

Réseau Informatique

Guthemberg Silvestre

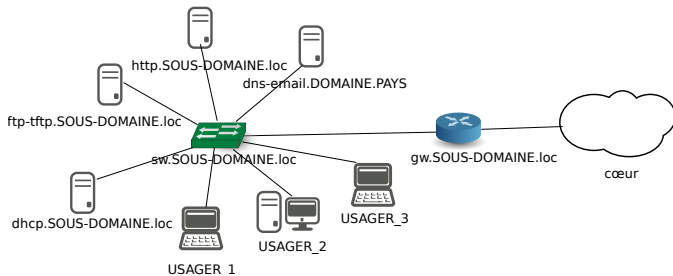
Lundi, le 20 juin 2016



ECOLE NATIONALE DE L'AVIATION CIVILE

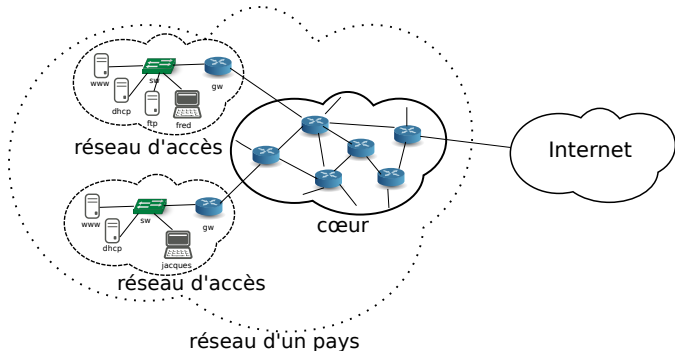
Les ordinateurs des usagers sont branchés aux réseaux d'accès

- Nous allons configurer des serveurs, des ordinateurs, des commutateurs, des routeurs d'accès, *etc.*



Le cœur du réseau transmet les informations aux réseaux d'accès distants

- Le routage de données joue un rôle très important dans le cœur du réseau



Les objectifs des cours et des travaux pratiques sur Réseau Informatique pour le Cycle Ingénieur SIAE

- Comprendre les services offerts par le réseau informatique, en particulier, les services de la pile de protocoles de l'Internet
- Être capable de décrire les liens logiques d'un réseau informatique
- Être capable de planifier un réseau local ou d'accès
- Savoir déployer et configurer un nœud dans un réseau local
- Comprendre le routage statique et dynamique de données
- Savoir déployer et configurer des routes statiques sur routeur dans un réseau d'accès
- Être capable de diagnostiquer et résoudre de problèmes relatifs à interconnexion d'un réseau local

Ce module est organisé en trois parties

- Les cours
 - Le réseau informatique et l'Internet
 - Les applications
 - La couche transport
 - La couche réseau
 - La couche liaison
- Les travaux pratiques
 - Introduction à l'outil Cisco Packet Tracer, les applications et la mise en place d'un réseau local fonctionnel ;
 - Les protocoles de transport : le multiplexage et la fiabilité de communications bout-à-bout (TCP/UDP) ;
 - L'interconnexions des réseaux avec l'Internet Protocol (IP) ;
 - La couche liaison, les réseaux virtuels locaux (VLAN) et la traduction d'adresse (NAT) ;
- Les examens : deux examens écrit et un pratique

Nous utiliserons la méthode *top-down* pour l'apprentissage du réseau informatique

- Le réseau informatique et l'Internet
- Les applications
- La couche transport
- La couche réseau
- La couche liaison

Nous allons approfondir nos connaissances avec des travaux pratiques

- Notre objectif est de mettre en place un réseau entre la Chine et la France :



- Vous allez travailler en équipes de deux : un s'occupera du réseau de la Chine et l'autre du réseau de la France
- Le réseau d'un pays est composé fondamentalement de plusieurs réseaux d'accès locaux interconnectés par un ensemble de routeurs qui forment le cœur du réseau d'un pays
- L'Internet sera représenté par une interconnexion entre les cœurs du réseau de deux pays
- Les travaux seront menés avec l'aide d'outil d'émulation Cisco Packet Tracer

Le réseau informatique et l'Internet

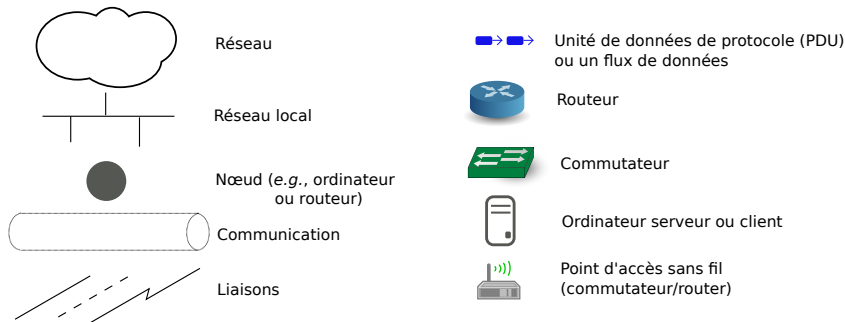
Qu'est-ce que nous allons apprendre?

- Comment une application utilise un réseau informatique.
- La structure modulaire de l'Internet : le modèle de pile de protocoles en 4 couches.
- Les principes de l'architecture de l'Internet.
 - La commutation de paquets.
 - Les protocoles/couches réseau offrent des services spécifiques et complémentaires.
 - L'encapsulation.

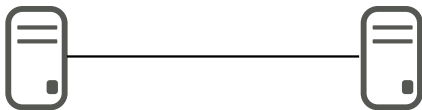
Le réseau informatique transmet des informations génériques

- L'usage d'un réseau de communication normalement est spécifique : réseau téléphonique ou le réseau de TV à câble.
- Le réseau informatique transporte des données génériques, *e.g.*, le la voix, du vidéo/télé, . . . , à la maison, au bureau, *etc.*

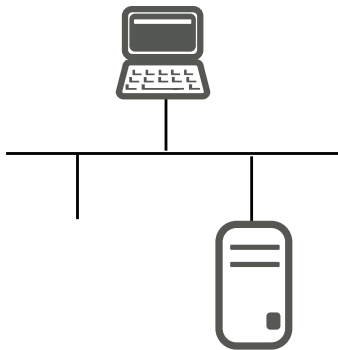
Nous utiliserons des images type pour représenter les éléments d'un réseau



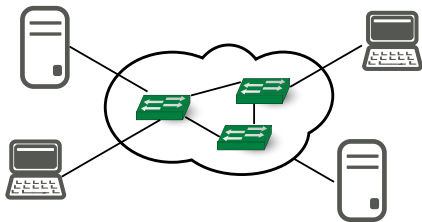
Une représentation logique d'un réseau entre deux nœuds



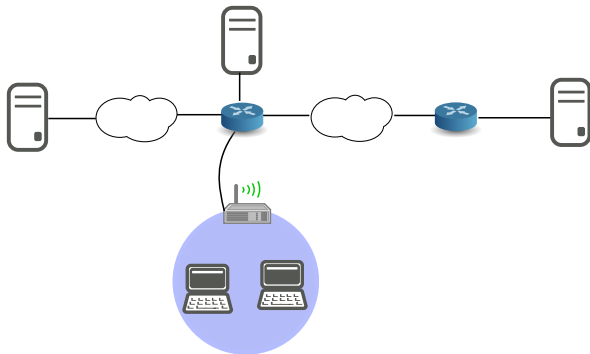
Une autre représentation logique d'un réseau local



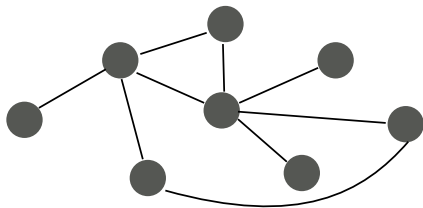
Le réseau local peut être constitué de plusieurs commutateurs



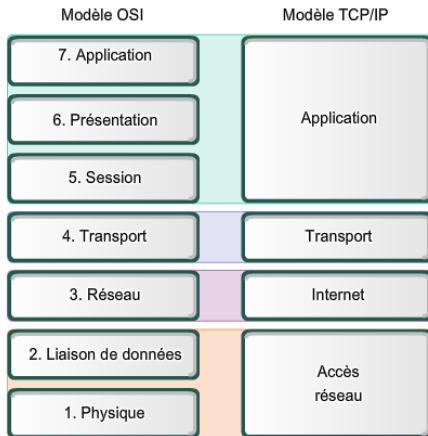
Nous pouvons représenter plusieurs réseaux interconnectés



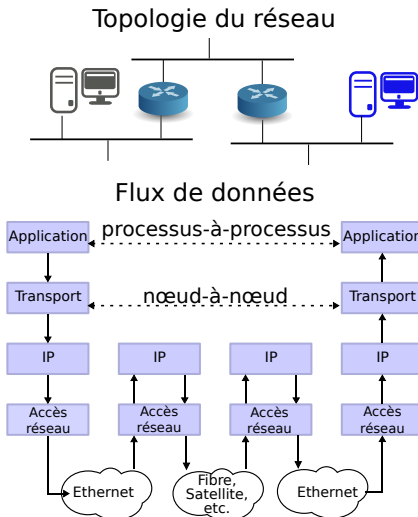
Nous pouvons représenter les nœuds et les liens d'un réseau en forme de graphe



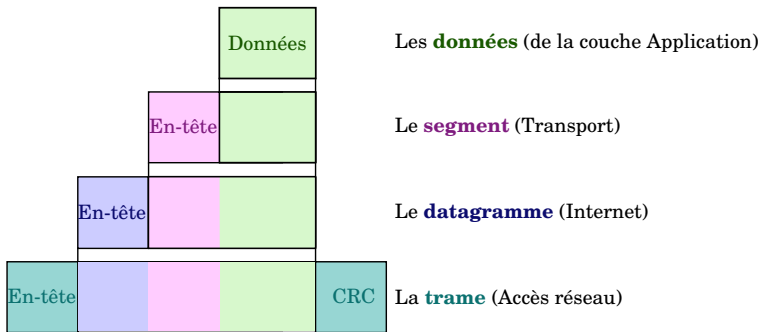
Les services de la pile de protocoles de l'Internet, le TCP/IP, sont organisés d'une façon modulaire



La communication entre les couches TCP/IP de nœuds voisins dépend du modèle du service offert

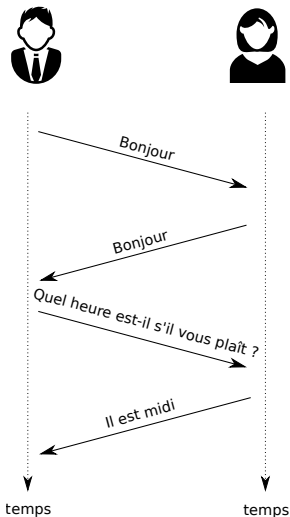


Chaque couche **offre un service** et **encapsule** les données de la couche qui est juste au-dessus d'elle

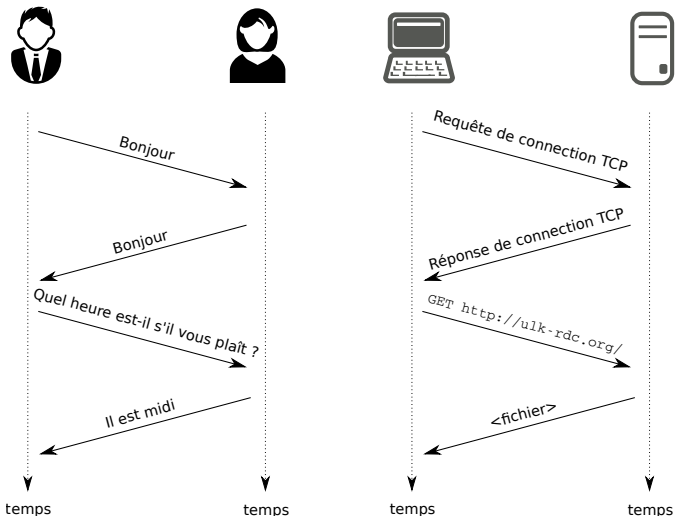


Les services sont définis par les protocoles de chaque couche

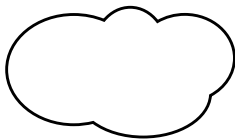
Un protocole réseau nous permet de formaliser une communication afin d'offrir un service



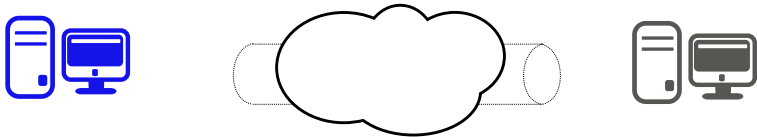
Un protocole réseau nous permet de formaliser une communication afin d'offrir un service



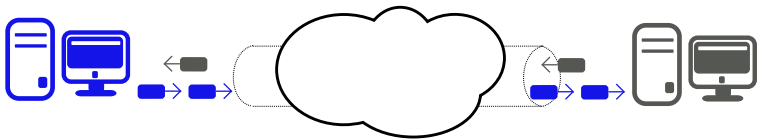
La plupart des applications utilisent des protocoles afin d'échanger des données d'une façon fiable



La plupart des applications utilisent des protocoles afin d'échanger des données d'une façon fiable



La plupart des applications utilisent des protocoles afin d'échanger des données d'une façon fiable

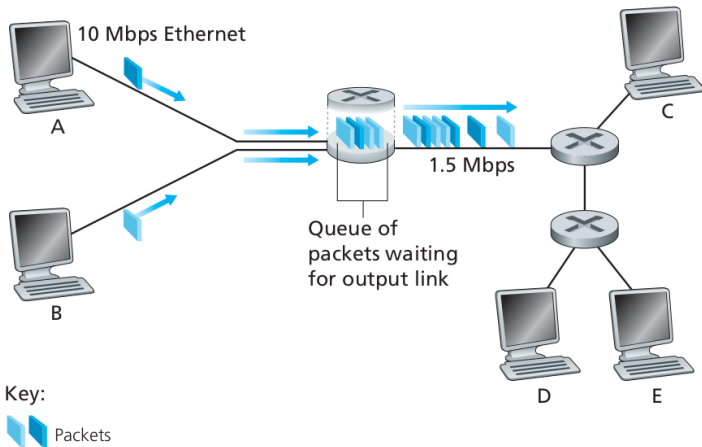


- Le réseau sert à écrire et lire des données d'une application
- Le modèle des communication prédominant permet un échange de données :
 - qui est fiable;
 - qui marche dans les deux sens;

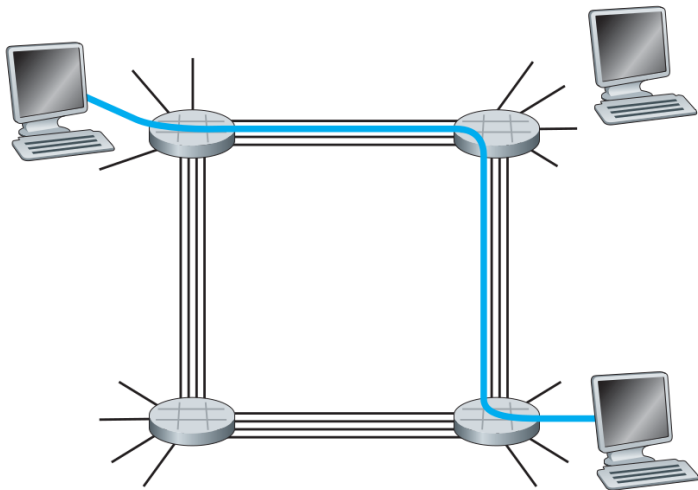
Les protocoles nous permettent de partager les ressources du réseau informatique entre les nœuds

- Les protocoles réseau utilisent la commutation de données afin de partager les ressources informatiques
- Il y a deux types principaux de commutation :
 - Commutation de paquets
 - Commutation de circuits

Dans la commutation de paquets, les en-têtes identifient les différentes communications



La commutation de circuit nous permet de réserver de ressources entre nœuds du réseau



Applications

Les applications sont fondamentales pour le développement des réseaux informatiques

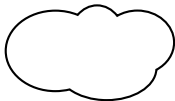
- Au tout début (1970-80, *en mode texte*) : e-mail, transfert de fichiers, accès à distance, *newsgroups*, etc.
- Une application exceptionnelle apparaît à la moitié des années 90 : World Wide Web
- Plusieurs applications sont devenues populaires depuis l'apparition de l'Internet : partage de fichiers par réseau pair-à-pair, la transmission de voix et vidéo sur IP (e.g., Skype, WhatsApp, NetFlix), les jeux vidéo (e.g., Second Life, World of Warcraft, League of Legends),...
- Et plus récemment : les réseaux sociaux (e.g., Facebook, Twiter,...)

Nous allons étudier des applications essentielles

- HTTP : *Hypertext Transfer Protocol*
- DNS : *Domain Name System*
- DHCP : *Dynamic Host Configuration Protocol*
- FTP : *File Transfer Protocol*
- SMTP : *Simple Mail Transfer Protocol*
- POP : *Post Office Protocol*
- TFTP : *Trivial File Transfer Protocol*
- Applications pair-à-pair

Nous utilisons le *Hypertext Transfer Protocol* (HTTP) pour télécharger les données d'une application web

Serveur
HTTP

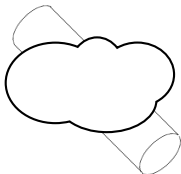


Client

- HTTP est le protocole le plus utilisé pour les transferts des données sur l'Internet
- Ce protocole utilise un modèle de service du type client/serveur
- Protocole en format texte (ASCII), lisible pour nous
- Le protocole HTTP fonctionne par un échange de requêtes et réponses (*i.e.*, l'*application programming interface* (API) HTTP)

Le protocole HTTP échange de données d'une façon fiable

Serveur
HTTP

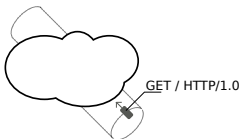


Client

- Étape préliminaire :
Établissement d'une connexion dont l'échange de données se fait d'une façon fiable
- La fiabilité de la connexion est assurée par la un service de la couche transport

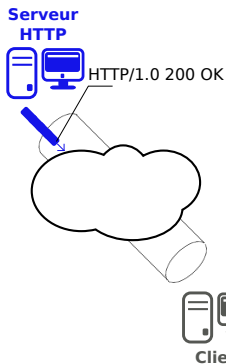
Une requête GET nous permet de solliciter un contenu

Serveur
HTTP



- Première étape : l'envoi d'une requête **GET** pour solliciter un contenu

La réponse au GET est encapsulée dans un message HTTP



- Le serveur traite la sollicitation et envoi une réponse au client
- Si le contenu n'est pas disponible, la réponse contient un message d'erreur (e.g., l'erreur 404, *page not found*)

Lorsque le client traite la réponse la connexion est terminée

Serveur
HTTP

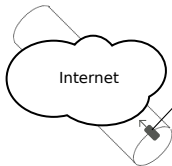


Client

- Les nœuds finalisent la connexion

L'envoi d'une requête GET au serveur web de l'*Civil Aviation University of China*

Civil Aviation
University of China



GET / HTTP/1.0

User-Agent: Wget/1.12 (linux-gnu)

Accept: */*

Host: www.cauc.edu.cn

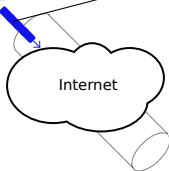
Connection: Keep-Alive



`wget http://www.cauc.edu.cn/zh/`

La réponse à notre GET contient la page principale (la réponse par défaut)

Civil Aviation
University of China



wget http://www.cauc.edu.cn/zh/

HTTP/1.0 200 OK

Content-Length: 814
Date: Tue, 10 Nov 2015 11:11:46 GMT
Content-Location: http://www.cauc.edu.cn/zh/default.html
Content-Type: text/html
ETag: "51fe9a315e53d11:2feb"
Server: Microsoft-IIS/6.0
Last-Modified: Wed, 20 Jan 2016 08:40:19 GMT
Accept-Ranges: bytes
X-Powered-By: ASP.NET

...

<!DOCTYPE html>

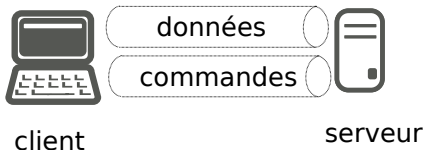
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml2/DTD...
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=gb2312" />
...

Le *Dynamic Host Configuration Protocol* (DHCP) permet aux nœuds d'un réseau local de s'auto-configurer

- Les informations essentielles : l'adresse IP, le masque de sous-réseau, la passerelle, et le serveur de noms
- Fonctionnement (*Request for Comments*, RFC, 2131) :
 1. Le **client** envoie une requête *Discover* vers tous les nœuds du réseau local
 2. Un **serveur** lui répond avec un message du type *Offer* avec une proposition d'adresse IP
 3. Le **client** répond au serveur en lui envoyant une requête (le message *Request*) de l'adresse proposée
 4. Un **serveur** acquitte la demande la requête du client. Le client a l'adresse pendant une durée déterminée (le temps de *lease*) au bout de laquelle le client doit redemander l'adresse

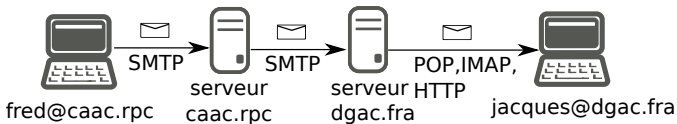
Le *File Transfert Protocol* nous permet de stocker ou partager de fichiers sur le réseau

- Ainsi comme les applications prétendantes, FTP utilise aussi le modèle client/serveur
- Le client démarre le transfert de fichiers vers ou à partir d'une serveur distant
- Il y a deux canaux de communication entre le client et le serveur : un pour les commandes et un autre pour les transferts de fichiers



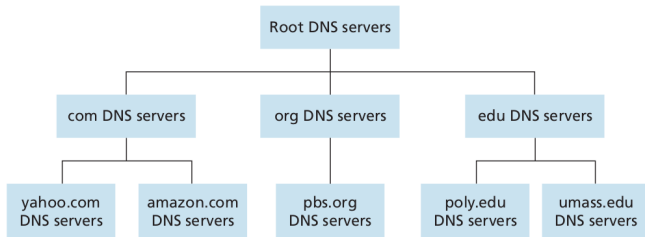
Le service de messages électroniques, l'e-mail

- Il y a trois composants principaux :
 - l'agent client : les applications qui nous permettent de rédiger, envoyer, recevoir et lire les e-mails ;
 - le serveurs d'e-mails : jouent un rôle de relai entre l'émetteur et le destinataire d'un message ;
 - *Simple Mail Transfert Protocol* (SMTP) crée des liens logiques entre les serveurs d'e-mails ainsi qu'entre client et serveur;
- Les agents implémentent des protocoles spécifiques pour récupérer les e-mails : *Post Office Protocol* (POP), *Internet Mail Access Protocol* (IMAP), HTTP



Le *Domain Name System* (DNS) traduit les noms de nœuds en identifiant réseau (IP)

- Le serveur de noms, le DNS, est essentiel pour le bon fonctionnement de l'Internet
- C'est beaucoup plus simple d'écrire et de mémoriser des noms que des identifiants décimaux ou binaires (e.g., le nom `www.google.fr` plutôt que l'identifiant (IP) `77.154.221.247`)



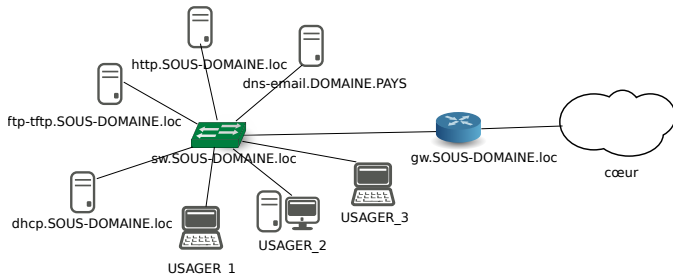
Le DNS utilise des registres dans un format spécifique pour diffuser les informations de noms

- Base de données distribuée pour stocker des ressources de registres (RR)
- Format de RR : (nom, valeur, type, temps de vie du registre)

type de registre	valeur	nom (description)
A	identifiant IP	le nom d'un nœud
NS	le serveur qui répond pour ce domaine	le nom du domaine
CNAME	le vrai nom	un alias du nœud
MX	le serveur SMTP	le domaine de l'email

Travail pratique sur Cisco Packet Tracer

- Une introduction à l'usage de l'outil de simulation Cisco Packet Tracer ¹
- Mettre en œuvre des applications sur un réseau d'accès afin de le rendre opérationnel

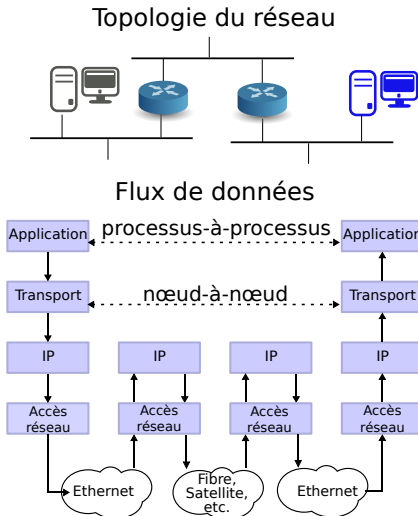


¹Cisco Packet Tracer, *an innovative network configuration simulation tool* :
<https://www.netacad.com/about-networking-academy/packet-tracer/>

Transport

La communication entre couches de nœuds voisins change selon le modèle de service offert (**TCP/IP**)

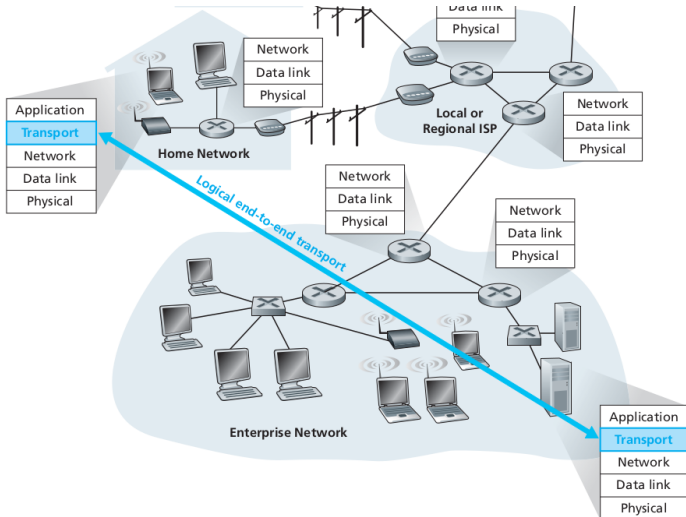
(Rappel)



La couche transport contrôle le transfert de flux de données d'une application entre pairs de nœuds

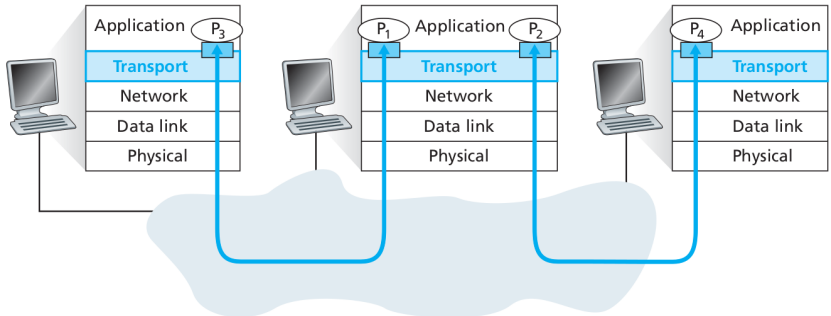
- Objectif : comprendre les services que la couche transport offre à la couche application :
 - Multiplexage/démultiplexage de communications entre nœuds
 - Le transfert fiable de données
 - Contrôle de flux
 - Contrôle de congestion
- Nous allons étudier le fonctionnement des deux principaux protocoles de transport de l'Internet
 - User Datagram Protocol (UDP) : transport de données **sans connexion**
 - Transmission Control Protocol (TCP) : transport fiable de données orienté **connexion**
 - Contrôle de congestion TCP

La couche transport offre un lien logique entre deux nœuds qui se communiquent

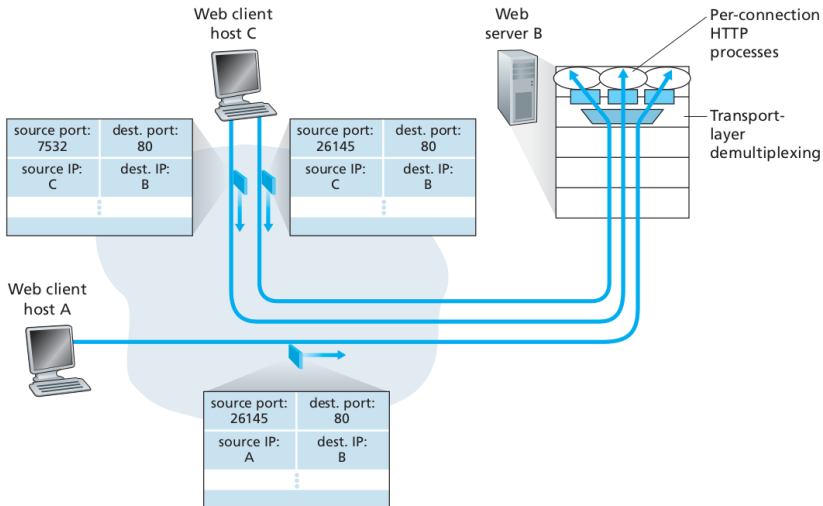


Le transport nous permet d'établir des communications simultanées vers/à partir les processus des applications

- Multiplexage par l'émetteur : manipule des données de différents processus (application) et ajoute des en-têtes
- Démultiplexage du côté du récepteur : utilise les informations des en-têtes afin de livrer les données vers les processus (destinataires) correspondants



Le transport utilise des ports afin de multiplexer/démultiplexer les communications



Le protocole de transport fonctionne de chaque côté de la communication

- L'émetteur découpe les messages de l'application en segments
- Le récepteur rassemble les segments pour passer à l'application

Application

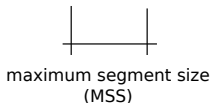
message

Transport

segment

segment

segment



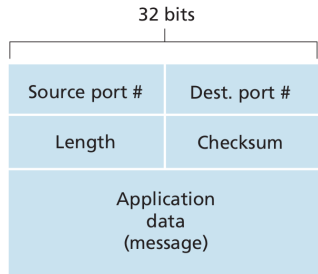
- Sur Internet, il y a notamment deux protocoles disponibles pour les applications : TCP et UDP

Les principales différences entre les protocoles TCP et UDP

- L'UDP offre aux applications un service simple et flexible, mais sans garanties (c'est du *best effort*, comme l'IP)
- Les segments sur TCP sont acheminés d'une façon fiable et en ordre
 - Gestion de connexions (son type de communication/tuyau)
 - Contrôle de congestion et de flux pour les applications : TCP et UDP
- Aucun des deux offre des garanties sur le débit ou le temps de réponse

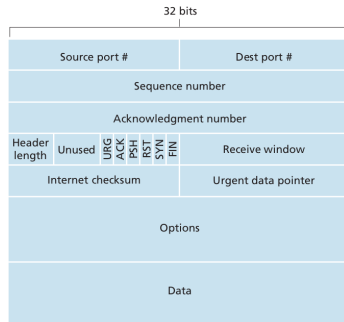
L'UDP offre un service de transport de données très simple et flexible pour les application

- Il n'y a pas de connexion et chaque segment est traité d'une façon indépendante
- Le champs *checksum* (l'en-tête y est compris) sert à détecter des erreurs
- La taille du segment comprend l'en-tête
- UDP est utilisé par :
 - les applications de streaming (tolère de pertes, haut débit)
 - DNS, RIP (routage, couche réseau)

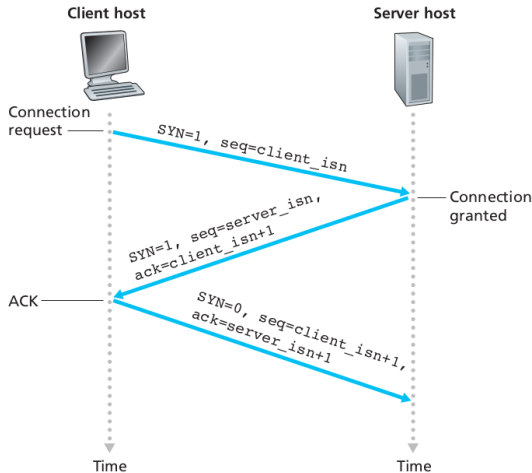


Le protocole TCP offre un service de transport de données fiable et orienté connexion

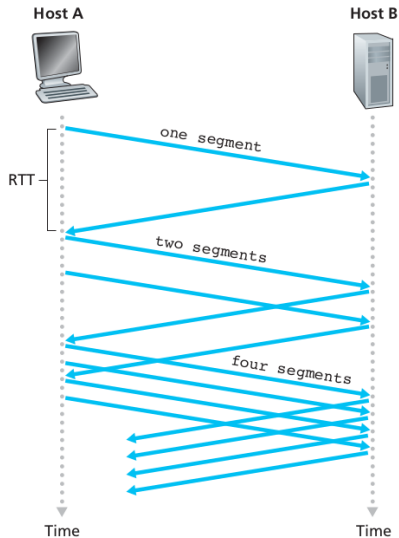
- Communication point à point bidirectionnelle
- TCP transmet d'une façon fiable des séquences d'octets en ordre
- Gestion d'un “tuyau” de communication (lien logique) où il y a le contrôle de flux (débit) et de congestion (perte)
- orienté connexion : (1) établissement de la connexion, (2) envoie de données, (3) fin de la connexion



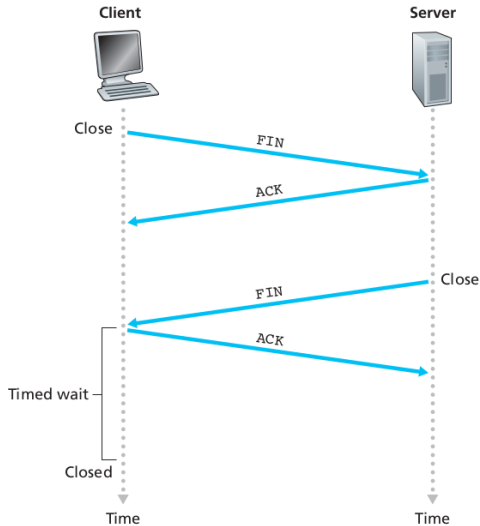
D'abord, le protocole TCP échange de segments entre les nœuds afin d'établir une connexion



Ensuite, le TCP envoie des données d'une façon progressive



Enfin, le TCP ferme la connexion



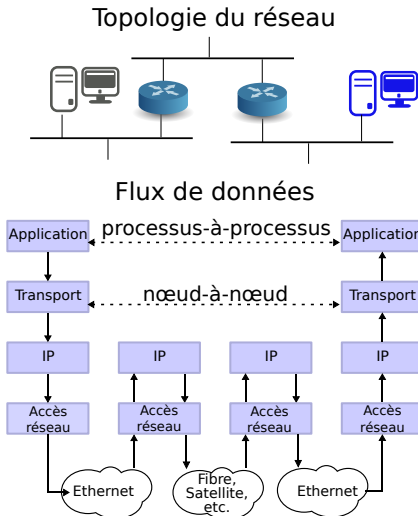
Liste d'applications et leur protocoles de transport

Application	Protocole	Transport
E-mail	SMTP	TCP
Accès au terminal distant	Telnet	TCP
Web	HTTP	TCP
Transfert de fichiers	FTP	TCP
Système de fichiers sur le réseau	NFS	Généralement UDP
Streaming (vidéo)	Généralement propriétaire	UDP ou TCP
Téléphonie sur Internet	Généralement propriétaire	UDP ou TCP
Protocole de routage	RIP	Généralement UDP
Service de traduction de noms	DNS	Généralement UDP

Réseau

La communication entre couches de nœuds voisins change selon le modèle de service offert (**TCP/IP**)

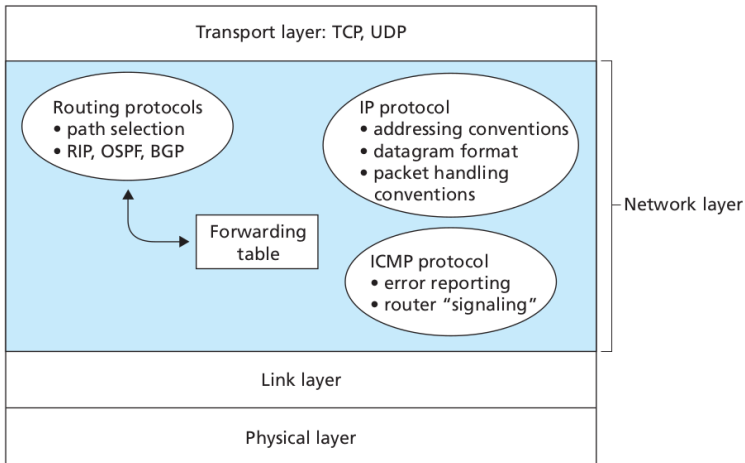
(Rappel)



La couche réseau est fondamentale pour la interconnexion des nœuds en large échelle

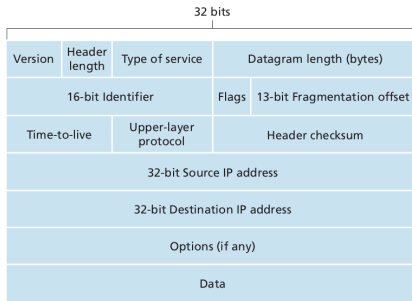
- Cette couche nous offre le transport de segments qui sont encapsulés en datagrammes
- La couche réseau doit être présent sur tous les nœuds du réseau (e.g., ordinateurs, routeurs, serveurs,...)
- Deux fonctions essentielles :
 - Créer des liens logiques avec les nœuds voisins
 - Déterminer la route d'un paquet de la source à sa destination
 - Acheminement de messages qui arrivent à une interface d'entrée vers la "bonne" interface de sortie
- Sur l'Internet nous utilisons l'Internet Protocol (IP) version 4 (IPv4) et version 6 (IPv6)
- Types de communication : *broadcast*, *multicast* et *unicast*

L'IP est le protocole de la couche réseau qui interconnecte tous nœuds sur Internet



L'en-tête du datagramme d'IPv4 a 20 octets

- *Version* : La version du protocole IP (4 bits)
- *Time to Live* (TTL) : temps de vie d'un datagramme
- *Protocol* : TCP (6), UDP (17), etc.
- Les adresses de source et destination

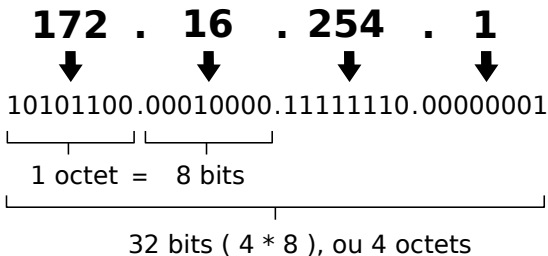


Source : Kurose, J., and K. Ross. Computer Networking: A Top Down Approach, 4e. Vol. 1. 2012.

Les adresses (binaires) IP sont représentées par une notation décimal à point

- Un hôte est interconnecté par (ou moins) une interface réseau; un routeur a souvent deux ou plusieurs interfaces
- Chaque interface réseau a une adresse de 32 bits d'un sous-réseau IP

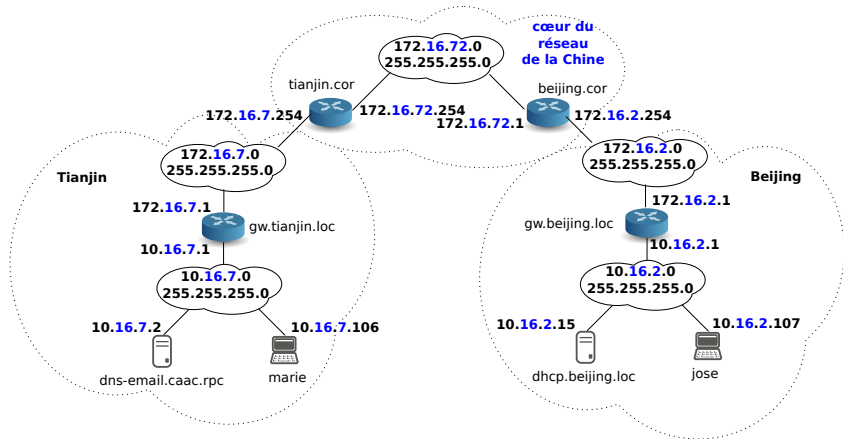
Une adresse IPv4 (notation décimale à point)



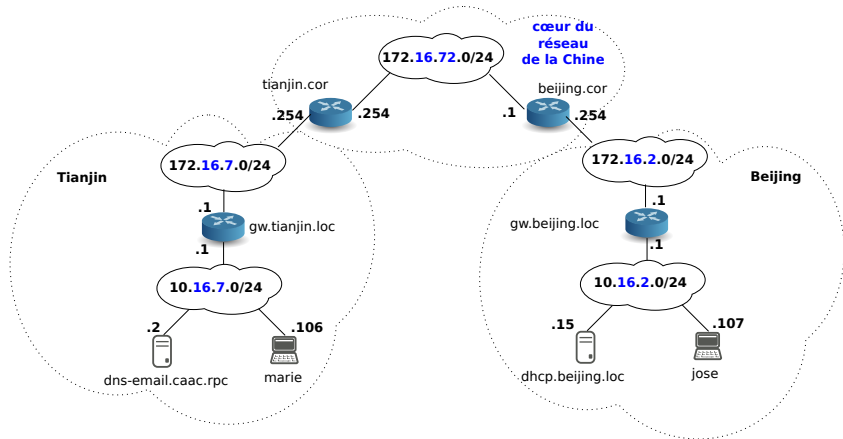
Un sous-réseau représente un ensemble d'adresses IP de noeuds "voisins" (réseau local)

- deux nœuds avec des interfaces du même sous-réseau peuvent se communiquer directement (pas besoin d'un routeur au milieu)
- l'adresse IP est composé de deux parties (exemple, l'adresse 172.16.7.1 et masque de sous-réseau 255.255.255.0, ou bien, 172.16.7.1/24) :
 - le sous-réseau (172.16.7, *high order bits*)
 - la partie hôte (1, *low order bits*)
- le masque de sous-réseau nous permet d'identifier la partie sous-réseau de l'adresse, pour 172.16.7.1/24, les 24 premiers bits représentent le sous-réseau

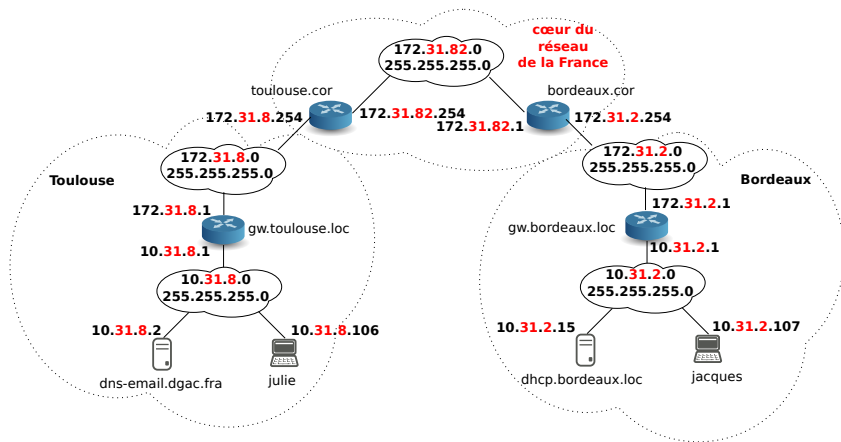
Il y cinq sous-réseaux entre les nœuds de Tianjin à Beijing



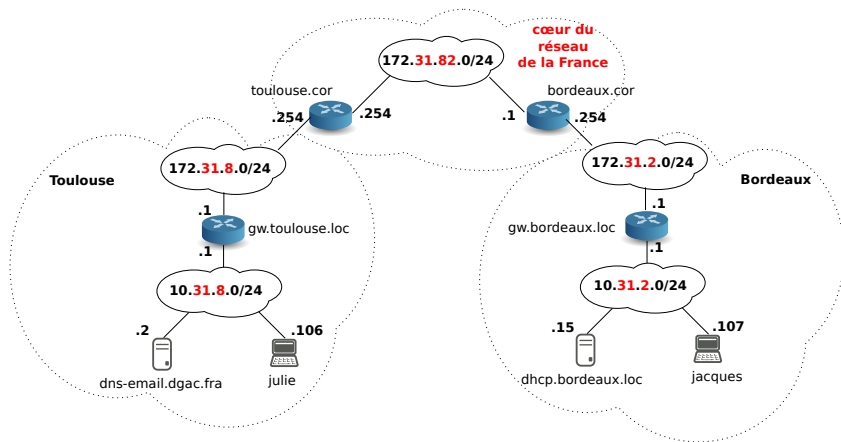
Il y cinq sous-réseaux entre les nœuds de Tianjin à Beijing



Il y cinq sous-réseaux entre les nœuds de Toulouse à Bordeaux



Il y cinq sous-réseaux entre les nœuds de Toulouse à Bordeaux



Le format de l'adresse sur Internet est flexible pour faciliter l'agrégation des préfixes de sous-réseaux

- l'Internet utilise des adresses IP en Classless InterDomain Routing (CIDR) : la taille en bits de la partie sous-réseau est arbitraire
- le format de l'adresse : a.b.c.d/x, où x est la quantité de bits de la partie sous-réseau de l'adresse IP

<u>11001000 00010111 00010000</u>	<u>0 00000000</u>
sous-réseau	hôte

200.23.16.0/23

L'ICMP permet aux nœuds de diffuser des informations sur l'interconnexion sur IP

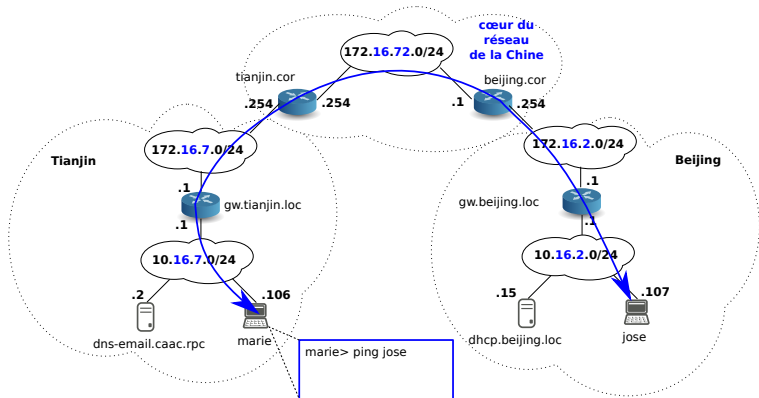
- il sert à diffuser des messages d'erreurs et des informations sur le routage
- les messages servent à notifier des erreurs sur un chemin d'une communication, ou bien, à vérifier si un hôte, un port, un réseau ou protocole est atteignable par le réseau
- les messages d'ICMP sont transmises dans des datagrammes IP
- ICMP sert aussi à vérifier la connectivité entre deux nœuds, cela se fait avec l'échange de messages suivant :
 1. l'émetteur, celui qui vérifie activement la connectivité, envoie d'un message du type *echo request* (message ICMP du type 8, code 0) vers une cible
 2. la cible traite le message et répond à l'émetteur avec un message ICMP *echo reply* (message ICMP du type 0, code 0)

Nous pouvons échanger des messages de vérification de connectivité par ICMP avec la commande ping

- Comment l'utiliser:

```
ping 'nom ou IP du noeud cible'
```

- Exemple : marie vérifie la connectivité au nœud jose



Le nœud jose est atteignable à partir de marie

```
marie>ping jose
```

```
Pinging 10.16.2.107 with 32 bytes of data:
```

```
Reply from 10.16.2.107: bytes=32 time=12ms TTL=124
```

```
Reply from 10.16.2.107: bytes=32 time=11ms TTL=124
```

```
Reply from 10.16.2.107: bytes=32 time=11ms TTL=124
```

```
Reply from 10.16.2.107: bytes=32 time=12ms TTL=124
```

```
Ping statistics for 10.16.2.107:
```

```
  Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
```

```
Approximate round trip times in milli-seconds:
```

```
  Minimum = 11ms, Maximum = 12ms, Average = 11ms
```


Les messages ICMP peuvent être utilisés pour vérifier chaque saut (routeur) du chemin entre deux nœuds

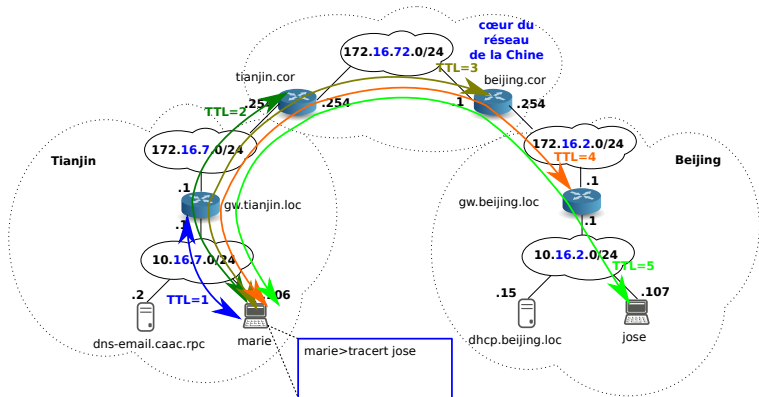
- Nous utilisons le *time to live* (TTL) du datagramme IP
- Le TTL est un nombre entier (un octet, donc 255 maximum) qui représente le temps de vie d'un datagramme IP
- A chaque saut, le TTL est décrémenté de 1
- Quand le TTL d'un datagramme atteint 0, il est supprimé et un message ICMP d'erreur (type 11, code 0) est envoyé à l'émetteur
- La vérification du chemin entre deux nœuds se fait avec des envois successifs à partir de la source de datagrammes avec des TTLs de plus en plus grands

Nous pouvons vérifier le chemin avec la commande traceroute (ou tracert)

- Comment l'utiliser:

```
traceroute 'nom ou IP du noeud cible'
```

- Exemple : marie vérifie le chemin vers jose



La sortie de la commande montre les routeurs entre marie et jose

```
marie>tracert jose
```

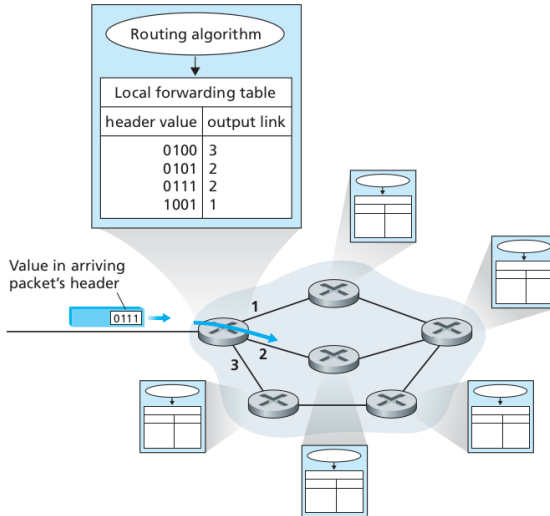
```
Tracing route to 10.16.2.107 over a max of 30 hops:
```

1	0 ms	0 ms	0 ms	10.16.7.1
2	10 ms	10 ms	0 ms	172.16.7.254
3	0 ms	3 ms	10 ms	172.16.72.1
4	10 ms	10 ms	14 ms	172.16.2.1
5	11 ms	11 ms	13 ms	10.16.2.107

```
Trace complete.
```

```
marie>
```

Les tables de routages déterminent le chemin du datagramme



Nous pouvons ajouter manuellement des routes à un routeur

Table de routage

Réseau et masque de sous-réseau	Prochain saut (ou passerelle)
10.16.7.0/24	172.16.7.1

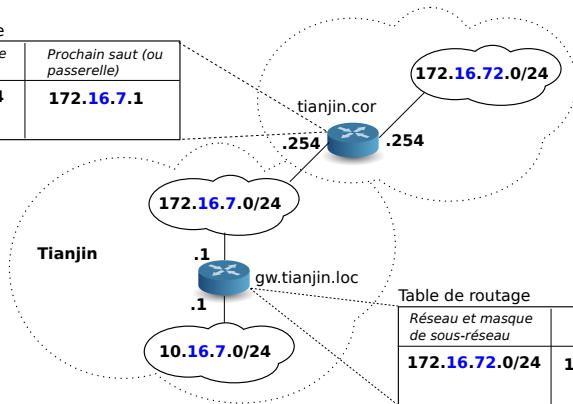


Table de routage

Réseau et masque de sous-réseau	Prochain saut (ou passerelle)
172.16.72.0/24	172.16.7.254

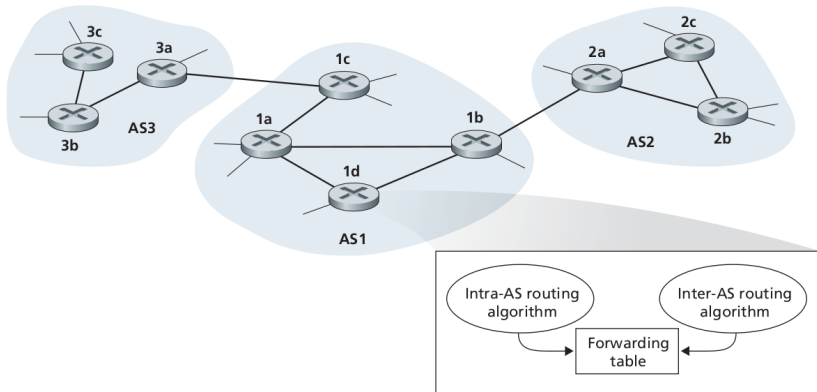
Nous classons les protocoles de routage selon leurs réactions aux modifications de la topologie

- Statique
 - les routes changes lentement
 - les modifications sont faites manuellement
- Dynamique
 - les routes changent rapidement
 - il y a des échanges d'informations périodiques
 - il répondent aux changement de coût des liens
 - Exemples : *Border Gateway Protocol* (BGP), *Open Shortest Path First* (OSPF), *Routing Information Protocol* (**RIP**)
- les protocoles de routage sur Internet ne s'adaptent pas dynamiquement aux changement de charge de liens (congestion)

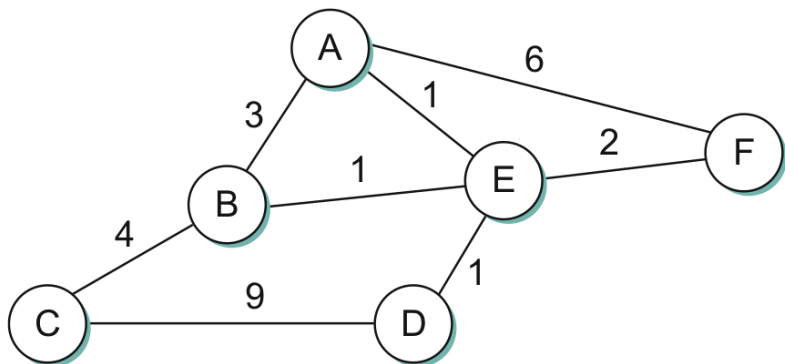
Les routeurs et leur protocoles de routage sont organisés d'une façon hiérarchique

- L'Internet est un système reparté composé de systèmes autonomes qui sont liés logiquement les uns aux autres
- Considérations importantes liées à la taille du système : le passage à l'échelle et l'autonomie de gestion
- Un **système autonome** (AS) sur Internet est un ensemble de routeurs qui sont gérés par une même organisation (e.g., fournisseur d'accès)
- Un système autonome a un seul protocole de routage qui est intérieur
- Il y a deux type de protocole de routage :
 1. **Intra-AS** *routing protocol*
 2. **Inter-AS** *routing protocol*

BGP4 est le *inter-AS routing protocol* qui crée les liens logiques entre tous les systèmes autonomes



Le réseau de routeurs peut être vu comme un graphe



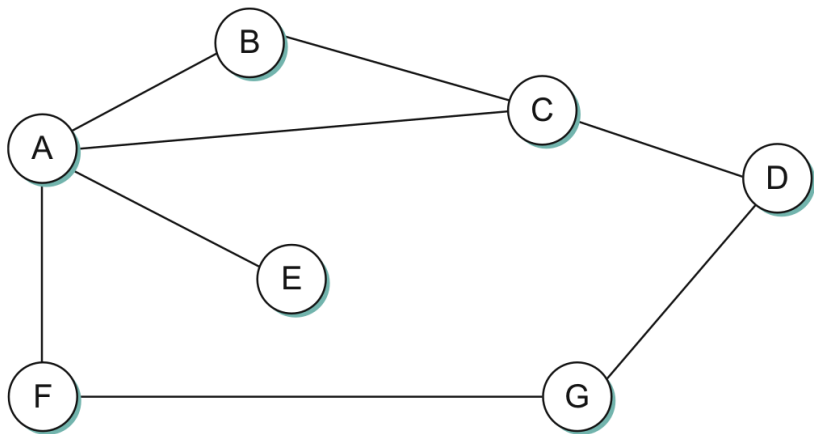
Un protocole de routage peut être classé selon la visibilité de la topologie

- Global
 - il connaît de toute la topologie (tous les routeurs, tous les liens logiques et leurs coûts respectifs)
 - il utilise un algorithme à état de liens (e.g., OSPF)
- Décentralisé
 - au départ, chaque routeur connaît seulement ses voisins les liens locaux et leurs coûts
 - les routeurs échangent des messages entre voisins afin de diffuser les informations sur la topologie (la topologie est créée au fur et à mesure)
 - il utilise un algorithme à vecteur de distance (e.g., **RIP**)

RIP est un protocole de routage du type intra-AS très utilisé sur Internet

- La mesure du coût ou de la distance du chemin est comptée en sauts
- Fonctionnement :
 1. chaque routeur maintient un vecteur qui contient les distances vers les autres routeurs
 2. un routeur diffuse son vecteur avec ses routeurs voisins
- Il y a deux versions, la RIPv1 (originale, années 80) :
 - basée sur les classes d'adresses (*classfull*)
 - les mises à jours sont envoyées en *broadcast* (255.255.255.255)
- RIPv2
 - support *Classless Inter-Domain Routing* (CIDR)
 - envoi de mises à jour en *multicast* (224.0.0.9)
 - authentification (en md5)

Nous allons étudier une topologie avec sept routeur afin de mieux comprendre le principe du RIP



Au départ, les routeurs connaissent les distances en sauts vers les voisins

Information Stored at Node	Distance to Reach Node						
	A	B	C	D	E	F	G
A	0	1	1	∞	1	1	∞
B	1	0	1	∞	∞	∞	∞
C	1	1	0	1	∞	∞	∞
D	∞	∞	1	0	∞	∞	1
E	1	∞	∞	∞	0	∞	∞
F	1	∞	∞	∞	∞	0	1
G	∞	∞	∞	1	∞	1	0

Le routeur A connaît le coût d'aller à B, C, E et F

Destination	Cost	NextHop
B	1	B
C	1	C
D	∞	—
E	1	E
F	1	F
G	∞	—

A la fin, A connaît les coût vers tous les noeuds de la topologie

Destination	Cost	NextHop
B	1	B
C	1	C
D	2	C
E	1	E
F	1	F
G	2	F

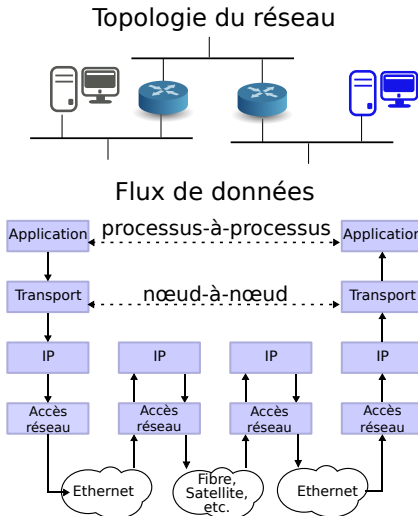
Nous avons la table suivante avec les vecteurs de tous le routeurs à la fin

Information Stored at Node	Distance to Reach Node						
	A	B	C	D	E	F	G
A	0	1	1	2	1	1	2
B	1	0	1	2	2	2	3
C	1	1	0	1	2	2	2
D	2	2	1	0	3	2	1
E	1	2	2	3	0	2	3
F	1	2	2	2	2	0	1
G	2	3	2	1	3	1	0

Liaison

La communication entre couches de nœuds voisins change selon le modèle de service offert (**TCP/IP**)

(Rappel)



Nous allons étudier les principes du service offert par la couche liaison

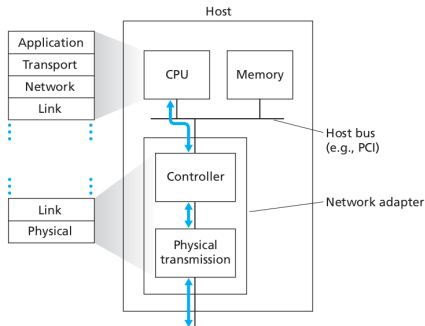
- Détection et correction d'erreurs
- Le partage d'un canal physique : multiples accès
- L'adressage de la couche liaison
- Instanciation et installation de technologies de lien variées

Le couche liaison est responsable du transfert entre nœuds physiquement adjacents par un lien physique

- Les canaux de communication qui connectent de nœuds adjacents sur un chemin de communication s'appellent **liens**
 - liens filaires
 - sans fil (e.g., WiFi)
 - *Local Area Network* (LAN)
- *Packet data unit* (PDU) de la couche liaison : **trame**, encapsule les datagrammes

Où est implémentée la couche liaison ?

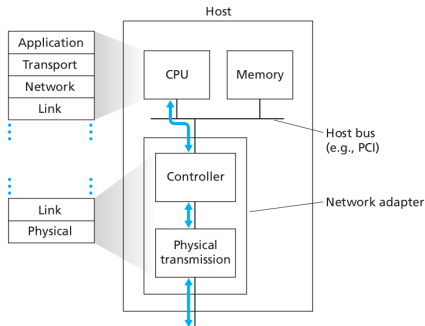
- Sur chaque nœud qui est interconnecté au réseau
- La couche liaison est implémenté sur un “adaptateur”, le *Network Interface Card* (NIC) ou sur un chip (e.g., carte Ethernet, carte 802.11, *chipset* Ethernet)



Source : Kurose, J., and K. Ross. Computer Networking: A Top Down Approach, 4e. Vol. 1. 2012.

Où est implémentée la couche liaison ?

- Le NIC/adaptateur est installé sur un bus du système du nœud et implémente la couche physique et la couche liaison
- Un mélange de logiciel, matériel, et *firmware*



Source : Kurose, J., and K. Ross. Computer Networking: A Top Down Approach, 4e. Vol. 1. 2012.

Quels sont les services qu'elle nous offre?

- Communication : *unicast*, *multicast*, et *broadcast*
- Accès au lien avec les trames
 - L'encapsulation de datagrammes en trames et ajout d'un en-tête et *trailer*
 - Accès partagé au canal de communication
- Transmission de données aux nœuds adjacents
 - contrôle de flux, détection d'erreurs, corrections d'erreurs, *half-duplex*, *full-duplex*

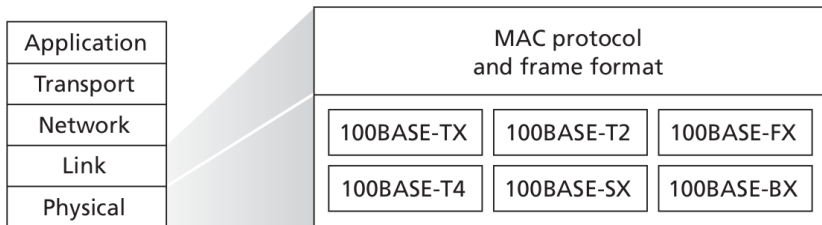
L'Ethernet est le protocole de la couche liaison le plus populaire

- Plus simple, populaire et moins chère que les autres technologies (e.g. ATM)
- Sans connexion et pas fiable
- Les champs de la trame Ethernet: données (46 à 1500 octets), adresse source (6 octets), type (2 octets, IP), *Cyclic redundancy check* (CRC) (4 octets, trame erronée est supprimée), *Preamble* (8 octets, synchronisation)



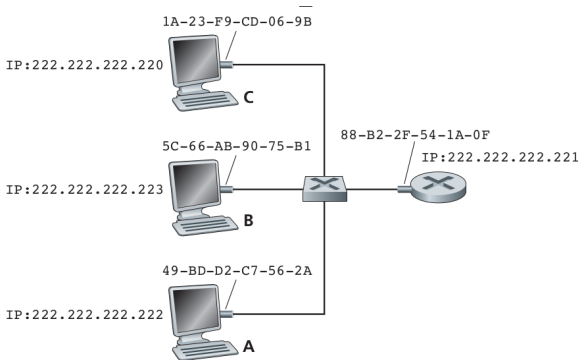
Il existe plusieurs standards d'Ethernet

- Ils ont le même protocole MAC et même format de l'en-tête
- Différentes vitesses : 2 Mbps, 10 Mbps, 100 Mbps, 1Gbps, 10Gbps
- Différentes support de lien physique : fibre, paire torsadée (UTP)



L'adresse *media access control* (MAC) est l'identifiant d'un nœud sur le réseau Ethernet

- Utilise la notation hexadécimal
- L'identifiant est unique : la première moitié correspond à l'identifiant du fabricant et la deuxième au numéro de série de la carte



Address resolution protocol (ARP) nous permet de trouver une MAC adresse à partir de l'adresse IP

- Tous les nœuds branchés au réseau Ethernet ont une table où chaque ligne contient :
 - Une correspondance entre IP et adresse MAC
 - *Time to live* est le temps après lequel la ligne sera supprimée
- Exemple :
 1. **A** souhaite envoyer un datagramme (IP) à **B**, mais **A** ne connaît pas l'adresse MAC de **B**
 2. **A** envoie un paquet en *broadcast* qui contient une requête ARP demandant l'adresse MAC de **B**
 3. **B** reçoit la requête de **A**, enregistre son adresse MAC dans sa table de correspondance et envoie une réponse à **A** avec son adresse MAC
 4. **A** reçoit la réponse de **B** et enregistre le MAC adresse de **B** dans sa propre table

Le commutateur Ethernet est transparent pour les nœuds et ne nécessite pas d'être configuré

- Il *store and forward* trames (use de la mémoire)
- Examine l'adresse MAC de la destination d'une trame afin de faire suivre d'une façon sélective
- Utilise du CSMA/CD pour partager le lien physique
- Il est complètement transparent pour les nœuds du réseau
- Le commutateur ne nécessite pas d'une configuration particulière pour être utilisé

Le commutateur maintient une table qui lui permet de faire suivre les trames

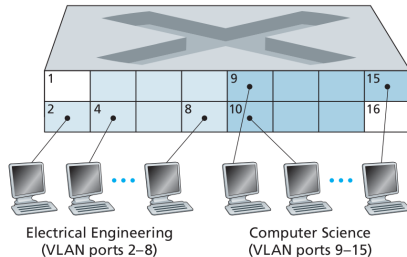
- Chaque ligne de la table contient un adresse MAC et un port, un adresse MAC et le temps de vie de la ligne
- Au début, la table est vide (pas besoin de configuration)
- Le commutateur apprend tout seul les adresses MAC (source) de nœuds et les ports qu'ils ont utilisent pour échanger de trames
- Le temps de vie est mis à zéro lors qu'une trame est échangée. La ligne est supprimé si l'adresse MAC est inactive pendant un certain temps

Nous pouvons lister les similarités entre routeurs et commutateurs

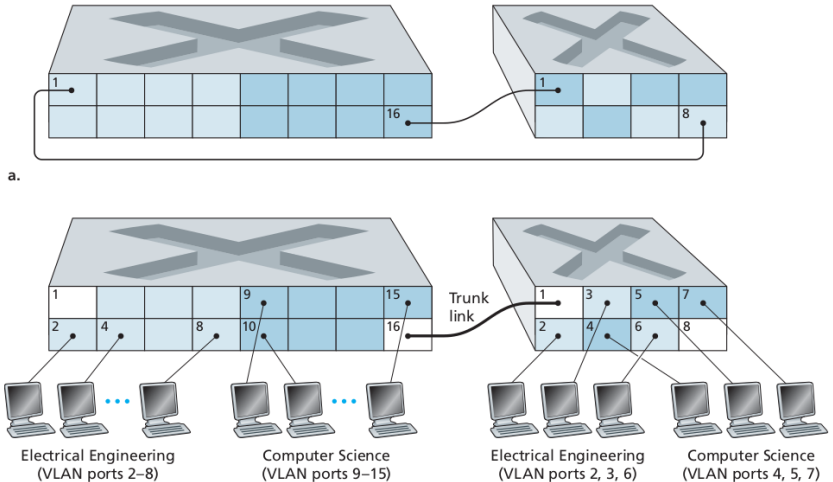
- Le paquet traverse ces équipements d'une façon similaire : *store-and-forward*
 - Routeurs : vérifient les en-têtes de la couche réseau
 - Commutateurs : les en-têtes de la couche liaison
- Tous les deux maintiennent une table qui leur permet de faire suivre les paquets
 - Routeurs : mettent à jour les tables avec les protocoles de routage, adresses IP
 - Commutateurs : apprennent leurs tables (i) en utilisant le *broadcast* pour les paquets dont les destinataires sont inconnus puis (ii) en enregistrant les adresses des nœuds sources de trames

La fonctionnalité *virtual LANs* (VLAN) d'un commutateur nous permettent de définir plusieurs LANs sur le même nœud

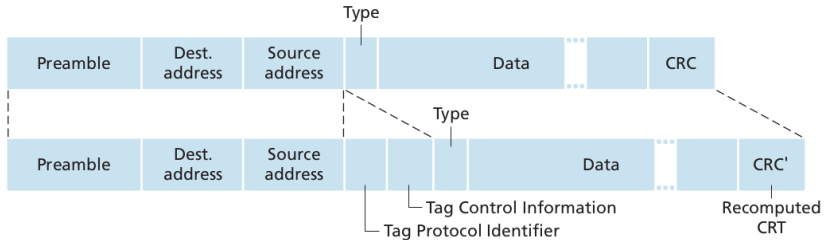
- Quand le lien physique est partagé, il y a un seul domaine de *broadcast*
 - tout le trafic de *broadcast*(ARP, DHCP, MAC adresses inconnus)
 - éventuels problèmes de sûreté/vie privé/performance
- Nous regroupons de sous-ensemble de ports d'un commutateur pour définir une VLAN



Nous pouvons regrouper aussi les ports de différents de commutateurs sur un seul VLAN



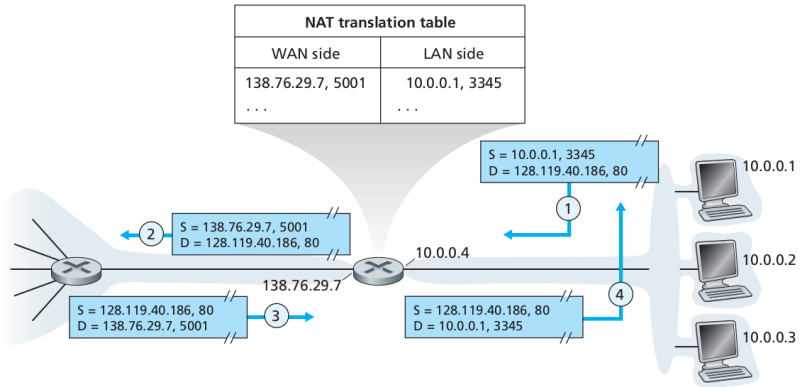
Le nouveau format de la trame pour supporter la communication par le *trunk* (802.1Q-tagged Ethernet)



Source : Kurose, J., and K. Ross. Computer Networking: A Top Down Approach, 4e. Vol. 1. 2012.

Network Adresse Translation

Le NAT nous permet d'utiliser un seul IP pour représenter un ensemble de nœuds d'un réseau

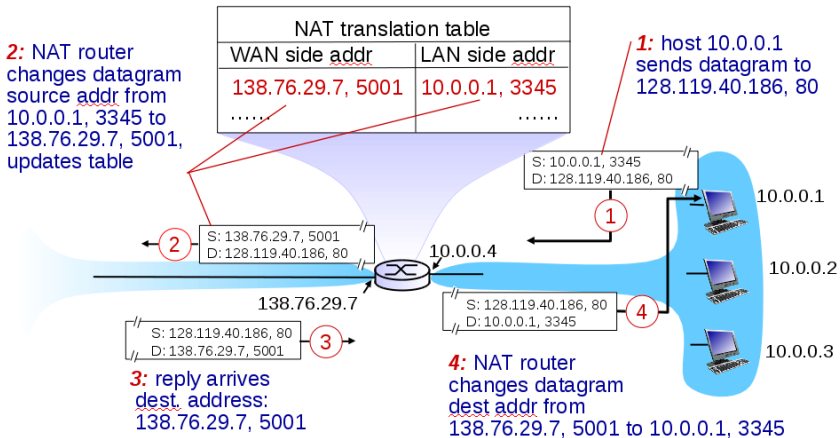


Source : Kurose, J., and K. Ross. Computer Networking: A Top Down Approach, 4e. Vol. 1. 2012.

Le NAT est très utilisé pour faire la traduction d'adresses privées d'un réseau local vers un IP publique (Internet)

- Pas besoin d'une plage d'adresse IP publique
- Flexibilité pour changer de fournisseur d'accès
- Les nœuds du réseau privé ne sont pas visibles sur Internet, donc ils sont un petit peu plus protégés

Le NAT maintient une table avec flux correspondants



Réseau Informatique

Guthemberg Silvestre

Lundi, le 20 juin 2016



ECOLE NATIONALE DE L'AVIATION CIVILE