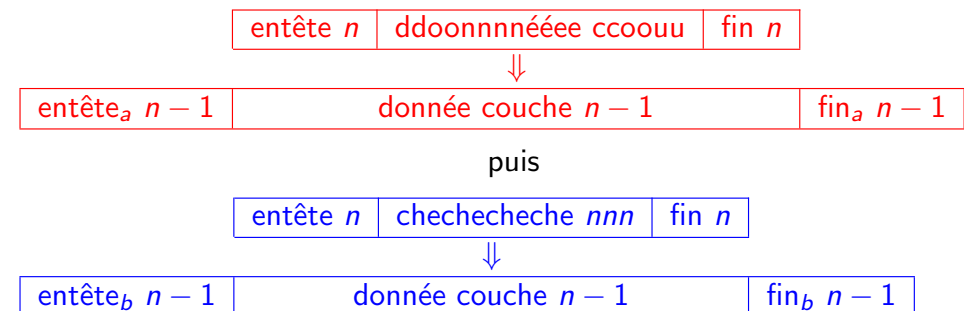
Mais il n'y a pas que des entiers à transmettre... **Quelques solutions**

- XDR
- langage ASN1

Encapsulation et Découpage

Pour tenir compte de ses propres caractéristiques, chaque couche peut être amenée, lors de l'encapsulation à

- découper ce paquet en tranches et transmettre alors chaque tranche dans un paquet



Exemple d'encapsulation

Un paquet de la couche réseau, dans le protocole IP de l'*Internet* encapsulé dans un paquet ethernet :

entête eth				ent. IP	donnée IP	CRC
donnée eth						
preamb	D eth	S eth	type	368 à 12000 bits		32 bits
64 bits	48 bits	48 bits	16 bits			

Sachant que :

longueur paquet IP ≤ 65536 octets

64 octets \leq longueur totale trame ethernet ≤ 1518 octets

Pour citer au moins une autre technologie : une trame ATM est de longueur fixe, 53 octets, dont 48 pour la donnée et 5 pour l'entête.

Interconnexion de réseaux – la base

niveau	outil
physique	répéteur, concentrateur, <i>hub</i>
liaison	pont, commutateur, <i>switch</i> pont filtrant, pont inter-réseaux
réseau	routeur, <i>router</i>
plus haut	passerelle de ..., <i>gateway</i>

- Il existe des produits intermédiaires : pont–routeur, ...
- À chaque niveau, la machine réalisant l'interconnexion est capable de traiter le paquet correspondant (sauf pour le niveau physique). Elle reconnaît et peut séparer tous les éléments de l'entête ou contrôler la validité du paquet.
- Le problème : performance \Rightarrow machines spécialisées.

Réciproquement

Chaque couche est amenée lors de la **réception** à :

- détecter une éventuelle anomalie en recalculant le code de contrôle,
- décapsuler le paquet : enlever entête et contrôle et transmettre au voisin.

Remarque : le code de contrôle n'est pas systématiquement présent dans toutes les couches ; il peut aussi être effectué sur une partie du paquet seulement et être incorporé dans l'entête.

Questions :

- est-ce que le découpage à l'encapsulation peut intervenir à une position quelconque dans le paquet découpé ? La réponse est oui (petit arrondi possible), mais la justification ?
- lorsqu'il y a eu découpage, qui (quelle couche, sur quel hôte) doit faire le réassemblage ? Réponse vaste à garder au chaud.