

CM d'interconnexion

Premier CM

Dans ce premier CM, on a vu des généralités sur l'interconnexion de réseaux, surtout le modèle OSI et les normes ISO dedans (en anglais c'est l'inverse).

Deuxième CM et, plus

En dessous de la couche transport → tout va bien mais au dessus de la couche transport → tout va mal. Standard aussi met en place standard sur métrique et sur les délais → usine à gaz infâme.

TCP

TCP = fiabilité (sans contrainte sur le délai)

Couche transport (IP) = on sait pas le chemin et en plus variance de temps sur le même chemin.

Plusieurs chemins avec des caractéristiques différentes et qui évoluent dans le temps. Donc couche de transport doit regarder le chemin le plus avantageux.

Protocole = prouver formellement (protocole 0 de OSI)

UDP

UDP = pas fiable mais très rapide.

Couche Session = point de synchronisation à différents points du transfert pour sauvegarder le temps de transfert (mais pas implanté dans OSI, laissé aux applications).

Couche présentation = se comprendre entre les PC (par exemple le codage des entiers sur 4 bits ou 8 bits) mais aussi codage sans perte, chiffrement. Mais pas implanté aussi.

Donc on passe de 7 à 5 couches.

couche appli = déjà pré-coder de base et on spécifie juste (Un truc bien dans le modèle OSI de base).

1 service = 1 ou plusieurs protocoles.

Protocole propriétaire = danger car si on veut implanter un truc on est obligé d'utiliser les machines de la même marque.

Protocole de routage = envoi d'info sur les routeurs pour savoir comment envoyer les paquets. Donc c'est applicatif ≠ algorithme de routage.

écart de caractéristiques = on peut pas tout faire en même temps. base de données = pas de perte de données, mais pas de rapidité. → (Merci Benjamin)

Problème des réseaux locaux = partage de ressources (beaucoup de collisions possibles) → Objectif des standards = éviter les collisions.

MAC = couche liaison de données + couche physique un peu. Car par exemple CDMA = écouter le support.

Monde des réseaux de Télécom

- Besoin de circuit
- mais besoin de différentes choses, donc plusieurs piles protocolaires (par exemple pendant l'appel pas de délai mais à l'établissement de l'appel pas besoin de gigas nul)

Si on sait pas comment on appelle une unité de données, on l'appelle un ...PDU (par exemple PHY-PDU, MAC-PDU, etc).

Différence entre routage et commutation = en routage on établit un chemin, en commutation on dirige de l'entrée vers la sortie.

Si tout était conforme au modèle aussi on aurait pas de gros problèmes d'interco

Interconnexion

On regarde où ça diverge et on met une passerelle entre ces deux endroits (dans le modèle OSI).

Problème, connecté vs non connecté.

- Non connecté avec non connecté = Simple
- Connecté avec connecté = un peu plus complexe
- Connecté avec non connecté = complexe
- Non connecté avec connecté = très complexe

Si on a deux visions de QOS, on discute ou OSEF ? Si on veut une garantie de service ? Notion de débit ?

Le plus souvent on fait rien.

1. Exemple 1 : réseau téléphonique

- Interconnexion par traduction au niveau applicatif

2. Exemple 2 : X.25 sur frame relay

- interco par encapsulation
- dans frame relay on a des tuyaux permanents
- C'est très fiable mais on tombe en panne beaucoup.

3. X.25 sur TCP

- On utilise TCP pour X.25
- On fait le contrôle de congestion plus bas que TCP car contrôle de congestion chez TCP.
- On met un protocole entre X.25 et TCP, c'est le protocole XOT. De niveau 5 (applicatif) et X.25 est de niveau 3 (réseau).
- Problème :
 - fiabilité = OK
 - Adressage = on a juste une table de correspondance.
 - numéro de connexion = gérer l'adressage

Troisième CM

On lit le numéro de téléphone du destinataire et on sait la destination. Ce qui va compter dans la QOS c'est le temps de propagation.

HDLC = avoir vie dure.

Q.931 = protocole de signalisation pour établir une connexion. (applicatif)

Q.921 = protocole de liaison de données (couche 2)

Si GSM alors Q.931 mais on doit faire une passerelle pour passer dans le réseau.

Voix sur IP

on envoie la parole dans les paquets IP. Mais on a des problèmes de délai et de gigue. car TCP pas fait pour ça.

comment on attape de la gigue ? On met des buffers en gros (on garde des paquets et on les lit en même temps). pour les signalisations on utilise TCP ou UDP.

Les opérateurs Télécom seront content quand la téléphonie sera morte (Car deux réseaux à opérer)

multicast = trouver un arbre de diffusion optimal mais c'est chiant.

On met une estampille temporelle dans les paquets pour savoir quand on les a reçus et avoir une idée du délai.

j'abandonne ici, ça va trop vite et le prof a dit je cite "*J'ai plus de temps je vais accélérer*"