

# Cours réseaux: Liaison de données

Kamal Mehaoued

Département d'informatique, Université de Béjaia

2022/2023

## Plan du cours

- 1 Fonction de la couche liaison de données
  - Services fournis à la couche réseaux
  - Découpage en trames
  - Contrôle d'erreurs
  - Contrôle de flux
- 2 Détection et correction d'erreurs
  - Codes correcteurs d'erreurs
  - Codes détecteurs d'erreurs

# Fonction de la couche liaison de données

- La couche liaison de données utilise les services de la couche physique pour l'envoi et la réception de bits sur le canal.
- Elle fournit également à la couche qui lui est supérieure(réseau), un certain ensemble de services
- Parmi les fonction de cette couche :
  - ① Mettre à la disposition de la couche réseau une interface de services bien définie
  - ② Traitement des erreurs de transmission
  - ③ Régulation des flux de données

## Services fournis à la couche réseaux

Trois types de services sont offerts à la couche réseau :

- Service sans connexion et sans accusé de réception
- Service sans connexion avec accusé de réception
- Service orienté connexion avec accusé de réception

# Services fournis à la couche réseaux

## Le service sans connexion et sans accusé de réception.

- Aucune connexion logique est établie au préalable
- lorsqu'une machine source envoie une trame, elle n'attend aucun accusé de réception du destinataire.
- Si une trame est perdue, il n'y a aucun moyen d'y remédier (au moins au niveau de cette couche)
- Ethernet est un protocole qui offre ce type de service
- Ce type de service est très utile quand
  - 1 Le taux d'erreurs sur le canal est très faible
  - 2 On veut envoyer un trafic en temps réel tel que La voix ou la vidéo

# Services fournis à la couche réseaux

## Le service sans connexion avec accusé de réception

- Plus fiable que le service précédent.
- Aucune connexion logique n'est établie au préalable.
- Chaque trame envoyée est acquittée par le récepteur
- L'émetteur peut savoir, grâce à l'accusé de réception, si une trame a bien été reçue ou non.
- Si une trame n'est pas acquittée au bout d'un certain temps, l'émetteur la renvoie à nouveau.
- Ce service est intéressant dans le cas où le canal de communication est peu fiable
- Le 802.11 est un protocole offrant ce type de service

# Services fournis à la couche réseaux

## Le service orienté connexion avec accusé de réception

- Ce service est le plus fiable
- Une connexion est établie avant tout envoi de données
- Chaque trame envoyée est numérotée
- Chaque trame envoyée est reçue une et une seule fois
- Grâce à la numérotation, ce service garantit que toutes les trames seront reçues dans le bon ordre.
- Ce service avec connexion fournit à la couche réseau un canal fiable.
- Ce service est approprié dans le cas de :
  - ① De longs liens non fiable, comme un canal satellite
  - ② Un circuit téléphonique de longue distance

# Découpage en trames

- Une trame envoyée, n'est pas exempte d'erreurs à son arrivée au niveau de la couche liaison du destinataire :
  - La valeur d'un bit peut changer à la réception.
  - Le nombre de bits reçus peut être différents du nombre de bits émis
- L'une des fonctions de la couche liaison de données est de détecter ces erreurs et de les corriger éventuellement.
- Pour cela, la couche liaison de données découpe les flux de bits en trames(frames)
- Une information redondante appelée somme de contrôle est insérée dans chaque trame
- La couche liaison du récepteur recalcule cette somme
- si la somme recalculée est différente de celle émise, alors la couche liaison du destinataire peut conclure une erreur de transmission.



# Découpage en trames

Pour découper les flux de bits en trames, la couche liaison de l'émetteur utilise certaines méthodes :

- 1 Le comptage d'octets
- 2 Fanions de signalisation avec remplissage d'octets
- 3 Fanions de début et de fin de trames avec remplissage de bits

# Le comptage d'octets

- Chaque trame est décomposée en plusieurs octets
- La valeur du premier sert à indiquer le nombre d'octets contenus dans la trame
- En lisant la valeur du premier octet, la couche liaison du destinataire sait où se termine la trame pour passer à la suivante.

Nous pouvons illustrer cette technique dans les exemples suivants :

Cet exemple est correct

3	2	3	5	8	9	10	11	3	7	14	4	16	17	18	19
---	---	---	---	---	---	----	----	---	---	----	---	----	----	----	----

En revanche, l'exemple suivant ne l'est pas

5	2	3	4	5	6	7	8	9	10	11	3	7	14	5	16	18	19	1
---	---	---	---	---	---	---	---	---	----	----	---	---	----	---	----	----	----	---

- 1 Si la valeur du premier octet d'une trame est affectée au cours de la transmission, le récepteur perd la synchronisation et ne peut plus connaître le début de la trame suivante
- 2 Il pourra savoir que la trame reçue est erronée grâce à la somme de contrôle
- 3 Comme lui même ne peut pas savoir combien d'octets il faut remonter pour revenir au début de la transmission, donc il est impossible de demander à l'émetteur la transmission des données.
- 4 Ce qui fait que cette méthode est rarement utilisée.

# Fanions de signalisation avec remplissage d'octets

dans cette deuxième méthode :

- Chaque trame est délimitée au début et à la fin par des octets spéciaux appelés fanions de signalisation (flags)
- Souvent le même fanion sert de délimiteur de début et de fin d'une trame
- Deux fanions consécutifs désignent la fin d'une trame et le début d'une autre.
- Deux fanions consécutifs permettent au récepteur ayant perdu la synchronisation de trouver la fin de la trame courante et le début de la suivante
- La taille de la trame est toujours un multiple de 8 bits

# Fanions de signalisation avec remplissage d'octets

Et si le fanion de signalisation est lui-même une donnée ?

- Pour distinguer un fanion faisant partie de la donnée d'un fanion jouant le rôle d'un délimiteur, on insère un caractère spécial appelé Échappement (ESC) avant un fanion considéré comme donnée
- Un autre problème persiste lorsque le caractère ESC lui-même doit être envoyé comme donnée
- Dans ce cas, il faut le précéder également par le caractère spécial ESC.

## Exemple

Trame à transmettre 

Fan	A	ESC	B
-----	---	-----	---

Trame transmise 

Fan	ESC	Fan	A	ESC	ESC	B	Fan
-----	-----	-----	---	-----	-----	---	-----

# Fanions de signalisation avec remplissage de bits

- Avec cette méthode, la taille des trames peut être de taille quelconque de bits
- Chaque trame est délimitée par une combinaison spéciale (Fanion de signalisation) : 01111110
- Couche liaison émettrice : après chaque cinq 1 successifs dans trame à transmettre, un 0 est inséré.
- Couche liaison réceptrice : Tout 0 après cinq 1 sera supprimé.
- Le fanion 01111110 sera transmis comme donnée sous la forme : 01111101.

## Exemple

Trame à transmettre 0111011011110111111110

Trame transmise 011111100111011011110111110111001111110

# Problématique I

Supposons qu'un récepteur est capable de savoir si la trame reçue est correcte ou non :

- Comment informer l'émetteur sur l'état la transmission :
  - ① renvoyer un acquittement positif si la trame reçue est correcte
  - ② Renvoyer un acquittement négatif si la trame reçue est erronée
- Et si la trame ou l'acquittement est perdu à cause d'un problème matériel :
  - ① Si c'est la trame qui est perdue, le destinataire ne pourra pas agir car n'ayant reçu aucune trame
  - ② Si c'est l'acquittement qui est perdu, l'émetteur ne pourra pas agir car n'ayant aucune idée sur la trame transmise
- Pour résoudre ce problème la solution est d'utiliser un temporisateur au niveau de l'émetteur
  - ① L'émetteur fixe, grace au temporisateur, une certaine durée
  - ② la durée est largement suffisante pour l'envoi et le traitement de la trame ainsi que la réception de l'acquittement

# Problématique II

- ③ Si au bout de cette période l'émetteur ne reçoit pas d'acquittement, il renvoie la dernière trame émise.
- Si c'est l'acquittement qui est perdu :
  - ① La trame a bien été reçue par l'émetteur
  - ② l'émetteur risque de renvoyer plusieurs fois la même trame
  - ③ La couche liaison du récepteur risque de délivrer plusieurs fois la même trame à la couche réseau du destinataire
  - ④ Solution : Numérotation des trames

# Contrôle de flux

- Avoir un émetteur rapide d'un coté et un récepteur lent d'un autre.
- L'émission et la réception ne sont pas cadencées avec le même rythme
- Même avec une transmission sans erreurs, un récepteur lent sera inondé par un émetteur rapide
- Résultat : Perte de données
- Solution : Contrôle de flux
- Pour cela, il existe deux méthodes
  - ① Contrôle de flux avec retour d'informations (Feed-back based flow control)
  - ② Contrôle de flux basé sur le débit (Rate-based flow control) qui n'est pas un protocole de la couche liaison de données



# Contrôle de flux avec retour d'informations

## Feed-back based flow control

Il existe plusieurs variantes mais le principe est le même :

- L'émetteur ne doit pas envoyer des trames s'il ne reçoit pas d'autorisation au préalable de la part du récepteur
- Voici un exemple :

### Exemple

Le receveur peut envoyer le message suivant à l'émetteur : Envoie moi n trames et tu arrêtes l'émission jusqu'à ce que je te le demande à nouveau

# Contrôle de flux avec retour d'informations

## Feed-back based flow control

- Un ordinateur rapide envoie des trames à un ordinateur lent B
- le récepteur génère deux trames de supervision RR(Receiver Ready) et RNR(Receiver Not Ready)
- Les trames RR et RNR ne servent que pour le dialogue
- A chaque réception de trame, le mécanisme est comme suit :
  - ① Le récepteur envoie une trame RR s'il est prêt à recevoir encore d'autres trames
  - ② Le récepteur envoie une trame RNR s'il ne peut plus recevoir d'autres mais il enverra un RR dès qu'il peut en recevoir

# Détection et correction d'erreurs

- Deux approches utilisées pour le traitement des erreurs
- les deux approches se basent sur l'ajout d'informations redondantes aux données transmises
- La première stratégie
  - ① inclut suffisamment de redondance afin de permettre au récepteur de restituer les données réellement envoyées.
  - ② Elle utilise des codes correcteurs d'erreurs ECC(Error Correcting Code)
- La seconde stratégie :
  - ① Ajoute de la redondance, pour permettre au récepteur de déduire qu'il y a erreur et par conséquent demande le renvoi du bloc erroné
  - ② Utilise des codes détecteurs d'erreurs EDC(Error Detecting Code)

# Codes correcteurs d'erreurs

## Error Correcting Code (ECC)

Il existe 4 codes pour la correction des erreurs

- ① Les codes de Hamming
- ② Les codes binaires convolutionnels
- ③ Les codes de Reed-Solomon
- ④ Les codes de contrôle de parité à faible densité

# Error Correcting Code (ECC)

- 1 Une trame est composée de  $m$  bits de données et  $r$  bits redondants
- 2 Les  $r$  bits de contrôle sont déduits en fonction des  $m$  bits de la donnée
- 3 Soit  $n = m + r$  la longueur du bloc total
- 4 ce bloc est appelé un mot de code
- 5 Le rendement du code =  $m/n$

# Code de Hamming

- le code de Hamming permet de détecter et de corriger une erreur survenue dans un bloc transmis.
- Principe :
  - ① On fixe un entier  $k$
  - ② on code chaque bloc de  $m = 2^k - k - 1$  bits de données
  - ③ On ajoute  $k$  bits de correction, à certaines positions au bloc de  $m$  bits
  - ④ Pour obtenir un block de  $n = 2^k - 1$
  - ⑤ Le mot de  $n$  bits est appelé un mot de code

## Code de Hamming

## Tableau récapitulatif

$k$	$m = 2^k - k - 1$	$n = 2^k - 1$
$k = 3$	$m = 4$	$n = 7$
$k = 4$	$m = 11$	$n = 15$
$k = 5$	$m = 26$	$n = 31$

# Code de Hamming

## Positions des bits de correction

- les  $k$  bits de correction sont placés dans le bloc à envoyer aux positions d'indice puissance de 2 en comptant à partir de la gauche.
- En notant  $k_1k_2k_3$  les bits de correction
- En notant  $m_1m_2m_3m_4$  les bits de données
- Le bloc à envoyer devient  $k_1k_2m_1k_3m_2m_3m_4$



# Code de Hamming

## Calcul des bits de correction

- Le code de partié est calculé à partir d'une matrice de parité  $H$  telle que

$$H = \begin{vmatrix} 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{vmatrix}$$

- La colonne  $i$  de la matrice  $H$  représente en binaire la valeur de  $i$
- Les  $k$  bits de correction sont calculés en considérant le vecteur

$$A = \begin{vmatrix} k1 \\ k2 \\ m1 \\ k3 \\ m2 \\ m3 \\ m4 \end{vmatrix} \text{ tel que } H.A = \begin{vmatrix} 0 \\ 0 \\ 0 \end{vmatrix}$$

- Les trois équations permettant de calcul les bits de correction :
  - ①  $k1 + m1 + m2 + m4 = 0$
  - ②  $k2 + m1 + m3 + m4 = 0$
  - ③  $k3 + m2 + m3 + m4 = 0$

# Code de Hamming

## Réception des données et correction

- A la réception du bloc  $C = c_1c_2c_3c_4c_5c_6c_7$  éventuellement différent du bloc  $A$

- On calcul  $S = H.C = \begin{vmatrix} s_1 \\ s_2 \\ s_3 \end{vmatrix}$

- Si  $S = \begin{vmatrix} 0 \\ 0 \\ 0 \end{vmatrix}$  Pas d'erreur

- Si  $S \neq \begin{vmatrix} 0 \\ 0 \\ 0 \end{vmatrix}$  alors  $S$  est une colonne de la matrice de parité  $H$

dont l'indice nous renseigne sur la position du bit affecté dans la donnée reçue  $C$ .

# Code de Hamming

## Exemple

Considérons la trame suivante à envoyer 1 0 1 0 1 1 0 0 0 1 1

- 1 Établir la matrice de parité
- 2 Calculer le code de Hamming.
- 3 Quel est le bloc envoyé

Considérons le bloc reçu suivant 1 0 1 0 1 1 0 1 0 1 1 :

- 1 Vérifier s'il est correct
- 2 Corriger s'il y a des erreurs

# Codes détecteurs d'erreurs

- Un code robuste de détection d'erreurs et le plus utilisé est le CRC(Cyclic Redundancy Check)
- Ce code est également appelé code polynomial
- Dans ce genre de codes, les chaînes de bits sont considérés comme les coefficients d'un polynôme
- Un bloc de  $n$  bits est vu comme un polynome comprenant  $n$  termes de  $x^{n-1}$  à  $x^0$
- un tel polynôme est de degrés  $n - 1$
- Le bit de poids fort est le coefficient de  $x^{n-1}$  et ainsi de suite

# Codes détecteurs d'erreurs

## Exemple

la chaîne 110001 sera représentée par le polynôme  $x^5 + x^4 + 1$

- Pour utiliser un code polynomial, Le récepteur et l'émetteur se mettent d'accord sur un polynôme générateur  $G(x)$
- Le générateur doit avoir le bit de poids fort et le bit de poids faible à 1

# Codes détecteurs d'erreurs

- Pour calculer le CRC d'une trame de  $m$  bits correspondant au polynôme  $M(x)$ , il faut que la trame soit plus longue que le polynôme générateur
- Le principe est de concaténer un CRC à la fin de la trame de sorte que la trame résultante soit divisible par  $G(x)$
- Quant le récepteur reçoit une trame, il la divise par  $G(x)$ , si le reste de la division n'est pas nul alors il y a erreur de transmission.

# Codes détecteurs d'erreurs

- 1 Soit  $r$  le degrés de  $G(x)$ , ajouter  $r$  bits à 0 après le bit de poids faible de la trame, la trame résultante contient ainsi  $m + r$  bits elle correspond au polynome  $x^r M(x)$
- 2 Effectuer la division du polynome  $x^r M(x)$  par  $G(x)$
- 3 Soustraire le reste de la division de la chaine de bits correspondant au polynome  $x^r M(x)$ .
- 4 Le résultat de cette opération et la trame envoyée au destinataire. Appelons  $T(x)$  le polynôme correspondant

# Codes détecteurs d'erreurs

Exemple 1 : Supposons que nous voulons envoyer la trame  
1101011111 avec un générateur de code 10011

$$\begin{array}{r}
 1\ 1\ 0\ 1\ 0\ 1\ 1\ 1\ 1\ 1\ 0\ 0\ 0\ 0 \\
 1\ 0\ 0\ 1\ 1 \\
 0\ 1\ 0\ 0\ 1\ 1 \\
 \quad 1\ 0\ 0\ 1\ 1 \\
 \quad\quad 0\ 0\ 0\ 0\ 0\ 1\ 1\ 1\ 1\ 0 \\
 \quad\quad\quad 1\ 0\ 0\ 1\ 1 \\
 \quad\quad\quad\quad 0\ 1\ 1\ 0\ 1\ 0 \\
 \quad\quad\quad\quad\quad 1\ 0\ 0\ 1\ 1 \\
 \quad\quad\quad\quad\quad\quad 0\ 1\ 0\ 0\ 1\ 0 \\
 \quad\quad\quad\quad\quad\quad\quad 1\ 0\ 0\ 1\ 1 \\
 \quad\quad\quad\quad\quad\quad\quad\quad 0\ 0\ 0\ 0\ 1\ 0 \\
 1\ 1\ 0\ 1\ 0\ 1\ 1\ 1\ 1\ 1\ 1\ 0\ 0\ 1\ 0 \\
 T(x) =
 \end{array}$$



## Codes détecteurs d'erreurs

```

1 1 0 1 0 1 1 1 1 1 0 0 1 0
1 0 0 1 1
0 1 0 0 1 1
  1 0 0 1 1
    0 0 0 0 0 1 1 1 1 0
      1 0 0 1 1
        0 1 1 0 1 0
          1 0 0 1 1
            0 1 0 0 1 1
              1 0 0 1 1
                0 0 0 0 0 0

```

T(x) transmis est correcte

# Codes détecteurs d'erreurs

Exemple 2 : Trame : 1010010111 Code : 10111 Quelle est la trame envoyée. Expliquer comment au niveau du récepteur, on pourra savoir que la trame est reçue correctement.