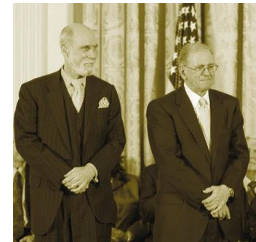


NSI – 1ere	COURS Séquence 7A : Transmission de données dans un réseau.	LFV
------------	--	-----

**Vinton Cerf** (1943-) et **Robert Kahn** (1938-) ont inventé, au début des années 1970, le protocole de transmission de paquets de données IP (*Internet Protocol*) et le protocole de contrôle de flux de données TCP (*Transmission Control Protocol*). Il s'agit des deux principaux protocoles du réseau Internet. Ils donnent à ce réseau sa fiabilité, sa robustesse en cas de pannes ou de modifications et sa capacité à évoluer. Cela a valu à leurs auteurs le surnom de « pères d'Internet. »



## I. Qu'est-ce qu'un réseau ?

La principale fonction de tout réseau, qu'il soit informatique ou autre, est de relier des objets identiques en utilisant un ensemble de règles garantissant un service fiable.

- Réseau postal (adresse bien formatée, poids et taille des colis etc.)
- Réseau routier (respect du code de la route, panneaux de direction conformes etc.)
- Réseau de trains (respect des horaires, des voies, des aiguillages etc.)

Les réseaux informatiques respectent eux aussi des règles de base :

- Plusieurs ordinateurs doivent être capables de s'identifier à travers le réseau.
- Le réseau doit être capable de déterminer la destination des informations à transmettre.
- L'information doit être restituée avec fiabilité, sans altération des données.

## II. Les composantes d'un réseau informatique.

### A. Les équipements réseau.

Certains matériels n'ont pas besoin d'être en réseau pour fonctionner : c'est le cas des ordinateurs et des photocopieuses numériques. Ces équipements peuvent cependant être mis en réseau afin qu'ils communiquent ensemble. Dans ce cas :

- Une station de travail **client** est un ordinateur sur lequel travaille un utilisateur.
- **Un serveur est un ordinateur dont des ressources sont partagées avec les autres ordinateurs.**
- **On parle d'architecture client-serveur lorsque deux programmes situés sur deux machines différentes cherchent à communiquer à travers un réseau.**

D'autres matériels sont spécifiquement conçus pour travailler en réseau :

- Le câblage, bien que n'étant pas un équipement à proprement parler, est un élément important à prendre en compte : il doit répondre à des normes très strictes pour que le réseau fonctionne correctement.
- **Un commutateur (switch) est une multiprise "intelligente" qui relie ensemble les machines d'un même réseau.** Il va trier les informations qui lui parviennent en prenant soin de ne les transmettre qu'à leur destinataire.
- **Un routeur permet de relier plusieurs réseaux ensemble.** Il s'agit de matériels intelligents qui connaissent les meilleurs chemins que les données devront emprunter pour aller d'un point A à un point B et ils apprennent en permanence de nouvelles routes.



## B. Le réseau d'un point de vue logiciel (ou logique)

Les différents équipements constituant un réseau sont équipés de logiciels leur permettant d'utiliser ou de gérer le réseau physique. **Les protocoles réseau sont des règles communes de fonctionnement partagées par les machines d'un réseau.** Ils sont mis en œuvre par des logiciels et sont nécessaires aux communications sur un réseau. Ce sont des conventions et des traitements sur les données que les ordinateurs utilisent pour dialoguer entre eux.

Dans un réseau informatique, les protocoles sont organisés en piles et :

- lors de l'expédition les protocoles «du haut» transfèrent les messages aux protocoles «du bas»
- lors de la réception, les protocoles «du bas» transfèrent les messages aux protocoles «du haut»
- les machines intermédiaires (commutateurs, routeurs) n'ont pas besoin d'effectuer tous les protocoles.

C'est un point important du cours sur lequel nous allons nous attarder dans la suite.

Le réseau logique peut également inclure d'autres éléments logiciels, par exemple les VLAN qui permettent de découper un gros réseau en plusieurs petits sous-réseaux logiques (on dit *virtuels* d'où le V de VLAN).

## III. Les protocoles réseau : modèle TCP/IP

### *Couche 1 : couche liaison (Protocole Ethernet)*

**Cette couche est dédiée à la gestion d'un seul réseau : des machines connectées entre elles par un commutateur.**

Elle est composée de deux parties : physique et liaison.

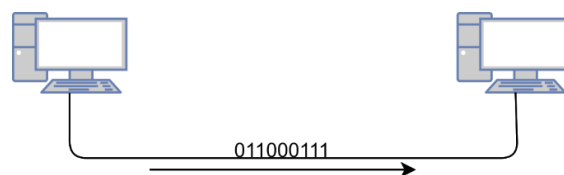
Le protocole de la partie physique s'occupe de la transmission des bits de façon brute sur un canal de communication. Cette couche normalise les caractéristiques électriques, optiques ou radios (par exemple le nombre de bits par seconde) ainsi que les caractéristiques mécaniques (forme des connecteurs, type de câbles...).

*La partie liaison contient, en simplifiant, le protocole qui gère la communication entre plusieurs machines d'un même réseau reliées entre elles par un commutateur.*

Le protocole le plus utilisé est le protocole Ethernet (qui gère d'ailleurs également la partie physique). Il gère pour cela :

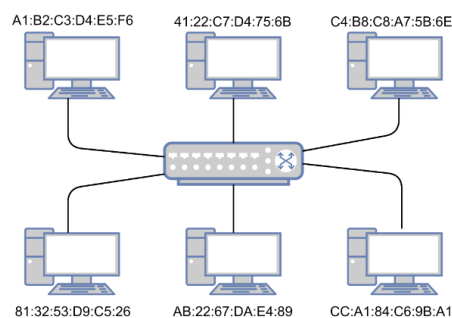
- L'adressage physique des machines grâce aux adresses matérielles (adresses dites MAC).
- L'évitement des collisions entre paquets de données.
- La commutation par paquets (un canal n'est occupé que pendant le temps durant lequel le paquet circule).

**Remarque 1 :** le protocole Ethernet gère à la fois la partie physique et la partie liaison. Il se décline en variantes selon le matériel utilisé (optique, câble, radio). Sur la planète, c'est LE protocole majoritaire de la couche liaison même si d'autres protocoles existent (en avionique, pour le cœur de réseau internet etc)



La partie physique gère, entre deux machines :

- type de signal (optique, radio, électrique)
- paramètres de conversion du signal en bits :
  - combien de bits par seconde ?
  - amplitude et longueur d'onde éventuelles?
  - etc



La partie liaison (par ex. protocole Ethernet) gère, entre plusieurs machines reliées entre elles par un commutateur :

- l'adressage physique (@MAC)
- l'évitement des collisions
- la commutation par paquets (un canal n'est occupé que pendant le temps où le paquet circule)

*Remarque 2* : les adresses MAC comportent six octets et sont traditionnellement notées en hexadécimal (chaque octet de 00 à FF). Elles ne peuvent pas être modifiées et sont attachées au *matériel* (d'où leur nom d'adresses *matérielles*). Ci-dessus on voit que c'est le bazar avec les adresses MAC : il n'y a aucun lien logique entre les adresses MAC d'un même réseau.

Au contraire, les adresses IP (voir plus bas) peuvent être modifiées car elles sont *logicielles* : en particulier on peut les attribuer d'une façon *logique* qui nous arrange : c'est beaucoup moins le bazar avec les IP !

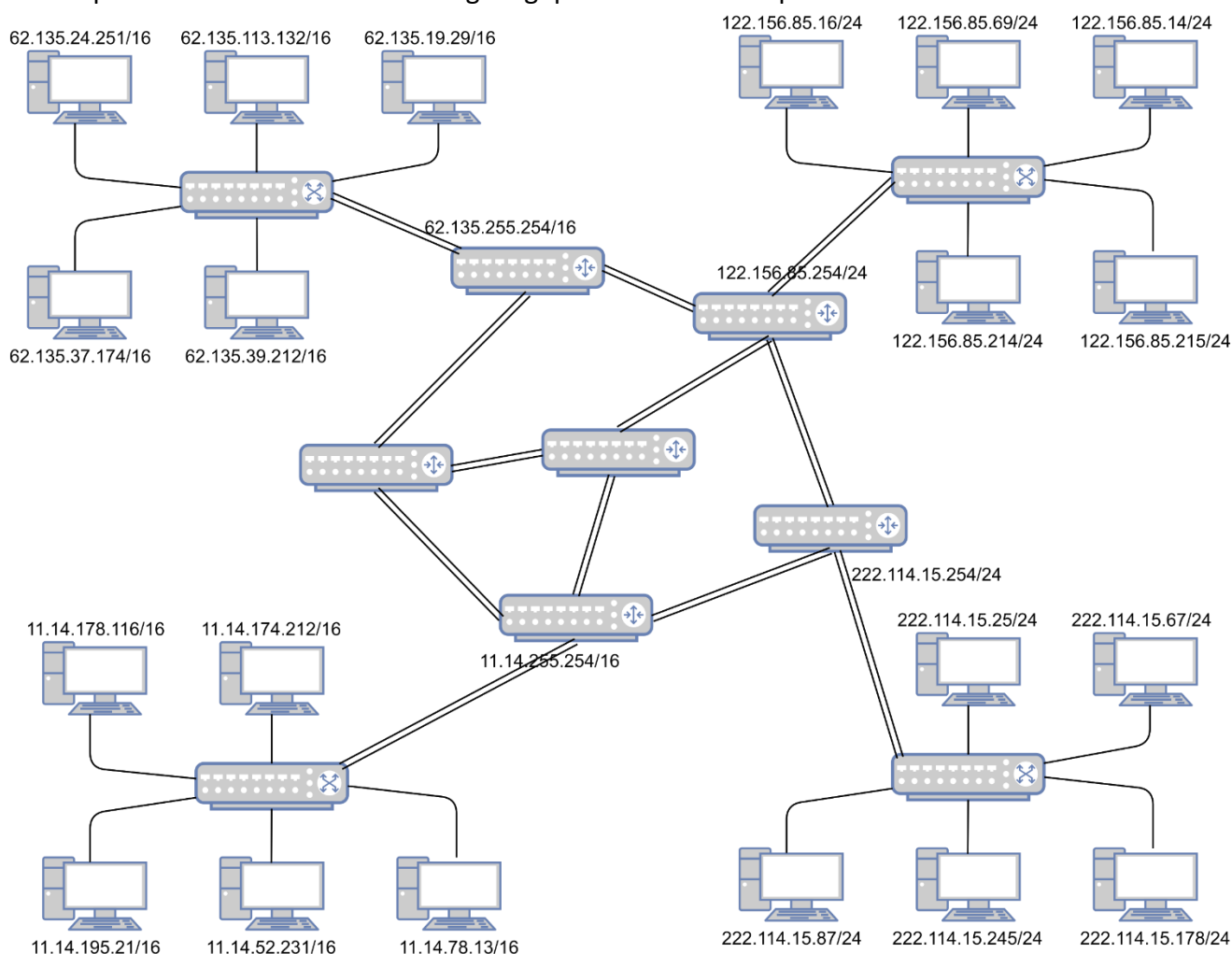
En résumé : le protocole Ethernet de la couche liaison gère les communications entre plusieurs machines d'un même réseau reliées par un commutateur. Il gère les caractéristiques physiques de la communication, gère les adresses physiques des machines sur le réseau (@MAC matérielles composées de six octets notés en hexadécimal) et gère l'évitement des collisions (grâce à la commutation par paquets).

## Couche 2 : couche Internet (Protocole IP)

Cette couche est dédiée à la gestion des réseaux connectés entre eux par des routeurs.

Sur le schéma, remarquer la différence de symbole entre les commutateurs  et les routeurs .

Elle fournit pour cela un schéma d'adressage logique en mettant en place les adresses IP.



Ces adresses de 32 bits = 4 octets notés en décimal (dans le cas d'Ipv4) se découpent en deux morceaux :

- le netid (partie gauche) qui est le numéro identifiant le réseau auquel appartient la machine,
- le hostid (partie droite) qui est le numéro identifiant la machine sur son réseau.

Le nombre de bits consacrés au netid est donné après le slash (en notation CIDR) et correspond à la notion de masque de réseau (le masque 11111111 00000000 00000000 00000000 attribue les huit premiers bits au netid). Sur le schéma ci-dessus on constate que toutes les machines situées sur un même réseau ont le même netid.

Par ailleurs, c'est bien parce que ces adresses sont logicielles (voir remarque 2 sur la couche 1 évoquant le bazar des adresses MAC), qu'elles peuvent être attribuées de façon logique et ordonnée.

Enfin, les protocoles de cette couche internet gèrent également le routage des paquets sur le réseau : le protocole IP s'occupe donc de faire transiter un paquet de l'expéditeur au destinataire en passant par tous les routeurs nécessaires (il permet de trouver la bonne route, le bon chemin entre les deux machines en utilisant les tables de routage disponibles dans les routeurs : les protocoles permettant de construire les tables de routage seront étudiés en terminale).

Pour fonctionner la couche Internet a bien entendu besoin de la couche liaison puisque les paquets vont traverser de nombreux réseaux.

*Remarque* : sur le schéma ci-dessus, il faut rappeler que chaque câble entre deux routeurs constitue un réseau à lui tout seul (un réseau constitué de seulement deux interfaces : une à chaque bout du câble). Ainsi la couche liaison est sollicitée sur le réseau de la machine de départ, sur le réseau de la machine d'arrivée, mais aussi à chaque saut entre deux routeurs !

En résumé : le protocole IP de la couche Internet gère l'adressage logique des machines appartenant à des réseaux connectés entre eux avec des adresses IP. En IPv4 les adresses sont constituées de 4 octets notés en décimal, la partie gauche indique le numéro de réseau (netid) et la partie droite indique le numéro de la machine sur le réseau (hostid). Le protocole internet assure également le routage des paquets à travers les différents routeurs. Pour effectuer la transmission entre deux machines, ce protocole a besoin d'utiliser la couche liaison.

### *Couche 3 : couche transport (Protocoles TCP et UDP)*

**Cette couche est dédiée au découpage des paquets, à l'identification des programmes expéditeurs et destinataires et, éventuellement, à la détection de pertes de paquets.**

Pour l'instant nous avons vu que le protocole IP de la couche Internet est capable de faire transiter un paquet d'une machine A vers une machine B sur Internet.

Au cours de ce transit, dès que le paquet circule sur un nouveau réseau, il faut faire appel à la couche liaison de ce réseau (c-à-d au protocole Ethernet ou autre protocole de liaison pour le cœur de réseau).

On peut alors avoir du mal à comprendre ce que vont être les couches supplémentaires situées au-dessus de la couche internet. Il faut pour cela se placer du point de vue d'une application (par exemple votre navigateur internet ou l'application de votre réseau social favori). Prenons l'exemple où cette application (le programme client) effectue une requête pour que le programme serveur lui envoie le contenu d'une image.

Tout d'abord, les données binaires de cette image – avant d'être envoyées à la couche internet via le protocole IP – vont être découpées en de nombreux petits paquets. Cela permet d'éviter la congestion du réseau : il est plus facile de *partager* des canaux de communication en y faisant circuler des petits paquets qu'en y faisant circuler de rares gros tas de données (*voir images animées gif fournies avec le cours*).

L'utilisation de petits paquets est d'ailleurs ce qui a assuré au protocole Ethernet de couche 1 son succès (grâce à la commutation de paquets qui permettait des architectures de câblage très souples et efficaces contrairement à la commutation de circuits).

Cette découpe en paquets permet également de mieux gérer les éventuelles erreurs de transmission (il y a alors un seul petit paquet à renvoyer et pas tout un gros tas de données). Bien entendu, lors de la réception des paquets, il faut prendre soin de les recombinaer entre eux et non pas les découper !

Par ailleurs, il y a un autre point important à régler : la répartition des paquets aux différentes applications. Lorsque sur une même machine deux applications A et B reçoivent chacune des informations, il faut être

capable - pour réassembler les paquets qui arrivent sur l'adresse IP - de savoir à quelle application est destinée tel ou tel paquet.

Pour cela, la couche transport du client attribue à chaque demande de connexion effectuée par une application un numéro de port source (aléatoire) qui est communiqué au serveur. En retour, le serveur enverra les paquets en spécifiant bien comme port de destination le port source qui a effectué la demande ce qui permettra d'associer le paquet à la bonne requête correspondante.

De façon symétrique, le programme serveur écoute également les requêtes sur un numéro de port bien défini (et non plus aléatoire) car plusieurs programmes serveurs peuvent être en écoute sur une même adresse IP. Il faut donc qu'une application cliente effectue ses requêtes sur le bon port (par exemple le port d'écoute d'un serveur Minecraft est 25565, le port d'écoute d'un serveur http est 80, d'un serveur https 443 etc.) : hors de question de demander à jouer à Minecraft à un serveur de messagerie mail !

Les deux protocoles UDP et TCP assurent donc ces deux fonctions :

- découpe en paquets
- fléchage des paquets vers la bonne application cliente ou vers le bon programme serveur grâce aux numéros de ports attribués aléatoirement (clients) ou pas (serveurs).

Enfin, le protocole TCP assure une troisième fonctionnalité : il assure la fiabilité de la transmission en proposant un protocole de détection de perte de paquets. Ainsi avec TCP on est certain de bien recevoir tous les paquets. Le protocole UDP ne propose pas cette fonctionnalité, il est donc plus rapide et est adapté par exemple pour du transport de vidéos en streaming ou la perte d'un paquet n'est pas dramatique.

*En résumé : Il y a un travail de mise en paquets à effectuer entre la couche Application (couche 4, tout en haut) et la couche Internet (couche 2). Ce découpage des informations en tout petits paquets est nécessaire pour pouvoir optimiser le trafic de données sur les réseaux et éviter les congestions (par exemple le protocole Ethernet de la couche liaison (couche 1) fonctionne par commutation de paquets). C'est la couche Transport (couche 3) qui gère cette mise en paquets grâce aux protocoles TCP ou UDP.*

*La couche Transport gère également la délivrance des paquets vers le bon programme informatique de la machine en ajoutant aux paquets des informations appelées numéros de port. Ces numéros de port correspondent en quelque sorte aux identifiants des différents programmes qui se connectent.*

*Enfin, TCP (mais pas UDP) assure une troisième fonctionnalité : il permet d'assurer la fiabilité de la transmission en effectuant un contrôle de perte de paquets. Si un paquet est perdu, le protocole TCP va permettre de demander à ce que ce paquet soit ré-expédié.*

#### ***Couche 4 : couche application (Protocoles HTTP, HTTPS, FTP, SMTP, MQTT, sur mesure ...)***

Les protocoles de la couche application gèrent la façon dont une application dialogue avec le programme serveur qu'elle souhaite contacter. Pour les comprendre il suffit de se dire que chaque application qui effectue des communications sur internet doit utiliser un protocole spécifique pour dialoguer avec le programme côté serveur : on comprend aisément qu'un jeu Minecraft ne va pas utiliser les mêmes conventions pour communiquer qu'un logiciel de messagerie web ou qu'un navigateur web. Certains protocoles sont utilisés par beaucoup de clients et plusieurs applications concurrentes les utilisent (http et https utilisés par tous les navigateurs internet, smtp pour les messageries de mél, mqtt pour les objets connectés etc.). D'autres protocoles de la couche application ne sont utilisés que par une application bien spécifique et ont été élaborés par un seul fabricant (protocole Minecraft, protocole de votre application de réseautage etc.).

En revanche tous ces protocoles utiliseront les protocoles TCP ou UDP pour communiquer.

Par exemple en langage Python, il est très facile de créer son propre protocole de communication entre deux programmes en utilisant les sockets qui vont permettre de s'appuyer sur TCP.

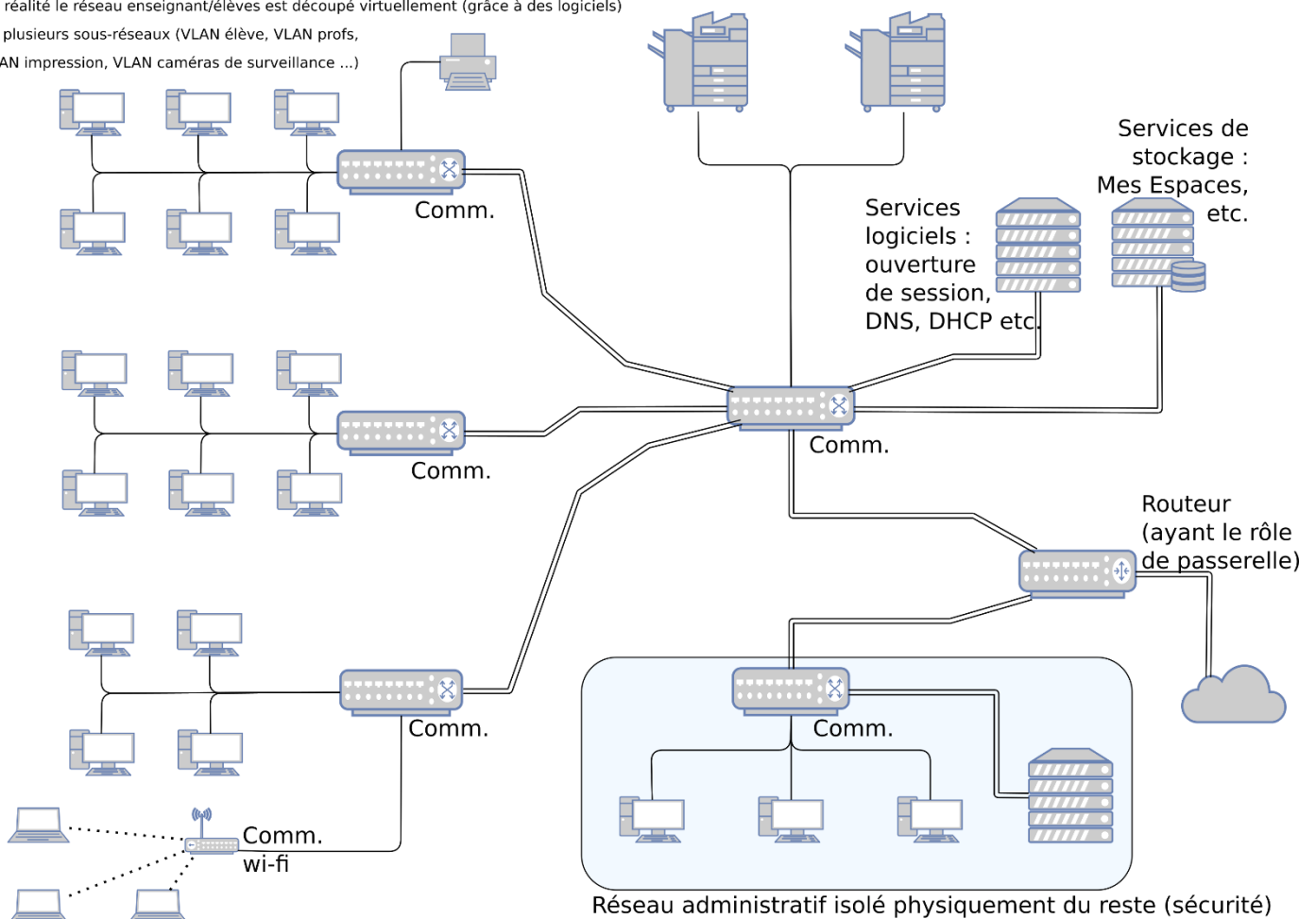
En résumé : la couche application met en œuvre les protocoles qui sont propres à l'application souhaitant communiquer à travers internet avec un programme serveur dédié. Ces protocoles sont parfois utilisés par plusieurs applications différentes ayant la même fonction (protocoles http et https pour les navigateurs web ou smtp pour les messageries de courrier électronique). A contrario, ces protocoles peuvent facilement être élaborés par des développeurs souhaitant créer une application ayant besoin de communiquer par internet avec une autre de leurs applications : il leur suffit pour cela de s'appuyer sur TCP ou UDP.

## IV. Quelques gros plans sur des points de cours ...

Après la vue d'ensemble des parties précédentes, nous allons ici effectuer quelques zooms sur des points de cours

### A. Le réseau du lycée

En réalité le réseau enseignant/élèves est découpé virtuellement (grâce à des logiciels) en plusieurs sous-réseaux (VLAN élève, VLAN profs, VLAN impression, VLAN caméras de surveillance ...)



Pour des raisons de sécurité, le réseau du lycée est constitué de deux réseaux différents : le réseau pédagogique (enseignants, élèves, photocopieuses, caméras de surveillance etc.) et le réseau administratif. Dans chacun des six bâtiments élèves, toutes les machines sont connectées au commutateur du bâtiment par des câbles RJ45 (ici seuls trois bâtiments sont symbolisés). Ces six commutateurs sont reliés (par une liaison en fibre optique) à un commutateur principal situé dans un local du CDI. Ce local contient également les deux ou trois machines «serveurs» hébergeant tous les logiciels «serveurs» nécessaires au bon fonctionnement du réseau du lycée :

- Serveur d'authentification des clients (lorsque vous ouvrez votre session)
- Serveur de stockage (qui stocke tous les fichiers hébergés dans «Mes Espaces»)

- Serveur DNS (qui fait le lien entre les adresses IP des machines et leurs noms du type C001-007)
- Serveur DHCP (qui attribue les adresses IP aux machines lors de leur connexion au réseau)
- Serveur proxy qui filtre les accès à internet
- Etc.

Enfin, toujours dans ce local, on trouve le routeur qui est la passerelle vers internet : ce routeur est relié par fibre optique d'un côté au réseau du fournisseur d'accès à internet du lycée (le réseau Renater) et de l'autre côté aux deux commutateurs principaux des deux réseaux (pédagogique et administratif) du lycée.

*Remarque 1* : avoir ce plan du réseau du lycée en tête est important car il peut permettre de diagnostiquer la cause d'un panne en fonction des défaillances observées

*Remarque 2* : sur le schéma ci-dessus il y a bien six ou quatre câbles de dessinés qui sortent des commutateurs de la partie gauche (ce n'est pas un seul câble qui se «divise»)

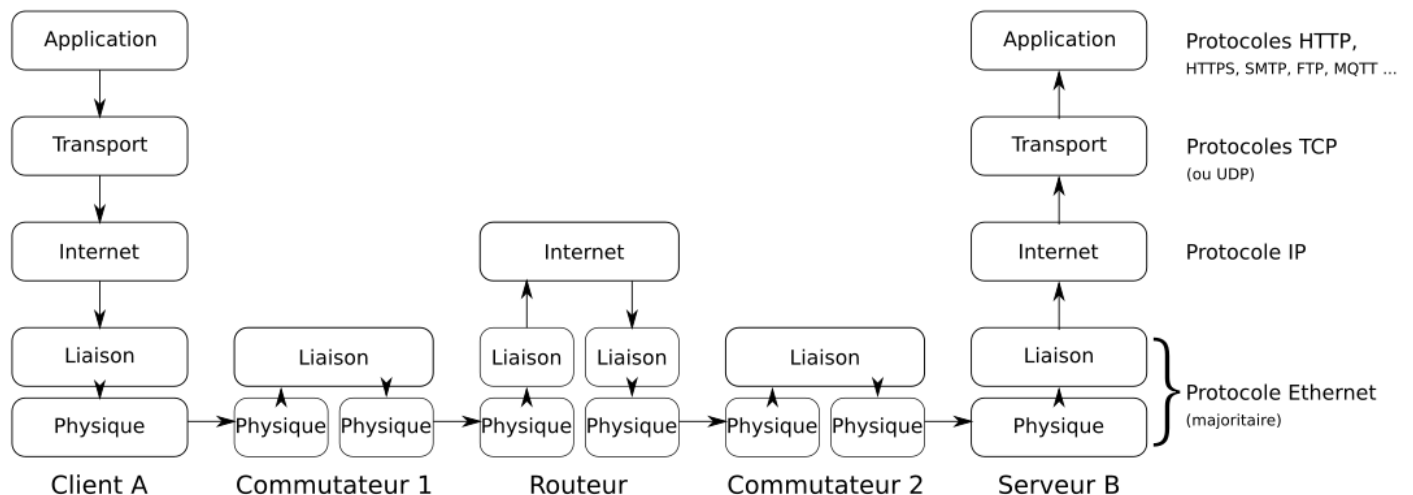
### B. Services DNS et DHCP

Comme indiqué plus haut et comme cela a été vu en SNT pour DNS, ces deux services (qui ne sont pas impérativement nécessaires au fonctionnement de TCP/IP) sont à connaître car très utilisés :

- DNS (Domain Name System) est le services qui permet, en contactant une hiérarchie de serveurs, d'associer un nom à une adresse IP.
  - Par exemple d'associer les noms des machines (C001-01, C001-02, C001-03 etc pour les machines de la salle C001) à leurs IP : ainsi pour gérer le réseau, l'administrateur voir les noms des machines et sait où elles sont : c'est beaucoup plus simple que de manipuler des adresses IP.
  - Par exemple d'associer des noms de domaine internet (www.legtfrancoisvillon.fr) à l'adresse IP du serveur qui gère le nom de domaine correspondant (dans ce cas on parle parfois d'*annuaire* d'internet).
- DHCP (Dynamic Host Configuration Protocol) est un protocole qui permet d'assurer la configuration automatique des paramètres IP d'une machine (adresse IP et masque de réseau pour savoir où découper l'adresse IP en deux).

### C. Communication entre les couches (difficile)

Voici un schéma résumant le circuit d'un ***tout petit paquet*** entre un client et un serveur et transitant à travers deux commutateurs et un routeur :



***Le fichier étant petit, il est découpé en un seul paquet (!)*** par la couche transport qui gère également la question des numéros de port.

La couche transport transmet alors le paquet à la couche internet qui enfin le transmet à la couche liaison (= liaison + physique). Cette couche envoie alors la paquet via un câble ou via le Wi-Fi ou via une fibre optique vers le commutateur.

Le commutateur étant un équipement de la couche liaison, il va traiter ce paquet uniquement grâce à la couche liaison qui gère les adresses MAC. Puis il va l'envoyer vers le routeur, toujours par câble ou wi-fi ou fibre optique. Le routeur, qui est un équipement de la couche internet manipulant les adresses IP, va traiter ce paquet avec sa couche internet. Dans le routeur, le paquet va donc remonter «plus haut» que dans le commutateur. Dit autrement, dans le routeur le paquet va être pris en charge par deux protocoles (Ethernet & IP) contre un seul (Ethernet) pour le commutateur. Le paquet va ensuite ressortir du routeur, passer par le commutateur et arriver sur la machine de destination.

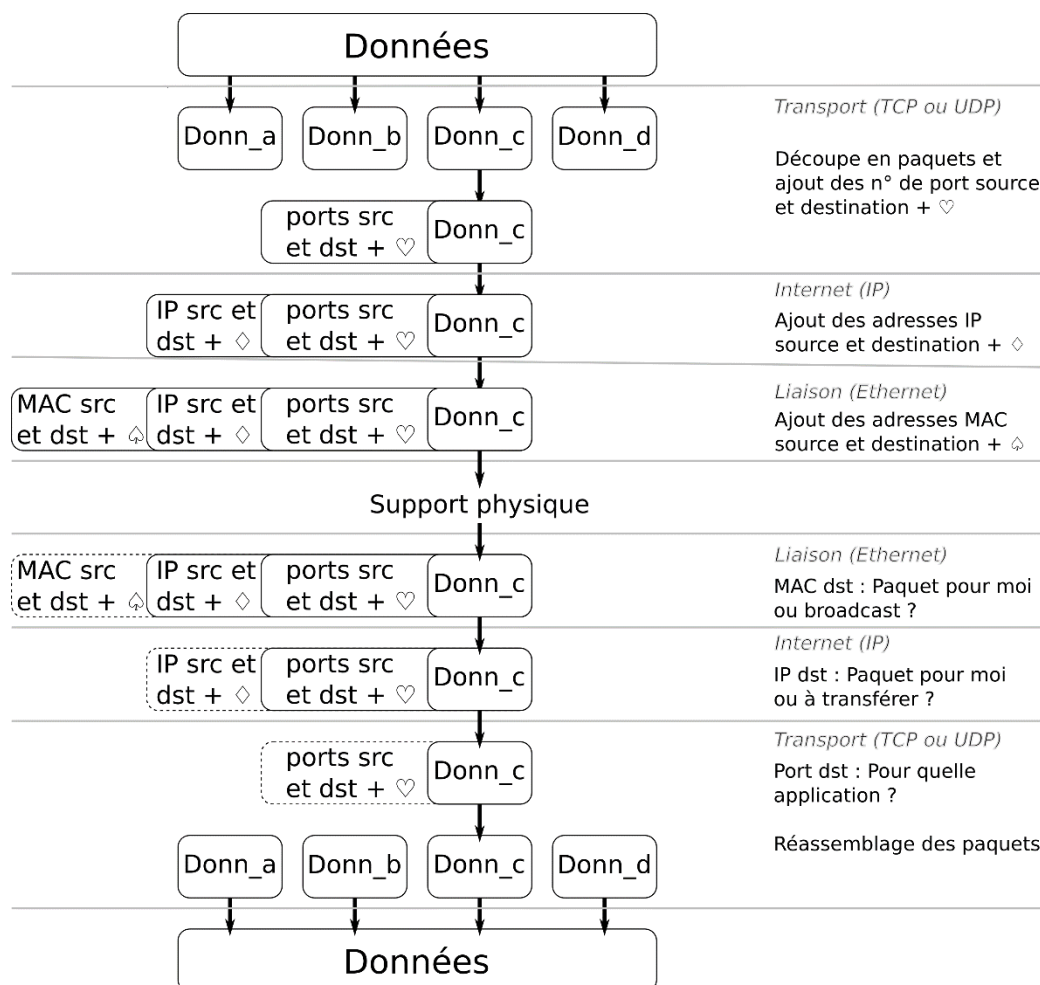
Arrivé sur la bonne machine, il va remonter jusqu'à la couche transport qui va lire le numéro de port de destination et délivrer le paquet à l'application correspondante.

*Remarque* : dans le cas de l'envoi d'un gros fichier, tout ce qui précède reste valable avec deux nuances :

- Le fichier sera découpé en de nombreux petits paquets,
- Chacun des paquets peut prendre un chemin différent sur internet : certains paquets traverseront peut-être un seul routeur, d'autres paquets en traverseront deux ou trois ou cinq ...

### D. Encapsulation (difficile)

Lorsqu'un paquet traverse passe de couche en couche, chaque protocole rajoute (au départ) ou enlève (à l'arrivée) des informations rajoutées dans en tête des données du paquet : on parle d'encapsulation (comme si on avait un colis postal que l'on mettait dans un autre colis avec une étiquette différente). Voici un schéma résumant cela :



L'encapsulation permet de rendre les couches indépendantes (on peut ainsi changer le protocole Ethernet par un autre protocole liaison ou le protocole TCP par UDP). Par ailleurs cela permet de bien gérer la



remontée des différentes couches lorsque le paquet arrive sur une machine (commutateur, routeur, client ou serveur).

*Remarque 1 :* Sur la couche liaison, pour faire le lien entre les adresses MAC et les adresses IP il est parfois nécessaire d'envoyer à toutes les machines du réseau un paquet destiné à une machine X bien précise (on parle d'envoi sur l'adresse IP de broadcast). C'est pourquoi une machine peut recevoir des paquets dont l'adresse MAC de destination ne lui correspond pas : dans ce cas le paquet est éliminé.

*Remarque 2 :* Sur la couche internet, lorsqu'un routeur reçoit un paquet à transférer, l'adresse MAC de destination est bien celle du routeur mais pas l'adresse IP de destination. La couche liaison du routeur va donc accepter le paquet (car MAC dst OK) et le donner à la couche internet du routeur qui va constater que IP dst ne lui correspond pas. La couche internet sait alors qu'il faut transférer le paquet : elle va donc le redonner à la couche liaison qui va le renvoyer sur le réseau.

*Remarque 3 :* Sur ce schéma, un certain nombre de questions n'ont pas de réponse : il serait trop long et hors-programme de s'attarder en détail sur les modalités de passage entre les différentes couches (par exemple expliciter le protocole ARP, les tables CAM, le port forwarding etc.). Par ailleurs un certain nombre d'informations sont ajoutées en plus des adresses source et destination (d'où les cœurs, carreau et pique).

### **E. Adresses MAC vs Adresses IP**

Les adresses MAC (couche liaison, protocole Ethernet) sont codées sur six octets et sont traditionnellement notées en hexadécimal avec deux points en séparateur. Voici un exemple d'adresse MAC valide : 45:C5:8F:98:9A:CB. Et voici deux exemples d'adresses MAC non valides : DE:56:9A:CC:F0 ainsi que AB:7F:30:5F:G2:8D.

Les adresses IPv4 (couche Internet, protocole IP) sont codées sur quatre octets et sont traditionnellement notées en décimal avec un point en séparateur. Voici un exemple d'adresse IPv4 valide : 178.54.241.123. Et voici deux exemples d'adresses IPv4 non valides : 245.65.98.120.32 et 212.111.276.24. Il est à noter qu'il y a trois plages d'adresses IPv4 réservées pour les réseaux privés (non accessibles sur internet) : 10.0.0.0/8, 192.168.0.0/16 et 172.16.0.0/16. Ici le 8 ou le 16 indiquent que les 1 ou 2 octets (8 ou 16 bits) les plus à gauche désignent le numéro de réseau alors que les autres octets désignent le numéro de machine sur le réseau. Les adresses IPv4 étant épuisées on est désormais passé aux adresses IPv6 codées sur seize octets.

#### **Adresses IP réservées ne pouvant être données à des machines :**

Le numéro de machine 0 est réservé pour l'adresse de réseau : 172.55.241.0/24 par exemple.

Et le plus grand numéro de machine possible est réservé pour donner l'adresse de broadcast c'est-à-dire l'adresse utilisée pour envoyer un message à toutes les machines du réseau à la fois. Ainsi si vous envoyez un paquet à 156.233.255.255/16 toutes les machines du réseau 156.233.0.0/16 recevront ce message.

### **F. Architecture client-serveur**

On parle d'architecture client-serveur lorsque deux programmes informatiques s'échangent des informations à travers un réseau. Le programme client est celui qui envoie des requêtes et le programme serveur est le programme qui répond aux requêtes. Pour que cette architecture soit fonctionnelle, le client et le serveur doivent utiliser le même protocole au niveau de la couche application et pour la couche transport (par exemple il n'est pas possible d'envoyer des paquets TCP et de chercher à les réceptionner avec UDP).

### **G. Protocole du bit alterné (capacité attendue au programme)**

Voir le TP : ce protocole du bit alterné est à savoir dérouler

### **H. Message, segment, paquet ou datagramme, trame, signal**

Dans tout ce cours nous avons utilisé le terme «paquet» : sachez qu'on peut être plus précis et parler de «message» au niveau de la couche application, de «segment» au niveau de la couche transport, de «paquet» ou «datagramme» au niveau de la couche internet, de «trame» au niveau de la couche liaison et enfin de «signal» au niveau de la partie physique de la couche liaison.