

## La couche Réseau Les Services

- Les services doivent être indépendants de la technologie utilisée dans le sous réseau,
- La couche Transport doit être indépendante des sous-réseaux présents,
- Les adresses de réseau qui sont à la disposition de la Couche Transport utilisent un plan uniforme de numérotation.

# La couche Réseau - Les Services orienté connexion

(connection oriented)

- n\_connect.request(appelé, appelant, confirmation réception, données exprès, qos, données utilisateur),
- n\_connect.indication(appelé, appelant, confirmation réception, données exprès, qos, données utilisateur),
- n\_connect.response(répondeur, confirmation réception, données exprès, qos, données utilisateur),
- n\_connect.confirmation(répondeur, confirmation réception, données exprès, qos, données utilisateur),
- n\_disconnect.request(origine, raison, données utilisateur, adresse en réponse),
- n\_disconnect.indication(origine, raison, données utilisateur, adresse en réponse),

- n\_data\_connect.request(données utilisateur),
- n\_data\_connect.indication(données utilisateur),
- n\_data\_connect.response(),
- n\_data\_connect.confirm(),
- n\_expedited\_data.request(données utilisateur),
- n\_expedited\_data.indication(données utilisateur),
- n\_reset.request(origine, raison),
- n\_reset.indication(origine, raison),
- n\_reset.response(),
- n\_reset.confirm(),

# La couche Réseau - Les Services orienté sans connexion

(connectionless)

- `n_unidata.request(adresse source, adresse destination, qos, données utilisateur)`,
- `n_unidata.indication(adresse source, adresse destination, qos, données utilisateur)`,
- `n_facility.request(qos)`,
- `n_facility.indication(adresse destination, qos, raison)`,
- `n_report.indication(adresse destination, qos, raison)`.

qos = qualité de service désirée,

raison = indique la raison de la commande.

## Interconnexion de réseaux

- couche 1 : les répéteurs copient les bits individuels entre des segments de câble.
- couche 2 : les ponts mémorisent et transmettent les trames entre réseaux locaux.
- couche 3 : les passerelles mémorisent et transmettent des paquets entre réseaux dissemblables.
- couche 4 : les convertisseurs de protocole fournissent une interface aux couches supérieures

## Deux types de Service

- mode « orienté connexion »
  - approche télécom
  - service minimum dans les hôtes
  - création de circuits avec garanties de services
- mode « sans connexion »
  - approche informatique
  - service minimum de la part du réseau
  - envoi et réception de paquets sans garanties

## Mode orienté connexions

- Communauté téléphonique
  - 100 ans d'expérience de réseaux non-informatique,
  - le réseau fait le maximum : le service sous-jacent est fiable
- mode « orienté connexion »
  - création d'une connexion préalable, maintenu pendant toute la communication et explicitement terminé,
  - négociation des paramètres de qualité de service à l'établissement
  - communication bidirectionnelle et ordonnée
  - contrôle de congestion

## Mode sans connexions

- Communauté Internet
  - 35 ans d'expérience des réseaux informatiques,
  - le réseau fait le minimum :
    - service sous-jacent non-fiable
    - hôte gérant la fiabilité (détection et correction des erreurs) et le contrôle de flux
- mode « sans connexion »
  - primitives : `SEND_PACKET` et `RECEIVE_PACKET`,
  - pas de fonctionnalités de réordonnement, contrôle de flux, ...



# Interconnexion de réseaux Ponts

Hypothèses :

- 802.3 : 1518 octets/trame 10 Mbit/s,
- 802.4 : 8191 octets/trame 10 Mbit/s,
- 802.5 : 5000 octets/trame 4 Mbit/s.

# Ponts

## Actions :

1. Reformater la trame et recalculer le môt de contrôle d'erreur,
2. Inverser l'ordre des bits,
3. Copier la priorité,
4. Générer une priorité fictive,
5. Annuler la priorité,
6. Vider l'anneau,
7. Positionner les bits A et C,
8. Problème de congestion (RLE rapide vers RLE lent),
9. Problème du jeton de transfert (ACK est reporté ou impossible),
10. Panique lorsque la trame est trop longue pour le RLE destinataire.

## Interconnexion de réseaux Ponts

|             | 802.3 (dest)      | 802.4 (dest)  | 802.5 (dest)      |
|-------------|-------------------|---------------|-------------------|
| 802.3 (src) |                   | 1, 4          | 1, 2, 4, 8        |
| 802.4 (src) | 1, 5, 8, 9, 10    | 9             | 1, 2, 3, 8, 9, 10 |
| 802.5 (src) | 1, 2, 5, 6, 7, 10 | 1, 2, 3, 6, 7 | 6, 7              |

# Interconnexion de réseaux

## Passerelles

- Passerelles en mode orienté connexion,  
Utilise le protocole X.75 qui construit un inter-réseau avec des sous circuits virtuels,
- Passerelles en mode non connecté,  
Dans ce système la passerelle injecte des datagrammes dans le sous-réseau ( C'est ainsi que fonctionne Internet ).

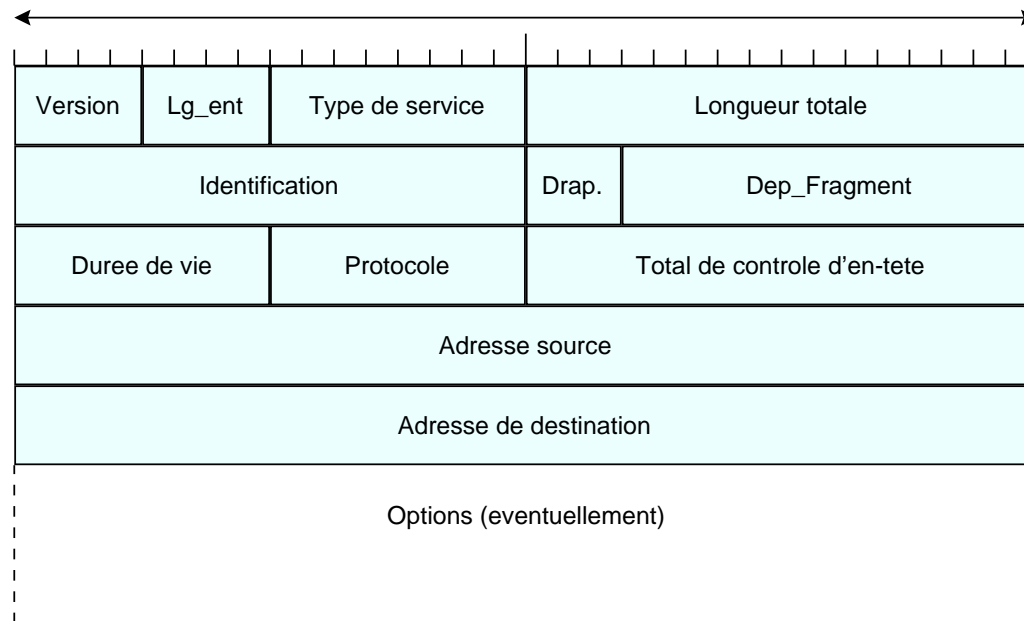
# Interconnexion de réseaux Passerelles

Problèmes (Taille max) :

- 802.4 : 65528 bits,
- X.25 : 32768 bits,
- Réseau radio par paquets ARPA : 2032 bits,
- ARPANET : 1008 bits,
- ALOHANET : 640 bits

La passerelle doit donc fragmenter ou défragmenter les trames.

# Le datagramme IP



# Propriétés des algorithmes de routage

- Exactitude (correctness)
- Simplicité (simplicity)
- Robustesse (robustness)
- Stabilité (stability)
- Équité (fairness)
- « Optimalité » (optimality)

## Routage Multichemin

On va affecter un poids a chaque chemin (à partir de D) :

|   |   |      |   |      |   |      |
|---|---|------|---|------|---|------|
| A | B | 0.80 | C | 0.15 | E | 0.05 |
| B | B | 0.90 | C | 0.05 | E | 0.05 |
| C | C | 0.95 | B | 0.04 | E | 0.01 |
| - | - | -    | - | -    | - | -    |



## Routage Hot Potato (pomme de terre brûlante)

C'est un algorithme simple de routage adaptatif :

Lorsqu'un paquet arrive on le met dans la file d'attente la plus courte.

(on peut aussi prendre en compte la longueur de la file d'attente et le poids du chemin)

# L'Algorithme du plus court chemin (Dijkstra)

