

Modèles OSI et TCP/IP

Couche physique

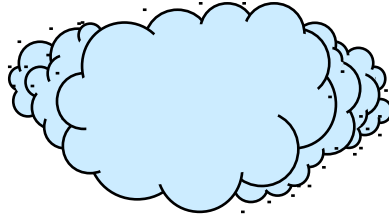
Couche liaison de données

Module FAR

Polytech Montpellier – IG3

David Delahaye

Pourquoi une architecture en couches ?



transfert de fichier de A à B via un réseau

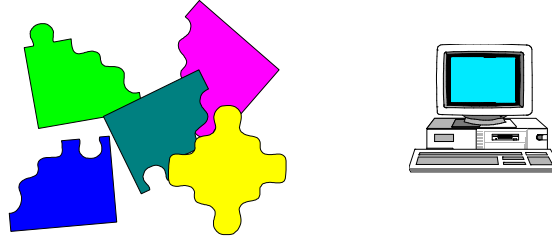
- acheminer des bits
- transporter des paquets
- gérer les échanges d'applications
- etc.
- altérations de données
- pertes de données
- congestions du réseau
- pannes matérielles
- etc.



il faut décomposer le problème !

Bcp de pb qd transfert. Donc on va diviser en une succession de couches pour résoudre ça.

Comment décomposer ?



principe : structuration en couches (niveaux)

- chaque couche est construite sur la précédente
- le nombre, le nom, le contenu et les fonctions des couches diffèrent d'une architecture à l'autre
- dans toutes les architectures, le rôle de chaque couche est d'offrir des services à la couche supérieure

Pourquoi une architecture en couches ?

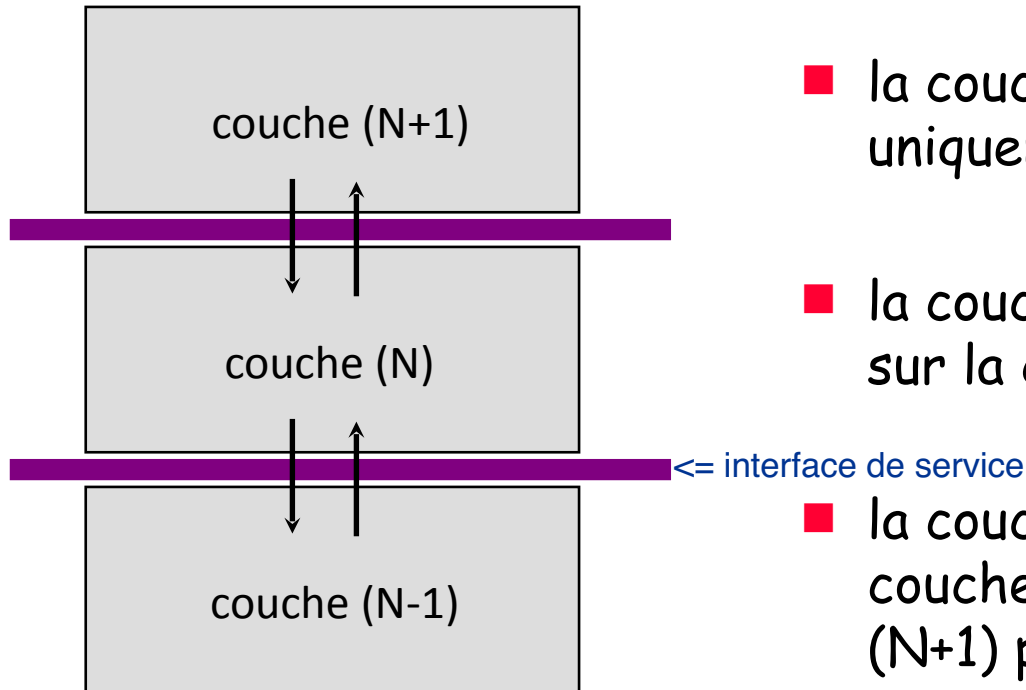
Lorsqu'on doit concevoir un système complexe

- une structure explicite permet l'identification et les relations entre les différentes parties du système
 - un modèle de référence en couches constitue une base de discussion
Pour avoir un vocab commun à tous
- la modularité facilite la maintenance et la mise à jour du système
 - la modification d'une couche reste transparente au reste du système



2 aspects : vertical et horizontal

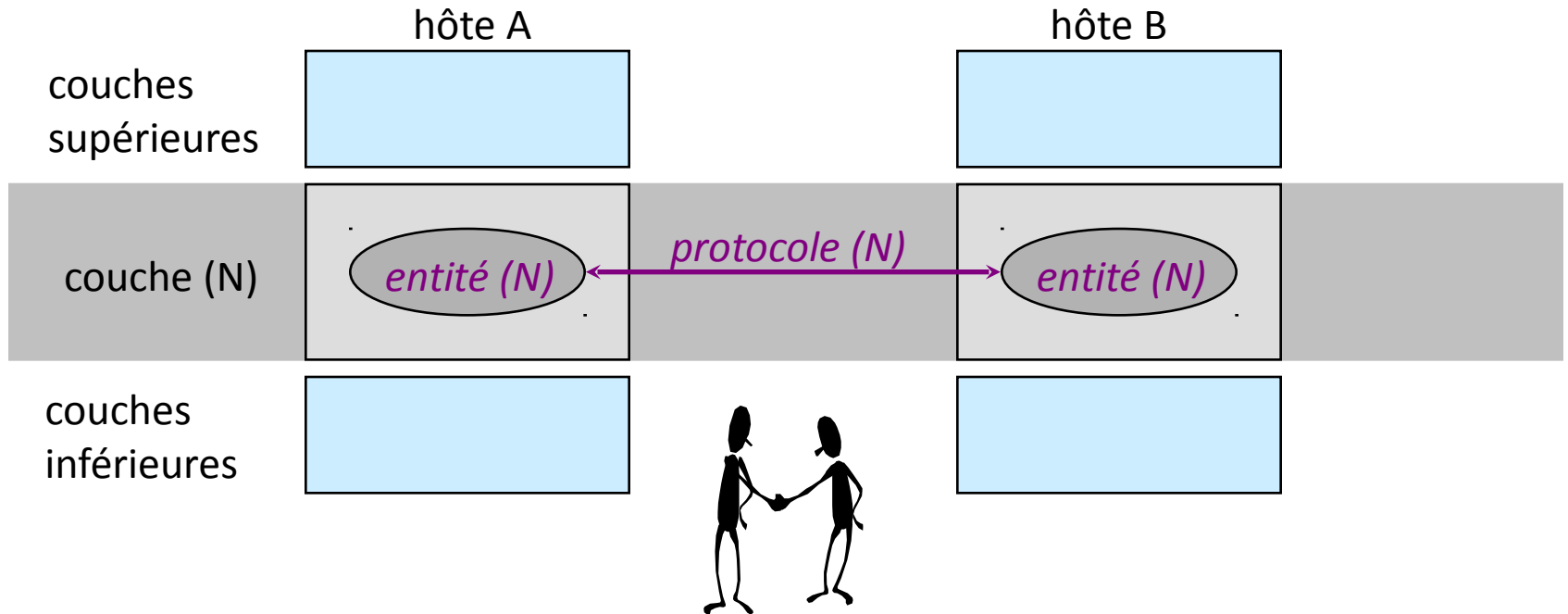
Aspect vertical ?



- la couche (N+1) voit la couche (N) uniquement par le *service* offert
- la couche (N+1) n'a aucune vue sur la couche (N-1)
- la couche (N) est séparée de la couche (N-1) et de la couche (N+1) par une *interface de service* bien définie

La couche N-1 est complètement indépendante de la couche N+1

Aspect horizontal

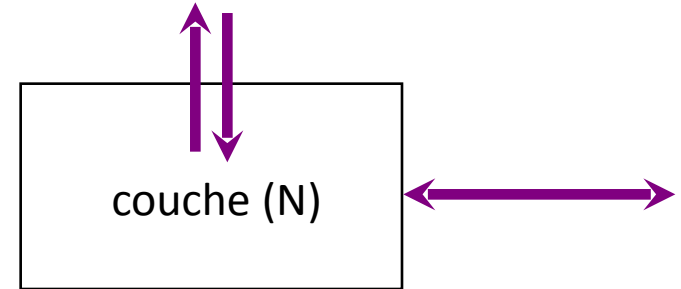


- le *protocole* (N) définit les règles de communication à l'intérieur de la couche (N)
- les *entités* (N) représentent les éléments actifs de la couche (N)

Une couche

Est définie par :

- ✓ une interface de service
- ✓ un protocole de communication
(= son langage)

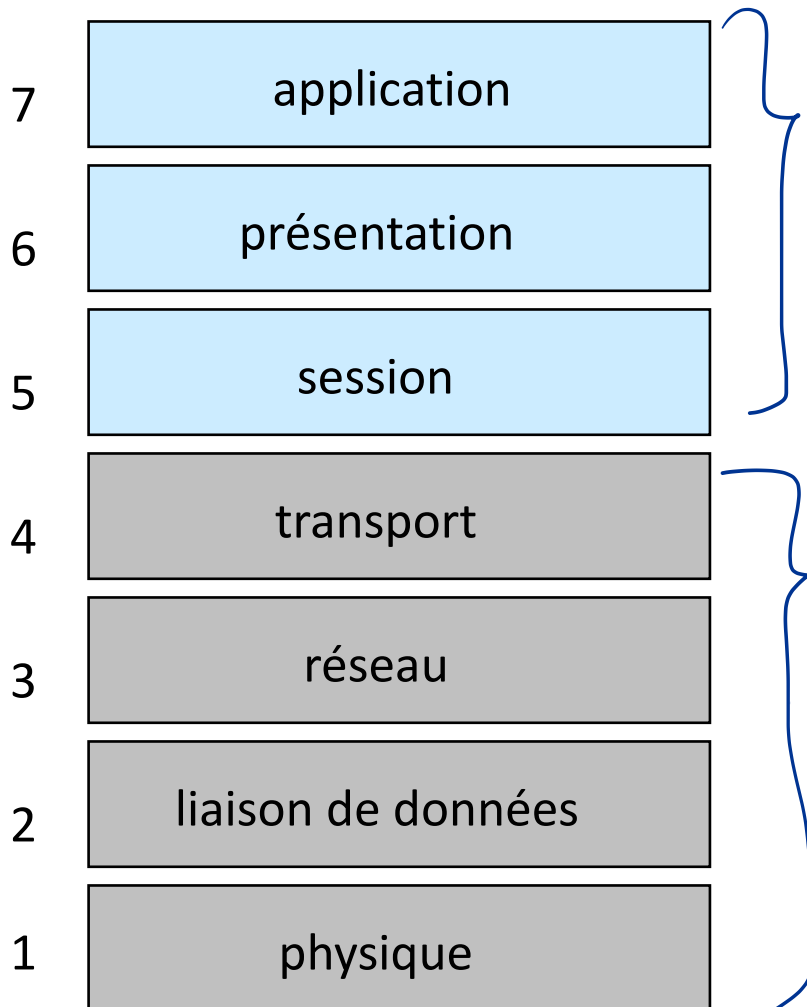


- ✓ Principes de l'empilement des couches
 - chaque couche utilise les services de la couche inférieure
 - chaque couche offre des services à la couche supérieure au travers d'une interface de service
 - les entités d'une même couche communiquent entre elles selon un protocole
- ✓ Une **architecture de réseau** = un ensemble de couches

Le modèle de référence OSI

- Open Systems Interconnection
- travaux entrepris à l'ISO en 1978
- norme parue en 1980 : IS 7498
- pourquoi faire ?
 - ↳ régler les problèmes de l'interconnexion de systèmes hétérogènes (logiciel et matériel)
- portée du modèle ?
 - ↳ il ne concerne que l'interconnexion et n'est utilisé que pour décrire les communications entre systèmes
 - modèle abstrait
 - modèle indépendant des logiciels et technologies

Les 7 couches OSI



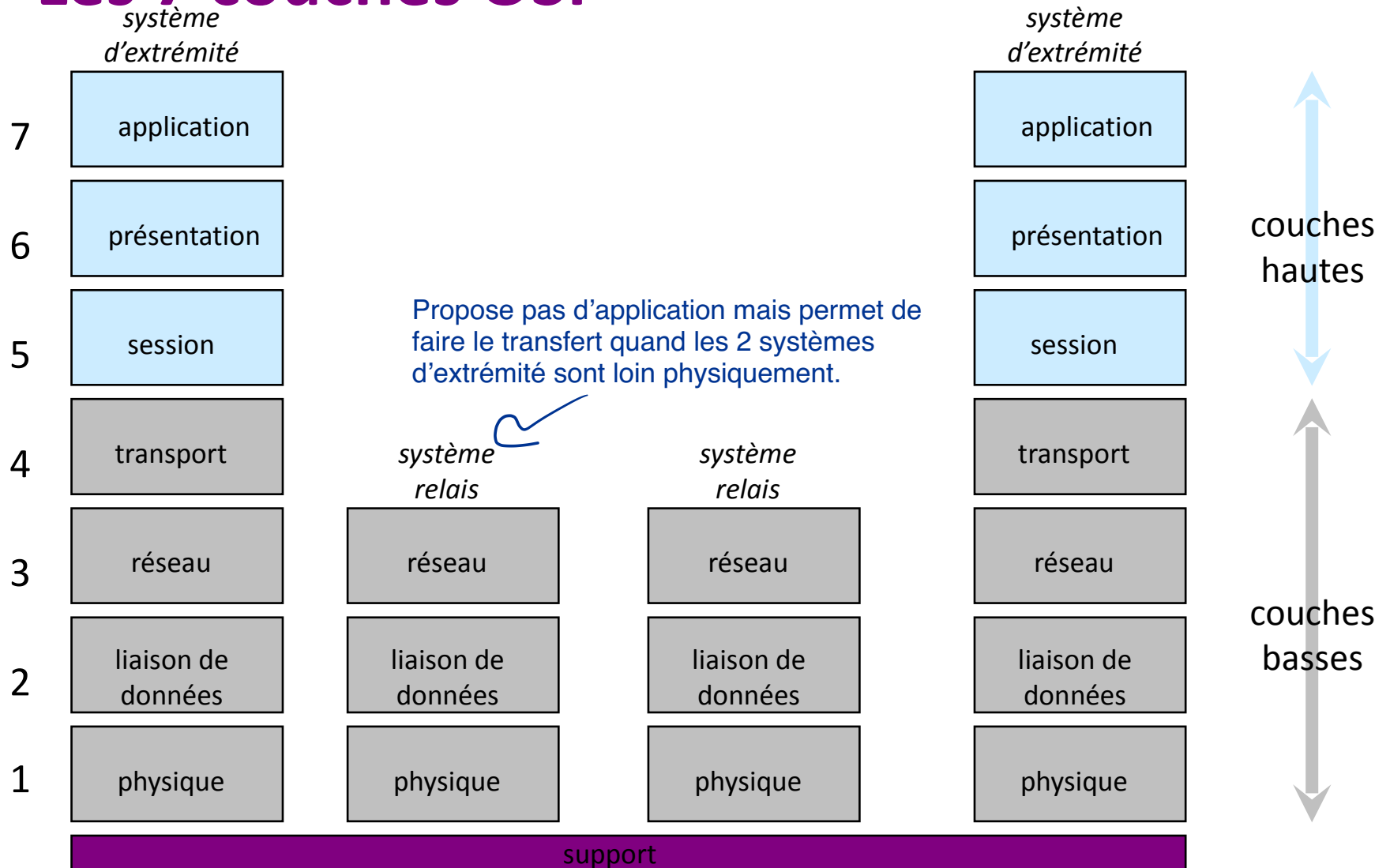
- **couches hautes**

- rendent un service d'accès
- comportent les fonctions de traitement sur les données transportées

- **couches basses** Propage l'info

- rendent un service de transport
- comportent les fonctions de transmission de données

Les 7 couches OSI



Rôle des 7 couches

- **physique** (= description du matériel (électroniquement))
 - responsable de la transmission des *bits* sur un circuit de communication
 - spécification des connecteurs
 - détermination des caractéristiques électriques des circuits
 - définition des procédures d'utilisation des connexions physiques
- **liaison de données** (met en place le transfert proprement, vérifié la syntaxe et contrôle la donnée si pas d'erreur a l'envoi et à la réception)
 - responsable de la transmission fiable de *trames* sur une connexion physique
 - contrôle de flux
 - contrôle d'erreur

Rôle des 7 couches

- **réseau** (La ou on donne les adresses des sources, Pour dire à quel ordi on envoie les données)
 - responsable du transfert de données à travers le réseau
 - adressage
 - routage
 - contrôle de congestion
- **transport**
 - responsable du transfert de bout-en-bout, avec fiabilité et efficacité
 - contrôle de flux
 - reprise sur erreur
 - optimisation

Rôle des 7 couches

- **session** Pour se connecter
 - responsable des mécanismes nécessaires à la gestion d'une session
 - organisation du dialogue
 - synchronisation du dialogue
 - établissement et libération d'une session
- **présentation**
 - responsable de la représentation des données échangées entre applications
 - traduction des données
 - compression
 - cryptage
- **application**
 - fournir à l'utilisateur des services pour réaliser une application répartie et pour accéder à l'environnement OSI

Communication en mode connecté

(Avec un câble)

- la communication comporte 3 phases
 - établissement de connexion
 - transfert de données
 - libération de connexion

😊 transfert fiable (Car que ceux qui sont connectés au câble ont accès aux infos)

😊 minimisation du volume d'information de contrôle

😞 procédures très lourdes (établi. et libération)

😞 convient mal à la diffusion d'information

😞 manque d'efficacité pour les échanges sporadiques

Communication en mode non connecté

- ni établissement ni libération
 - les données échangées sont considérées comme étant des messages (**datagrammes**) complètement indépendants les uns des autres

😊 simplicité

😊 robustesse

😞 transfert non fiable (Car tout le monde peut discuter sur le même réseau)

😞 information de contrôle récurrente

(On doit mettre bcp plus d'infos en plus de ceux qui nous intéressent)

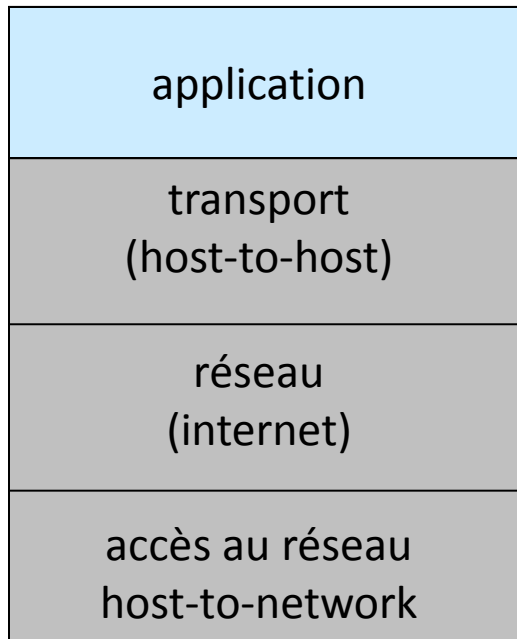
Un bit = la plus petite donnée (0 ou 1)

Séquence de bits = une infos

Datagram/trame/message = ensemble de données concaténé

L'architecture TCP/IP

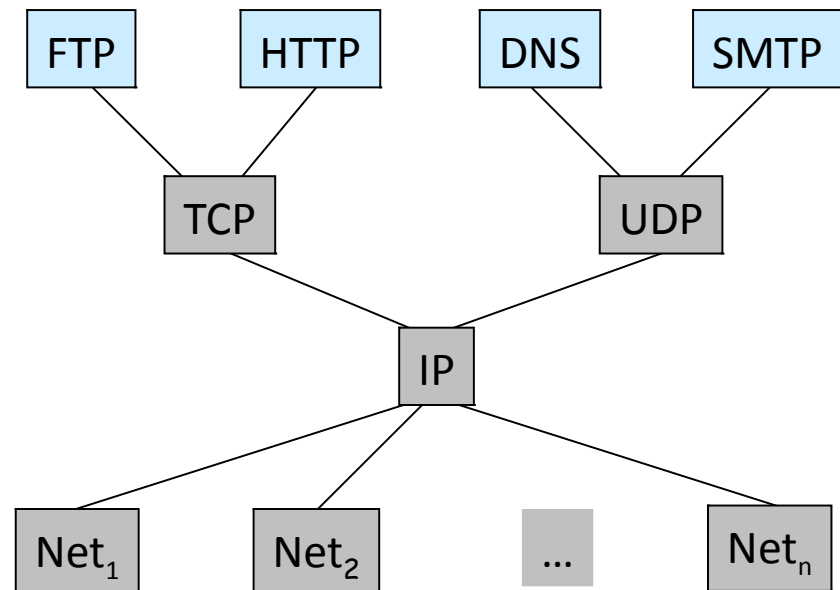
(Modèle plus simple que OSI)



Architecture
à 4 niveaux

TCP : on discute en direct qd communication établie (comme téléphone)

UDP : envoie l'info mais réponse Pas directe (comme lettre postale ou mail)



Graphe (simplifié)
des protocoles TCP/IP

Protocole FTP : pour échanger 2 fichiers entre 2 ordi

Protocole HTTP : pour échanger de l'info de façon bidirectionnelle

Protocole SMTP : pour envoyer des mails

L'architecture TCP/IP

- Les 4 couches :
 - La couche **application** est responsable des applications réseau.
 - La couche **transport** offre un service de transport de messages entre les extrémités client et serveur d'une application.
 - La couche **réseau** permet l'échange de données entre deux machines raccordées à des réseaux différents : elle est responsable du routage des datagrammes entre ces deux machines.
 - La couche **accès au réseau** traite les échanges de données entre un hôte et le réseau auquel il est attaché et se charge également du routage de données entre deux équipements rattachés au même réseau. (Cmt lordi envoie sons signal au réseau ...)

TCP/IP vs. OSI

Moins de couches , donc plus simple et facile à expliquer et moins de règles.

- 4 couches vs. 7
 - pas de session, ni de présentation
 - pas de couche physique
- couche réseau uniquement en mode non connecté
 - cahier des charges initial de l'ARPA
- La couche accès au réseau n'est pas une couche au sens réel du terme : c'est une interface entre les couches réseau et liaison OSI
- moins de formalisme dans la définition des couches
 - notions d'interfaces, de services ???
 - moins bonne isolation des protocoles
- ne fait pas de distinction claire entre services, interfaces et protocoles
- ne fait pas de différence entre spécification et implémentation
- n'est pas général et est peu approprié pour décrire toute autre pile de protocoles

Couche physique



- **ETTD** (Equipement Terminal de Traitement de Données), en anglais *DTE, Data Terminal Equipment*
 - effectue les traitements sur les données (ainsi que la supervision de la transmission de données)
 - il s'agit d'un ordinateur, d'une imprimante, d'un routeur, ...

Un support de transmission



Les différents supports

- Filaire (paires torsadées, câble)
- Optique (fibre optique)
- Aérien (radio terrestre, radio satellite)

Les critères de comparaison

- bande passante
- atténuation (Plus de perte avec câble que onde radio)
- sensibilités diverses
- coût
- encombrement
- poids, etc.

Supports de transmission

- Différents supports physiques avec différents débits :
 - Paires torsadées (de 4 Mbits/s à 1 Gbits/s) ;
 - Câble coaxial (10 Mbits/s) ;
 - Fibre optique (2 Gbits/s) ;
 - Infrarouges (4 Mbits/s) ;
 - Ondes radio (11 Mbits/s) ;
 - Micro-ondes (10 Mbits/s) ;
 - Satellite (20 Mbits/s). Comme telephone

Analogique vs. numérique

- **message** = données que l'utilisateur souhaite transmettre
- **message analogique** (CONTINU)
 - espace de temps continu, espace de valeurs continu
 - ex : voix, vidéo, données collectées par des capteurs



On transforme ce signal en sequence de 0 et de 1 :

- **message numérique** (DISCONTINU)
 - espace de temps discret, espace de valeurs discret
 - ex : texte, entiers

Analogique vs. numérique

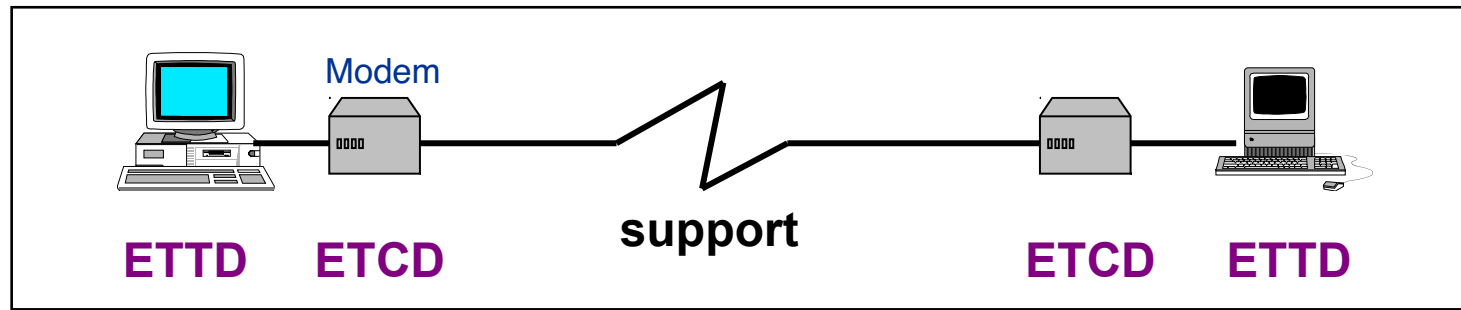
- La **transmission** est l'opération qui consiste à transporter le signal d'une machine vers une autre, sur un support donné
- **transmission analogique**
 - transport d'un signal analogique
 - nécessite, sur de longues distances, des **amplificateurs**
(Permet de réajuster la puissance du signal de façon continue)
- **transmission numérique**
 - transport d'un signal numérique
 - nécessite, sur de longues distances, des **répéteurs**
(Enregistre les infos dans sa mémoire et les réémet de manière discontinue (le renvoie plus tard))

La numérisation

- Avantages du numérique sur l'analogique
 - facilités de stockage, de traitement et de restitution
 - intégration (multimédia)
 - faible taux d'erreur des liaisons numériques par rapport aux liaisons analogiques (répéteurs vs. amplificateurs)
 - coût des composants (équipements) numériques inférieur à celui des composants analogiques *Et plus performants*
- ↳ la tendance : traiter des données numériques et les véhiculer par un signal numérique (*le tout numérique*)
- **numérisation** : transformation d'un message analogique en un message numérique

Signaux analogiques se rarifient de plus en plus car moins avantageux que numérique et moins pratique.

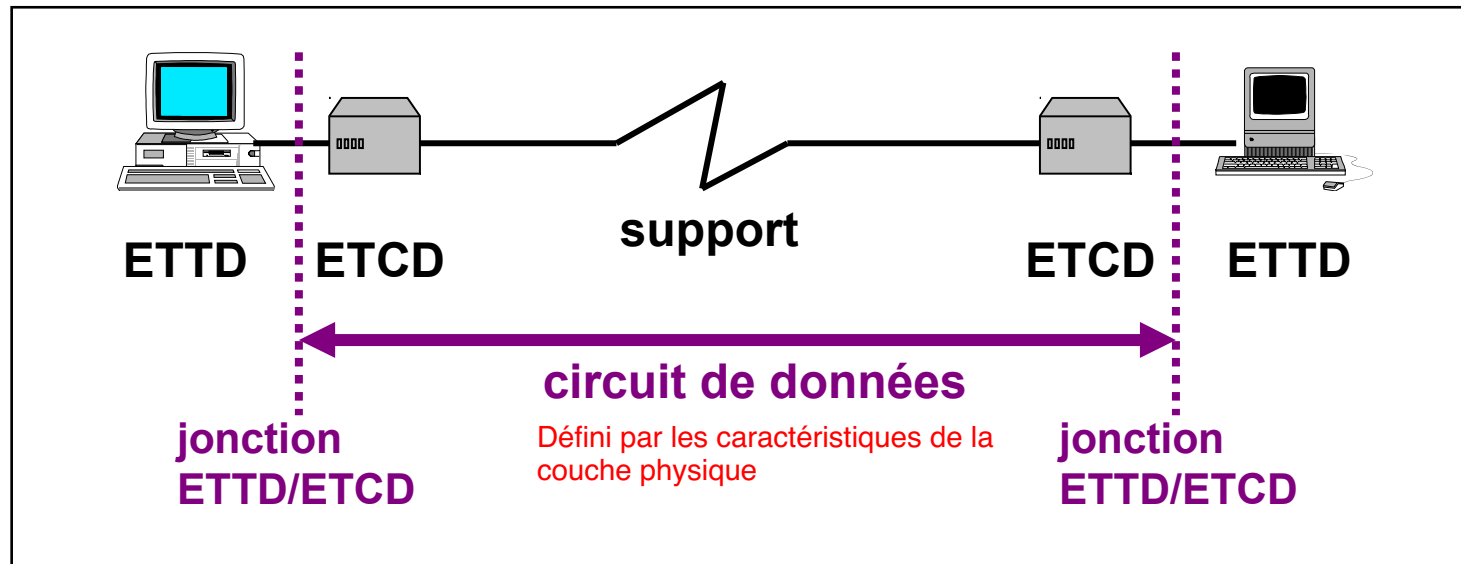
Quelques définitions



On veut s'assurer que pas de pb à l'envoi et que c'est bien reçu

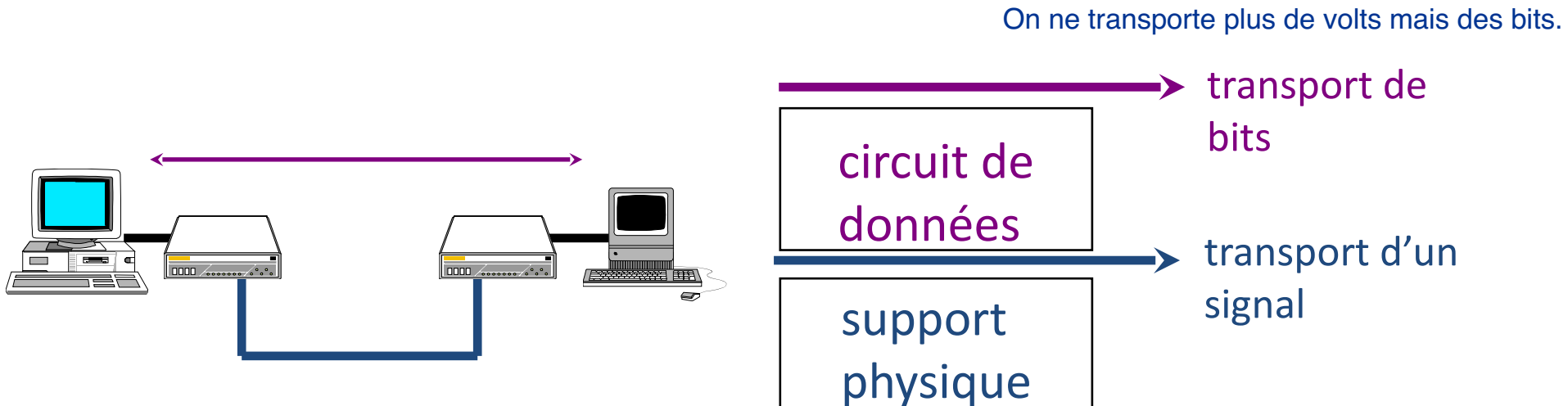
- **ETTD** (Equipement Terminal de Traitement de Données)
 - Effectue les traitements sur les données (ainsi que la supervision de la transmission de données)
 - Il s'agit d'une station (console, tposte de travail, etc.)
- **ETCD** (Equipement de Terminaison du Circuit de Données, en anglais, *DCTE, Data circuit-terminating equipment*)
 - Adapte le signal numérique délivré par l'ETTD aux caractéristiques du support (On envoie pas le même type de bit en fct du support physique)
 - Codage / décodage bande de base (codec)
 - Modulation / démodulation (modem)

Quelques définitions



- Le **circuit de données** est composé par l'ensemble des moyens (physiques) permettant le transport d'un signal de données
- La **jonction ETTD/ETCD** spécifie les caractéristiques de la connexion (interface) physique entre l'ETTD et l'ETCD

État de notre architecture



Pourquoi une liaison de données ?

- Le circuit de données permet d'émettre et/ou de recevoir des bits en série

mais avec :

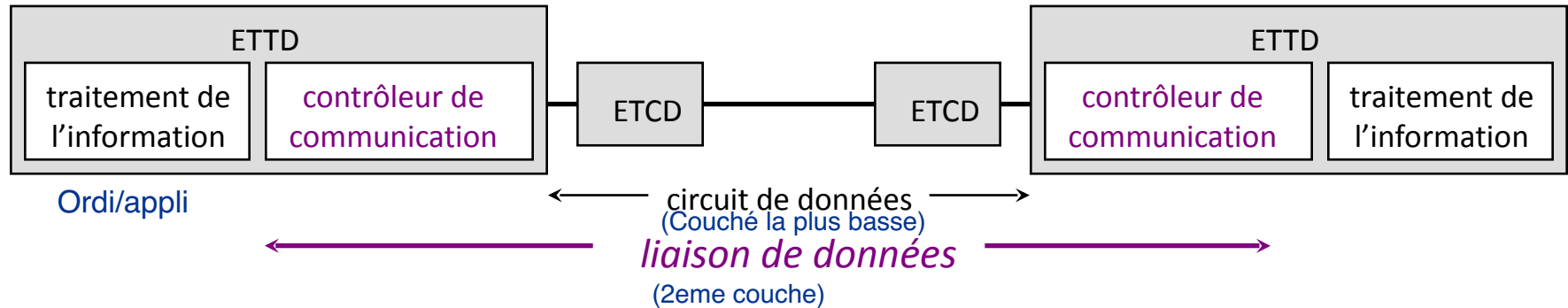
- Un certain débit ;
- Un certain délai ;
- Un certain taux d'erreurs.

En fct du type du support physique.

✚ C'est insuffisant !

✚ Ajout d'une interface logique. Pour s'adapter.

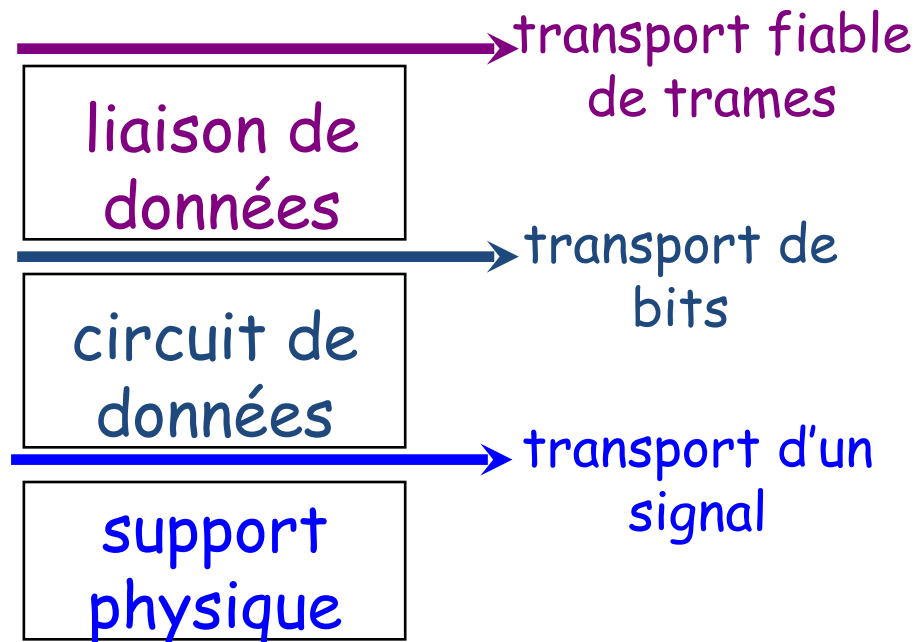
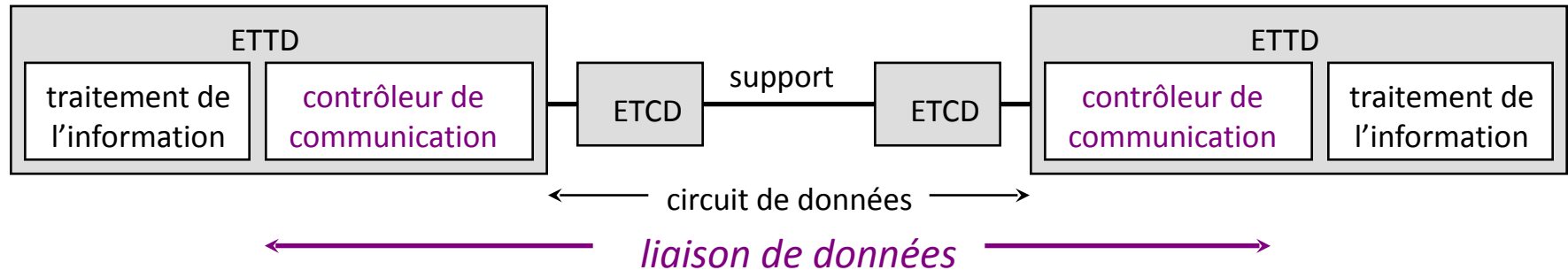
Qu'est-ce qu'une liaison de données ?



- Rôle : fiabiliser le transfert d'information entre 2 ETTD
- **Fiabilité :**
 - Pas d'erreur
 - Pas de perte
 - Pas de déséquencelement
 - Pas de duplication

Contrôleur de communication permet de vérifier qu'il n'y a pas d'erreur et les renvoie qd il y en a pour les régler.


Vue en couches



Services de liaison de données

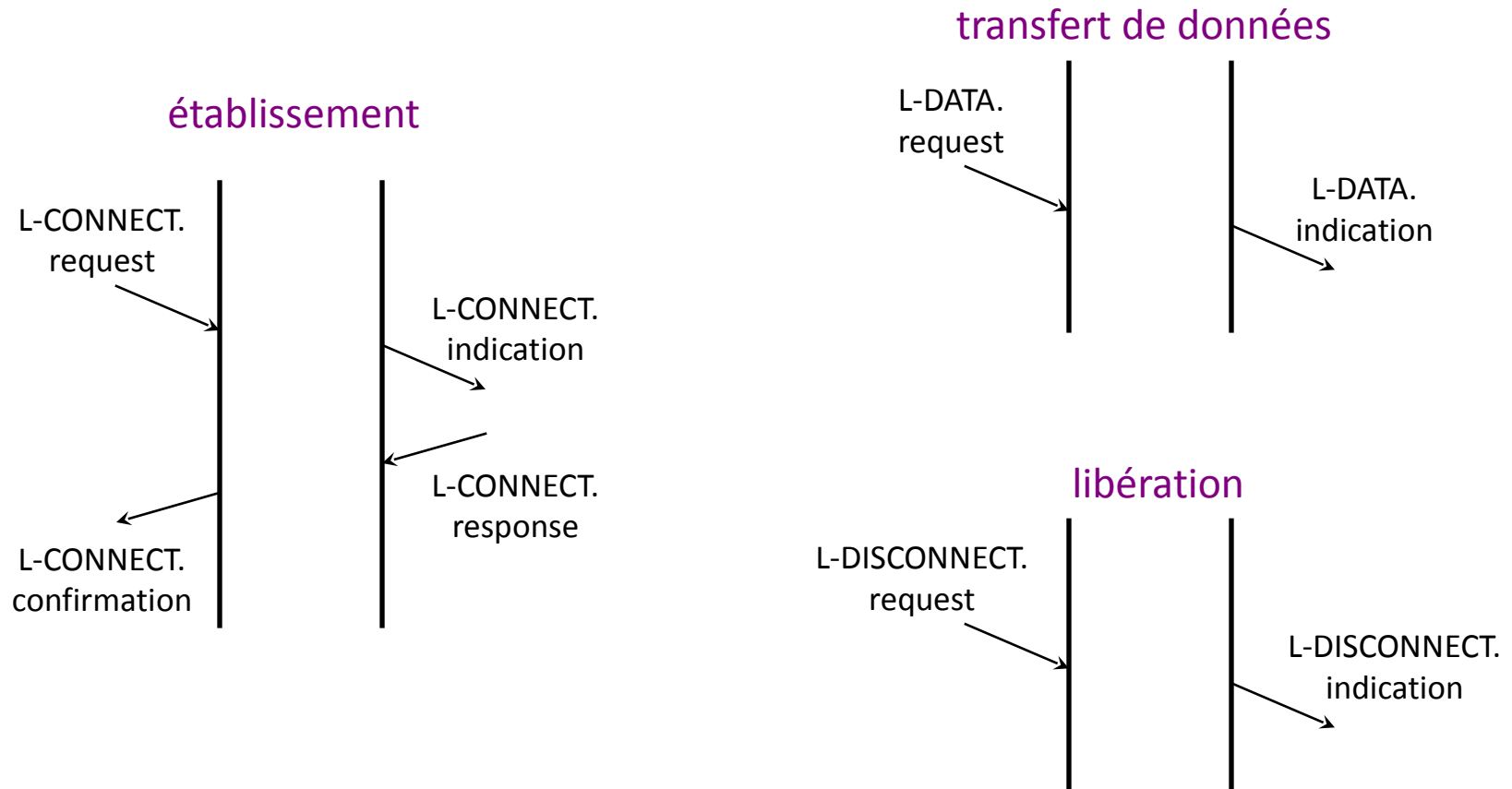
Service rendu par une couche = ce qui est donné par une couche à une couche supérieure
La couche liaison de données transmet ses données à la couche réseau

- Les services à utiliser
 - services offerts par le circuit de données
- Les services à rendre
 - les utilisateurs doivent avoir l'illusion d'un canal de transmission fiable
 - les utilisateurs doivent pouvoir établir et libérer logiquement une liaison au dessus d'un circuit

-  3 éléments de service
- établissement de liaison A la couche physique qui est envoyée à la couche réseau
 - transfert de données
 - libération de liaison

Représentation des services

Quand 2 terminaux (ordis) discutent., Il faut passer par toutes les couches.



Protocole de liaison de données

- une liaison de données : un canal physique capable de transmettre des bits, raccordant 2 (ou N) stations et leur permettant d'échanger de l'*information structurée en trames*
- un **protocole** de liaison de données : un *ensemble de règles* permettant de gérer la liaison (Ensemble de règles qui définissent la façon dont on construit une trame)
 - règles de **codage**
 - règles de **structuration**
 - règles **d'échange**
- le protocole met en œuvre un certain nombre de *mécanismes de communication*

Mécanismes de communication

- Problème de délimitation des trames

le récepteur doit savoir quand commence une trame et quand elle finit

- Idée

une trame doit commencer par un marqueur de début de trame et se finir par un marqueur de fin de trame

↳ **Fanion** de début/fin de trame = comment on marque le début et la fin d'une trame

Exemple de délimitation : 01111110

- Mécanisme de transparence
 - pour s'assurer qu'à l'intérieur de la trame, la configuration 01111110 ne soit pas interprétée comme un flag de fin
 - à l'émission : ajout d'un bit "0" après 5 "1" consécutifs
 - à la réception : destruction d'un bit "0" après 5 "1" consécutifs
- Émission d'une trame
 - construction de la trame
 - calcul des bits de contrôle
 - transparence (ajout)
- Réception d'une trame
 - transparence (retrait)
 - calcul des bits de contrôle
 - examen des champs de la trame

Exemple de délimitation : 01111110

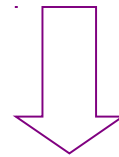
émission de

...1 1 0 1 1 0 1 1 1 1 0 1 0 1...

insertion de "0"

...1 1 0 1 1 0 1 1 1 1 1 **0** 0 1 0 1...

Pour dire qu'on est à la fin de la trame



On aura bien 1111101 pas interprété comme une fin de phrase grâce à ce mécanisme.

destruction de "0"

...1 1 0 1 1 0 1 1 1 1 1 ~~0~~ 0 1 0 1...

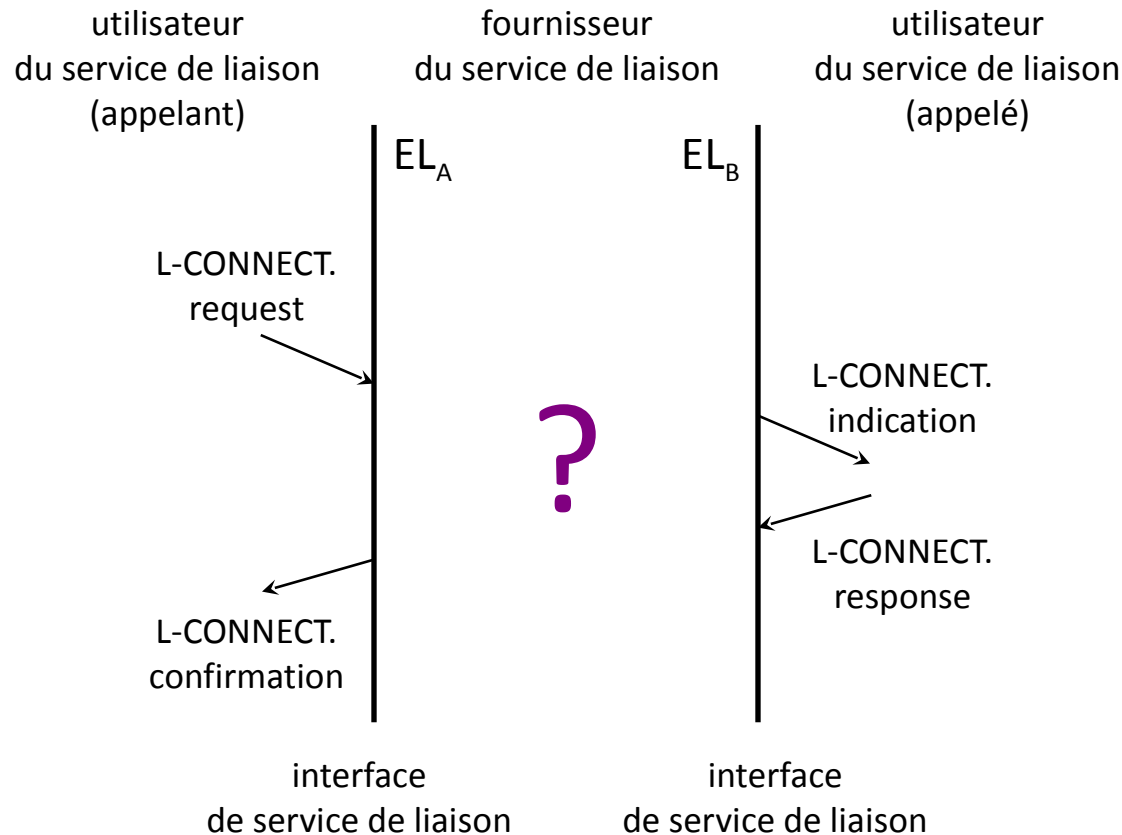
réception de

...1 1 0 1 1 0 1 1 1 1 1 0 1 0 1...

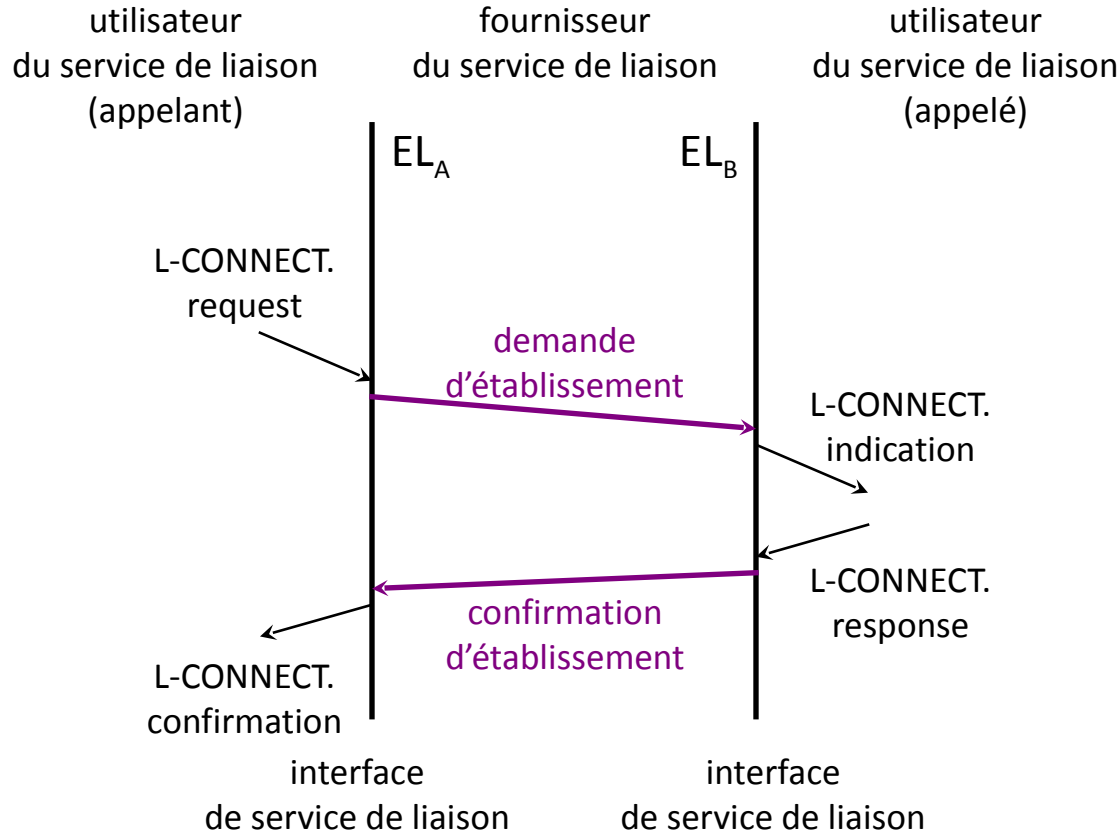
Chaque protocole a son type de fanion pour marquer le début et la fin de la trame.

Établissement de la liaison

Le circuit de données est préalablement établi entre A et B



Établissement de la liaison



Il faut définir 2 types de trames spécifiques :

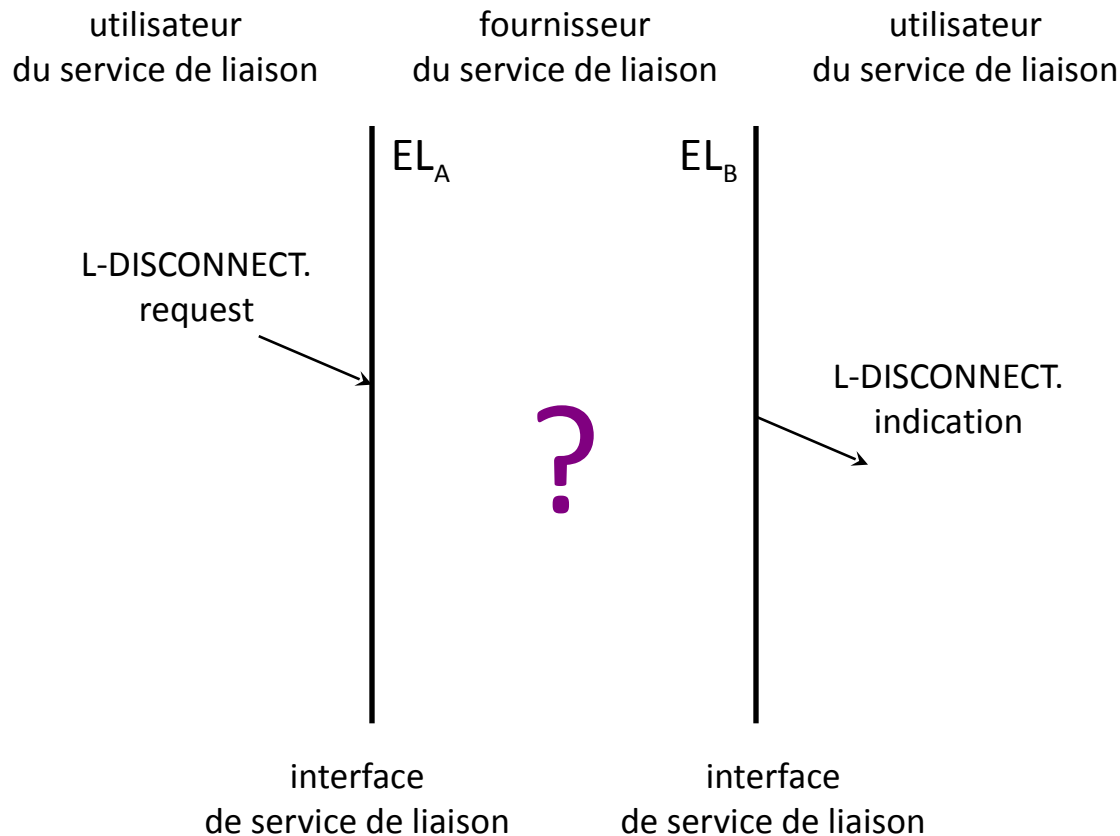
➤ Trame de **demande d'établissement**

(Ne contiennent pas données car juste pour faire la liaison)

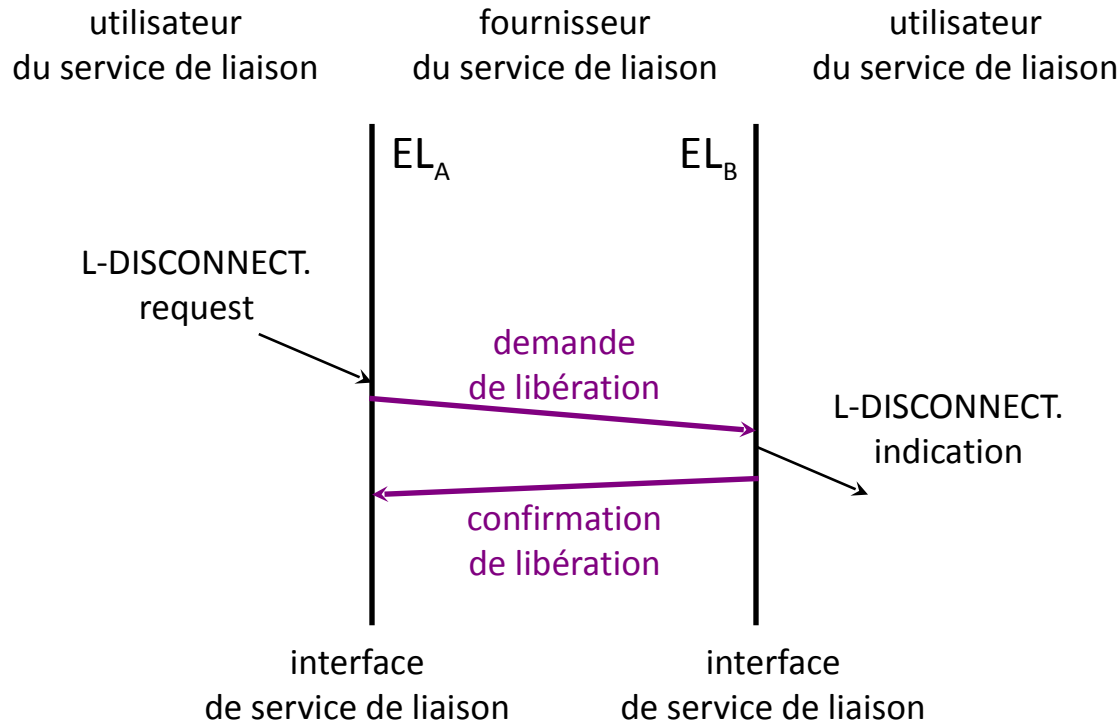
➤ Trame de **confirmation d'établissement**

Libération de la liaison

La liaison est en cours de transfert de données lorsque...



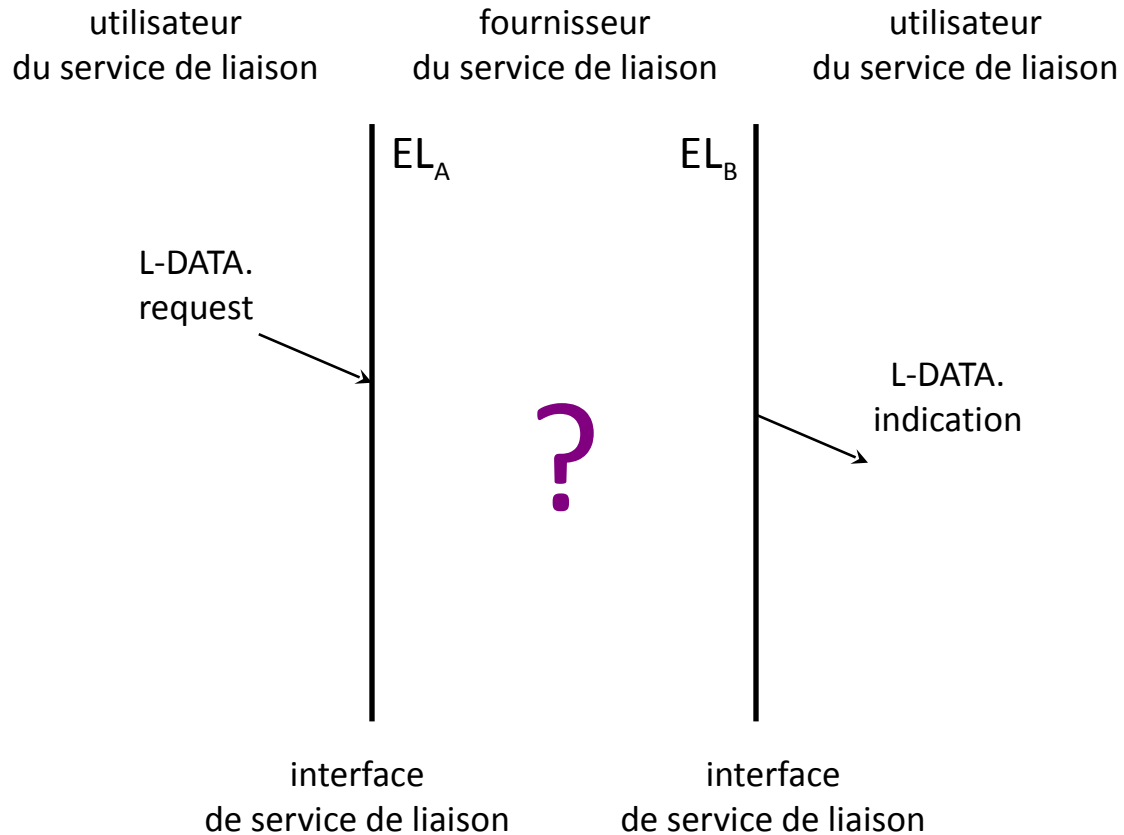
Libération de la liaison



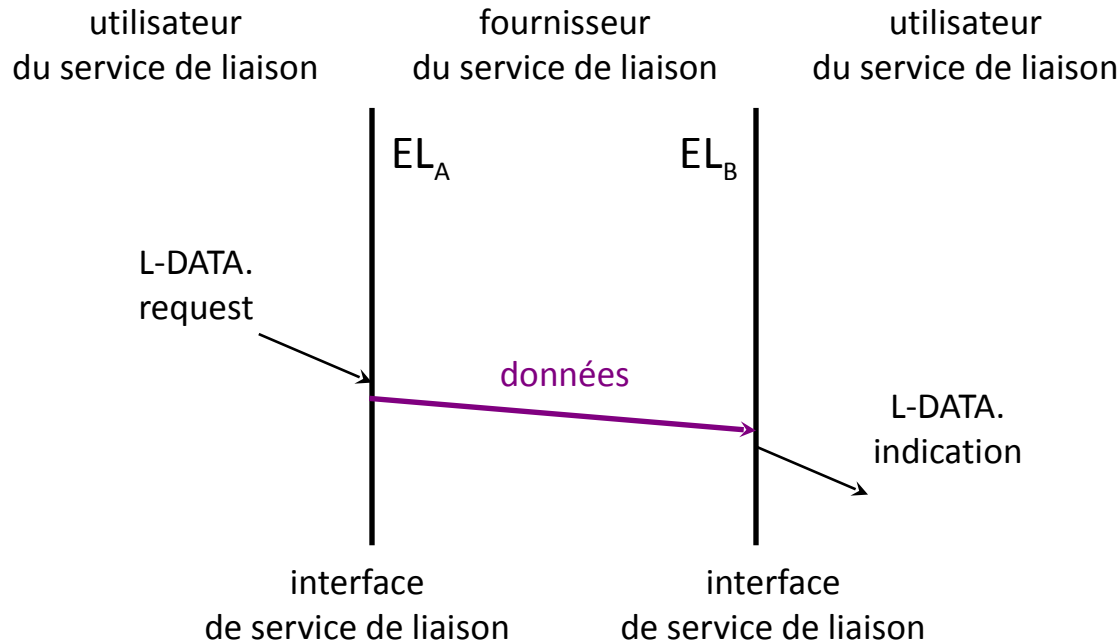
- Trame de demande de libération (Ne contiennent pas de données utiles)
- Trame de confirmation

Transfert de données

La liaison est établie



Transfert de données



⇒ Trame de **données** ou d'**information**

⇒ Est-ce suffisant ?

Non

Les problèmes possibles

- erreurs de transmission
- rupture du circuit de données
- pertes de trames
- débordement du récepteur
- panne d'un des ETTD
- panne d'un des ETCD
- etc.

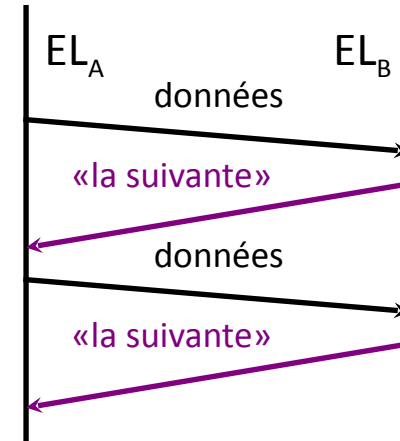
Contrôle de flux

- **Problème**

pertes de données possibles
si engorgement du
récepteur

- **Idée**

asservir le taux d'émission
de trames au taux
d'absorption du récepteur



↪ Trame « la suivante »

Protocole **Send-and-Wait**

On émet une donnée et on attend la réponse du récepteur comme
quoi elle est valide

Détection d'erreurs

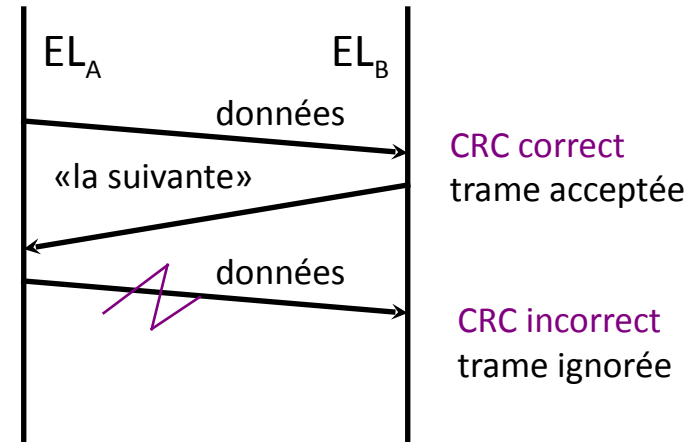
- **Problème**

Des erreurs de transmission peuvent altérer les données

- **Idée**

Introduire de l'information de contrôle permettant de détecter la présence d'erreurs de transmission dans une trame

J'envoie la donnée suivante que si la 1ère a bien été reçue, sinon je renvoie la 1ère.

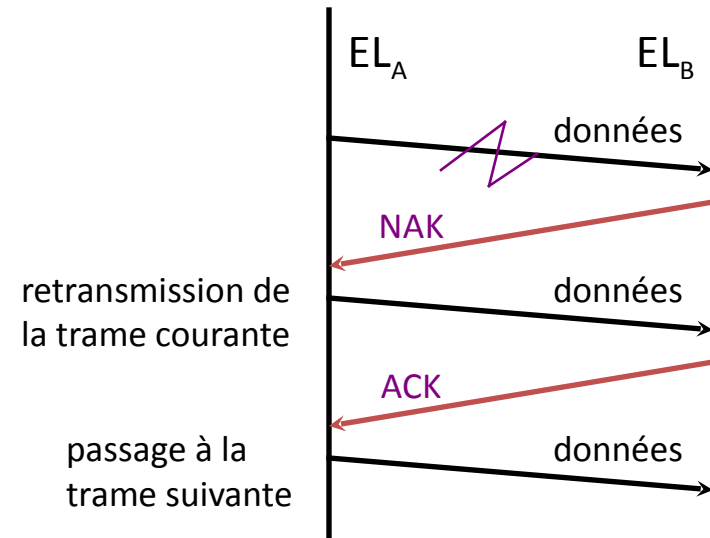


↪ champ de contrôle de type **CRC**

Consiste à rajouter à l'émission qq bits à la fin message correspondant à une opération « unique » (combinaison sur les bits du message). Donc à la réception, refais le même calcul pour retrouver le message.

Reprise sur erreur

- **Problème**
récupérer une trame de données en erreur
- **Idée**
introduire une trame de contrôle demandant la retransmission de la trame de données



✚ Trame de contrôle **NAK** Non acquiescement

✚ Trame de contrôle **ACK** (« la suivante ») Acquiescement

Rétention d'une copie

A l'émission, on enregistre la donnée qu'on envoie au cas où on aura besoin de la renvoyer.

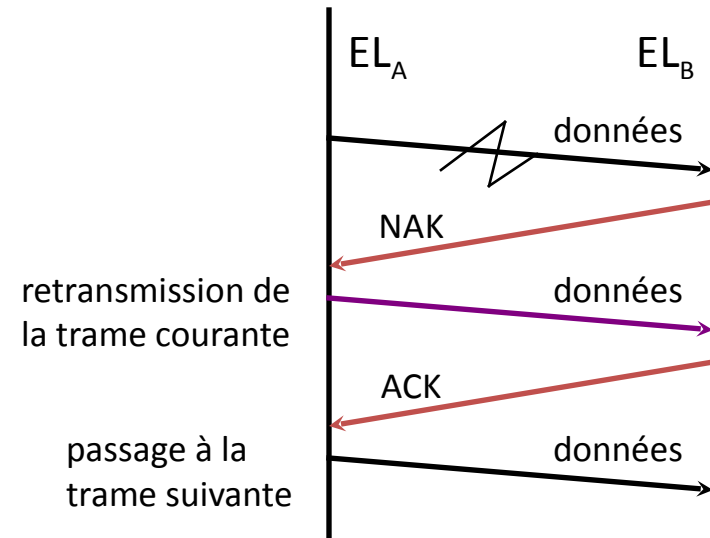
- **Problème**

pouvoir réémettre la dernière trame de données envoyée

- **Idée**

garder une copie de toute trame de données émise jusqu'à réception d'un acquittement positif

↪ **buffer** (tampon) d'émission

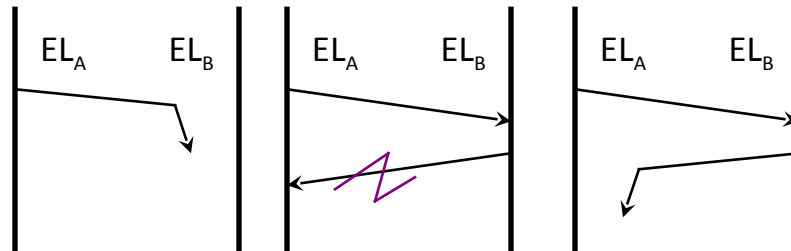


Temporisateur de retransmission

On rente td'envoyer la même donnée si je reçois pas de réponse de réception pendant un certain temps T_1 .

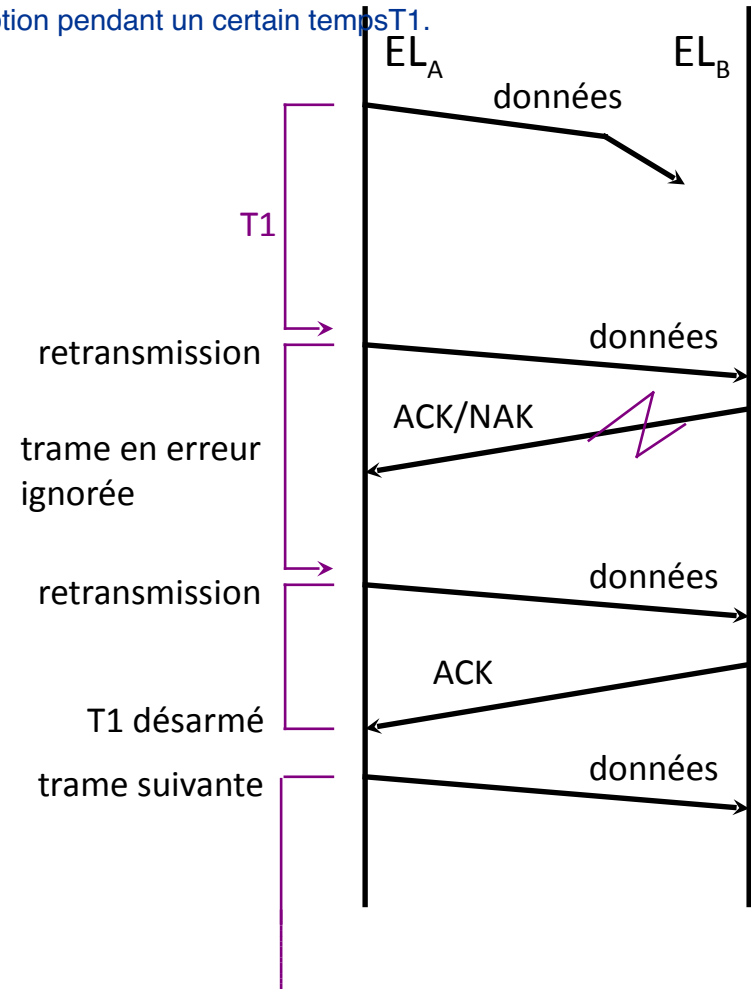
- **Problème**

pertes de trames possibles :
interblocages



- **Idée**

utiliser un mécanisme de temporisation limitant la durée d'attente d'une réponse



↪ temporisateur de retransmission T_1

Temporisateur de retransmission

- Dimensionnement de T1
 - T1 trop petit : retransmissions inutiles
 - T1 trop grand : reprise tardive
- Les trames NAK ne sont plus obligatoires
 - mais elles permettent d'accélérer la reprise

Nombre maximum de tentatives

- Problème

en cas d'incident grave du circuit de données,
retransmissions d'une même trame à l'infini

- Idée

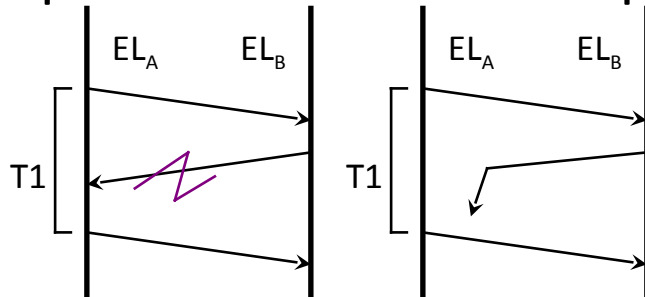
- limiter le nombre maximum de transmissions successives d'une même trame
- libérer éventuellement la liaison (et les ressources de communication associées)

↪ nombre maximum **N1** d'émissions d'une trame
De tentative

Numérotation des trames de données

- Problème

duplications de données possibles



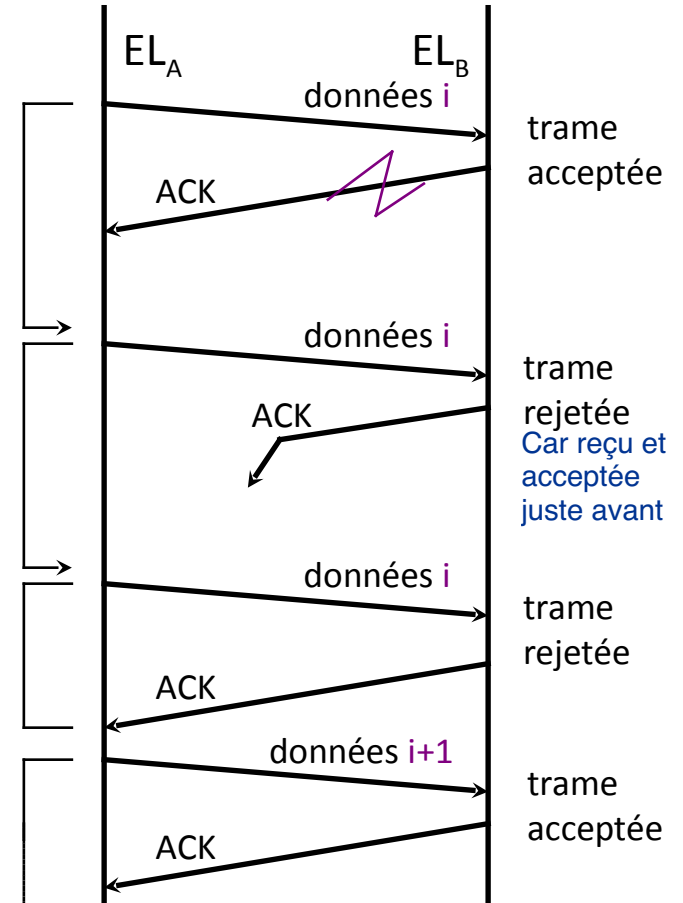
- Idée

utiliser dans la trame de données un champ de numérotation en séquence

↪ champ $N(S)$ pour les trames de données

- Numérotation modulo m Permet de repasser à 0

- Variable $V(S)$ en émission

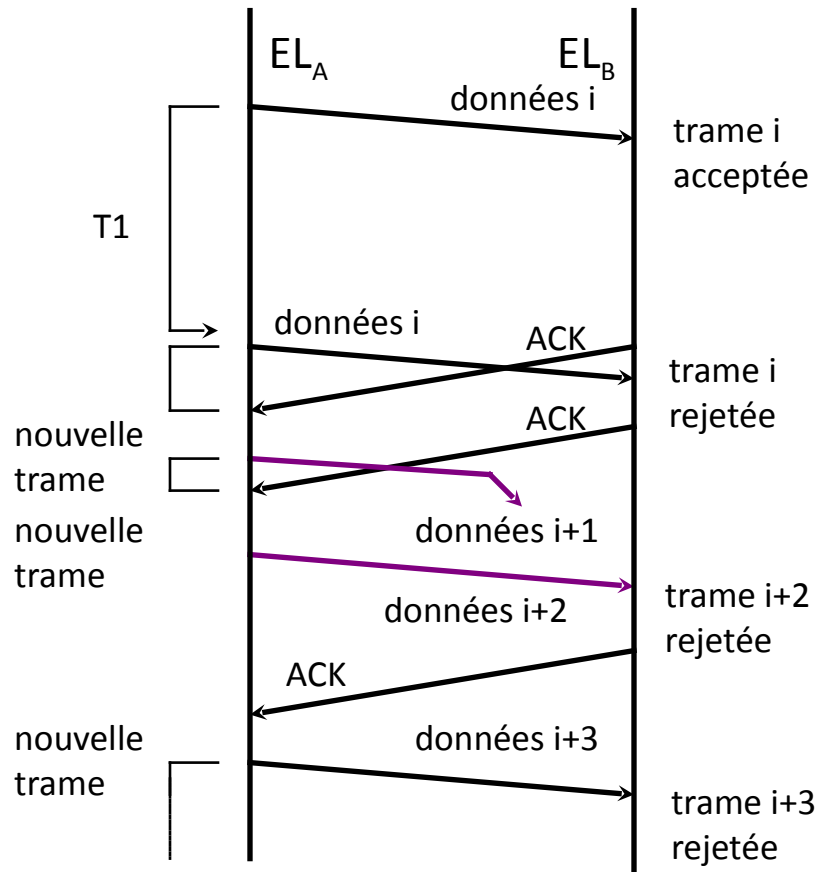


Il faut aussi numéroter les données en plus des réponses.

Numérotation des ACK

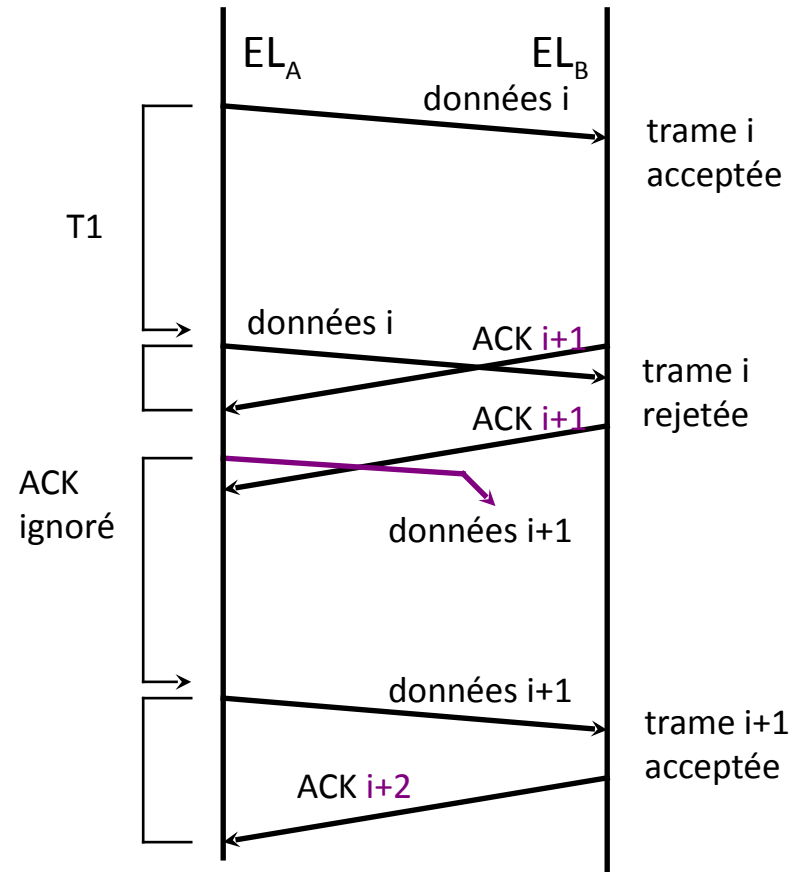
- Problème

pertes de données non
récupérables possibles



- Idée

utiliser dans la trame ACK un
champ de numérotation



Numérotation des ACK

- ↪ champ $N(R)$ pour les trames ACK
 - Numérotation modulo m
 - Variable $V(R)$ en réception

Détection d'inactivité

- **Problème**

ressources de communication bloquées inutilement

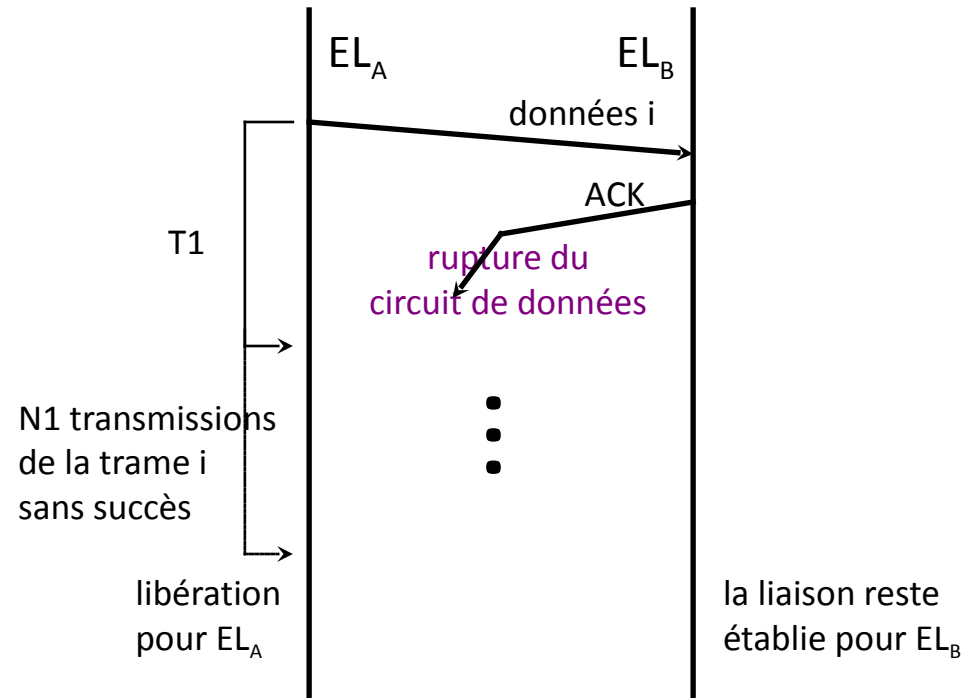
- **Idée**

utiliser un mécanisme de temporisation limitant la durée d'inactivité de la liaison

↳ temporisateur de détection d'inactivité *I*

- Dimensionnement

- Envoi de trames pour maintenir une activité

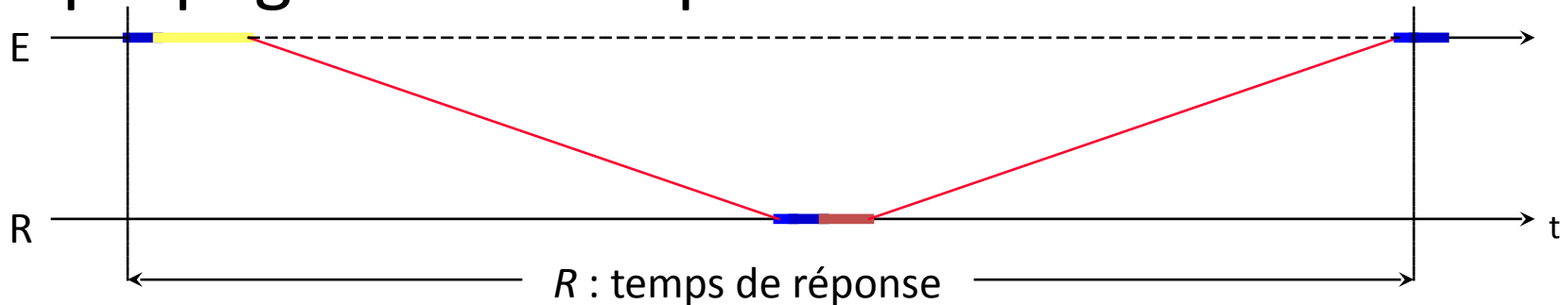


Au bout d'un certain temps et d'un certain nb de tentatives, si tjr pas d'activité, on libère le canal et on passe à la suite.

Anticipation

- Problème

mauvaise utilisation du circuit lorsque le temps de propagation est important



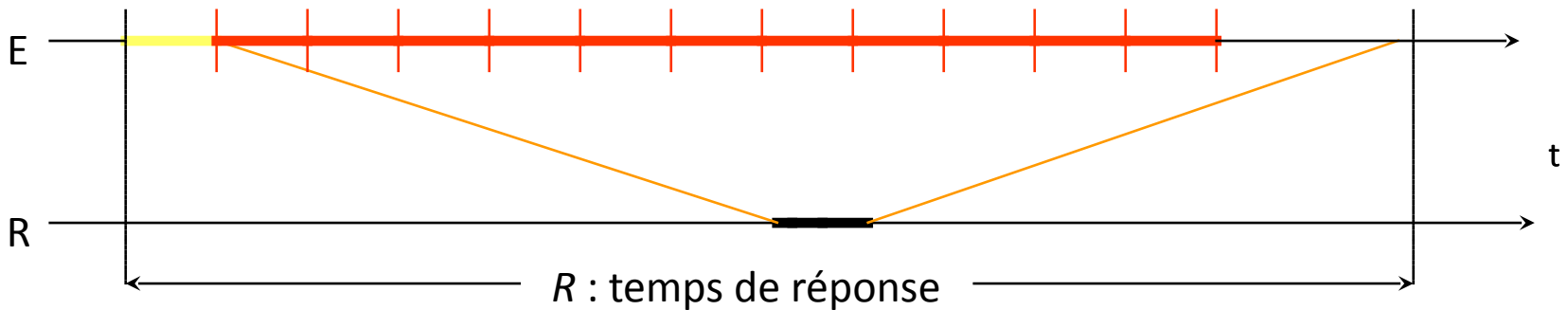
Débit de la connection trop lent. Donc on va anticiper l'envoi de la donnée et ne pas attendre la réponse

- T_{te} : temps de traitement en émission
- T_{td} : temps de transmission des données
- T_p : temps de propagation
- T_{tr} : temps de traitement en réception
- T_{ta} : temps de transmission de l'acquittement

Anticipation

- Idée

permettre à l'émetteur d'envoyer plusieurs trames consécutives avant de se bloquer en attente d'acquittement



On envoie plusieurs données le temps de recevoir la réponse à la première.

↪ fenêtre d'anticipation de taille W (Nb de trames de la fenêtre)

Anticipation

- Principe

- la fenêtre est une liste de W numéros de séquence
- l'émetteur est autorisé à envoyer les W trames de données dont le $N(S)$ est tel que :

$$\text{dernier } N(R) \text{ reçu} \leq N(S) \leq \text{dernier } N(R) \text{ reçu} + W - 1$$

- Fenêtre coulissante

ex: $w = 5$

$i = 3$

donnée $i+1$?

ACK: $N(R) = i+1$

$$i+1 \leq N(S) \leq i+1+w-1$$

$$4 \leq N(S) \leq 3+5-1=8$$

\Rightarrow on peut envoyer la 4^e, 5^e, 6^e, 7^e, et 8^e trame.

Anticipation

- Impacts sur les autres mécanismes

- acquittement positif : plusieurs stratégies

- un ACK par trame
 - un ACK par fenêtre, etc

- acquittement négatif : plusieurs stratégies

- rejet global
 - rejet sélectif

↪ Trame de **rejet global** De tte la fenetre (les W prochaines trames)

↪ Trame de **rejet sélectif**

- régulation de flux

↪ Trame « **stop** »

- pouvoir suspendre temporairement le flux de données
 - pouvoir reprendre le flux de données

Plus simple de faire rejet global mais moins optimisé

Bilan des mécanismes de communication

(Couche liaison de donnée)

- **Trames** Spécifiques de :
 - Demande d'établissement
 - Confirmation
 - Demande de libération
 - Données
 - Acquittement positif
 - Rejet global / sélectif
 - Stop
- **Temporisateurs**
 - Retransmission T1
 - Inactivité I
- **Paramètres**
 - Modulo de la num. m
 - Nb max de transm. N1
 - Taille de la fenêtre W

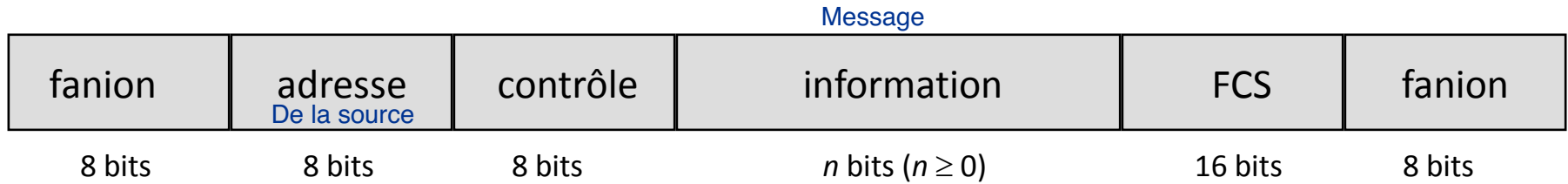
 Le protocole

Un exemple de protocole : HDLC

Pour la liaison de données

- « High-level Data Link Control »
- Norme ISO 13239
- Dérivé de SDLC développé par IBM
- Configuration point-à-point ou multipoint
- Exploitation en bidirectionnel à l'alternat ou simultané
- Fonctionnement en mode connecté
- Procédure orientée-bit
- Utilisation d'une fenêtre d'anticipation

Structure de la trame HDLC



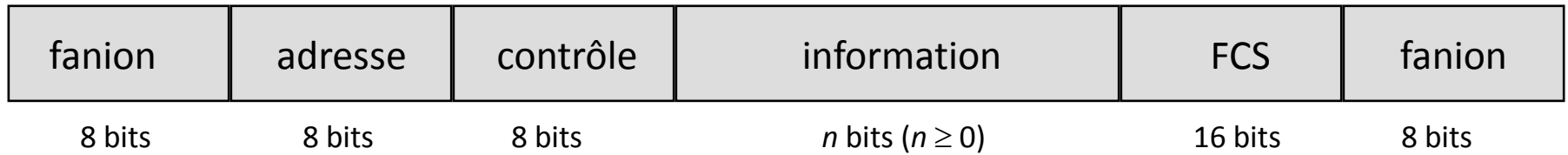
– Fanion : ‘01111110’

- Délimitation de la trame, début et fin
- Synchronisation
- Mécanisme de transparence : par insertion d’un ‘0’ après cinq ‘1’ consécutifs dans les données

– Adresse

- Identification de la station secondaire impliquée dans l’échange
- Trame de commande : la station secondaire destinataire
- Trame de réponse : la station secondaire émettrice

Structure de la trame HDLC



- **Contrôle**
 - Type de la trame
 - Numéros de séquence
 - Extension à 16 bits en modulo 128
- **Information**
 - Données de l'utilisateur
 - Toute trame de longueur inférieure à 6 octets est non valide
- **FCS (« Frame Check Sequence »)**
 - Calculé à partir d'un polynôme ($x^{16} + x^{12} + x^5 + 1$)

Situations d'anomalies

- Station temporairement occupée
 - Indication d'un état d'occupation
 - Indication d'un retour à la normale
- Erreur de transmission
 - Rejet des trames dont le FCS indique la présence d'erreurs
- Erreur de numéro de séquence
- Trame non valide

Protocole PPP

Ordi et box internet discutent avec le protocole PPP

- « Point to Point Protocol »
- Variante de HDLC
- Connexions Internet des particuliers (RTC, PPPoX)
- Fonctionnalités
 - Délimitation de trames
 - Détection d'erreurs
 - Négociation d'adresses
 - Authentification
 - Multi-protocole
 - Compression de données

Champ supplémentaire

- De deux octets
- Déterminant le protocole de niveau supérieur
 - 0x0021 : IPv4
 - 0x002B : IPX
 - 0x800F : IPv6
- Situé après le champ de contrôle du format HDLC