

# Une Introduction au Réseau Informatique

**Guthemberg Silvestre**

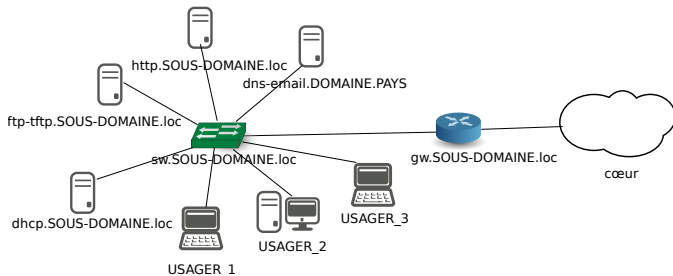
Lundi, le 20 juin 2016



ECOLE NATIONALE DE L'AVIATION CIVILE

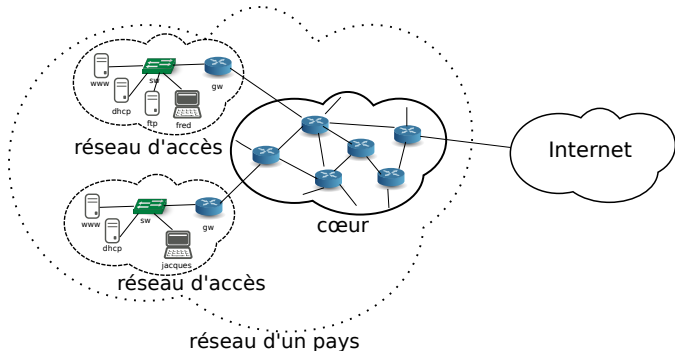
# Les ordinateurs des usagers sont branchés aux réseaux d'accès

- 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**



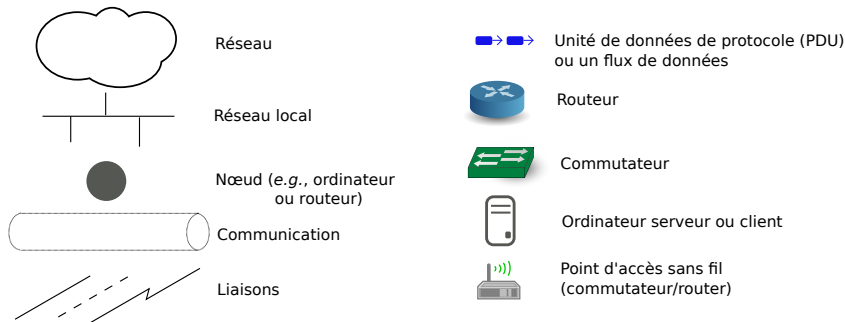
# Qu'est-ce que nous allons apprendre?

- 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.

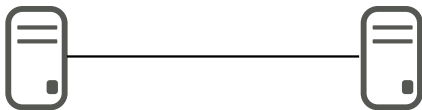
## Le réseau informatique transmet des informations génériques

- 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.*

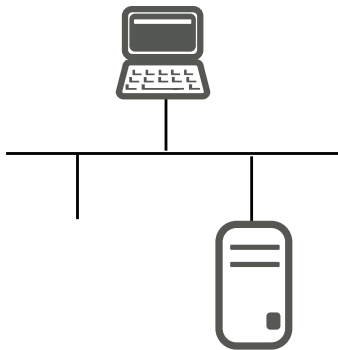
# Nous utiliserons des images type pour représenter les éléments d'un réseau



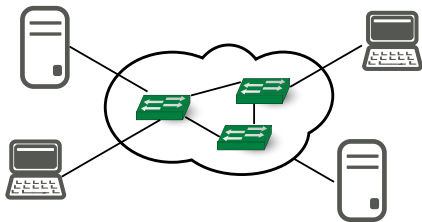
Une représentation logique d'un réseau entre deux nœuds



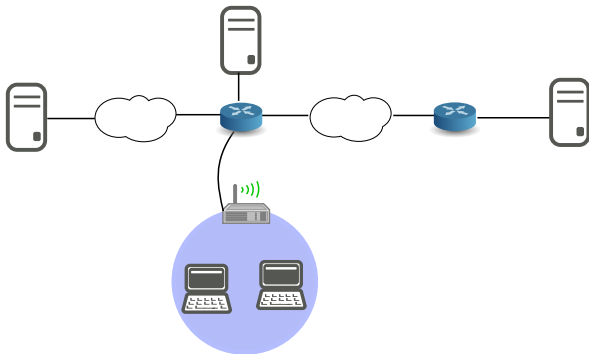
## Une autre représentation logique d'un réseau local



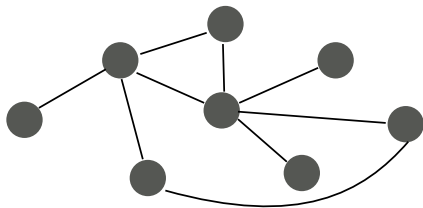
Le réseau local peut être constitué de plusieurs commutateurs



Nous pouvons représenter plusieurs réseaux interconnectés

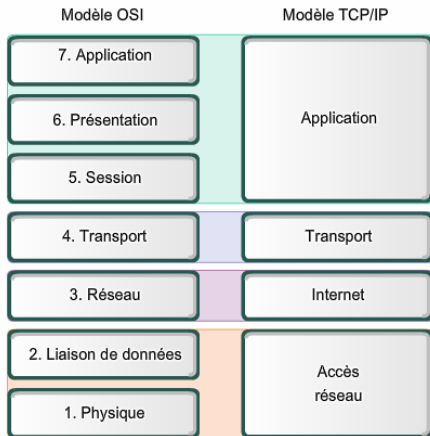


Nous pouvons représenter les nœuds et les liens d'un réseau en forme de graphe

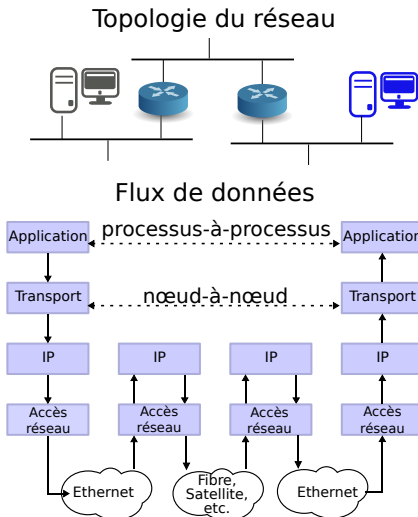




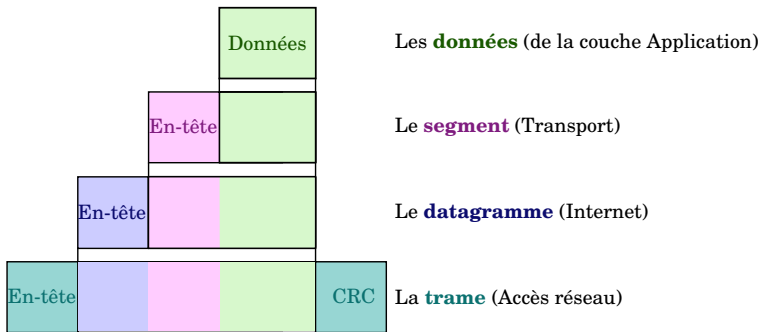
Les services de la pile de protocoles de l'Internet, le TCP/IP, sont organisés d'une façon modulaire



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

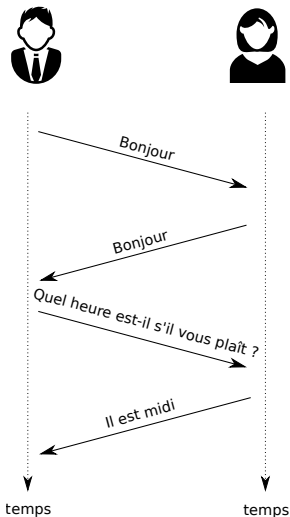


Chaque couche **offre un service** et **encapsule** les données de la couche qui est juste au-dessus d'elle

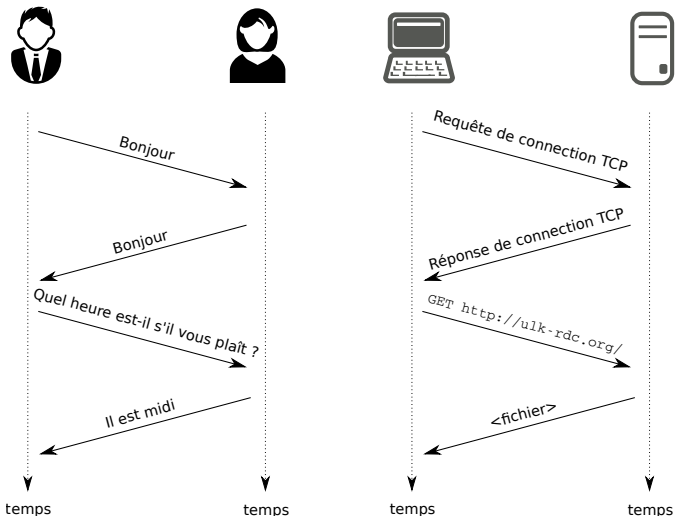


Les services sont définis par les protocoles de chaque couche

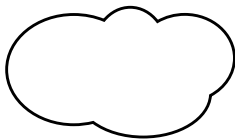
Un protocole réseau nous permet de formaliser une communication afin d'offrir un service



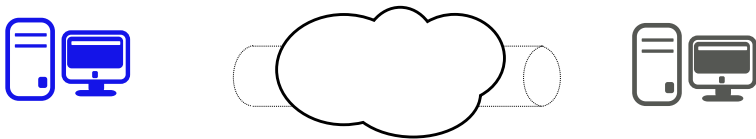
# Un protocole réseau nous permet de formaliser une communication afin d'offrir un service



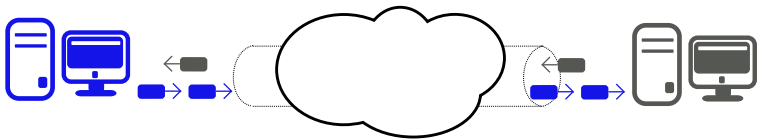
La plupart des applications utilisent des protocoles afin d'échanger des données d'une façon fiable



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



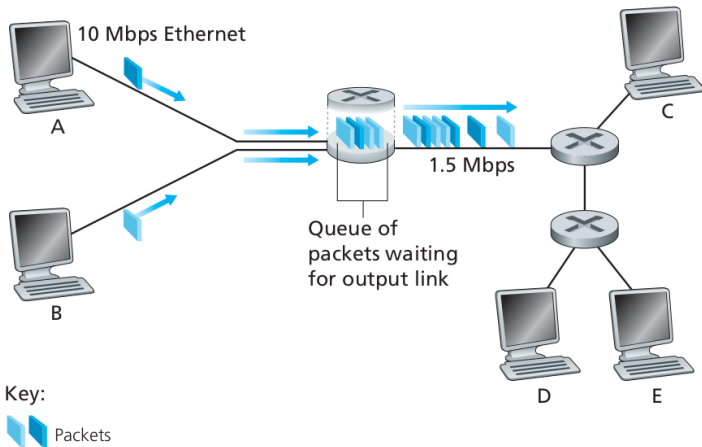
- Le réseau sert à écrire et lire des données d'une application
- Le modèle des communication prédominant permet un échange de données :
  - qui est fiable;
  - qui marche dans les deux sens;



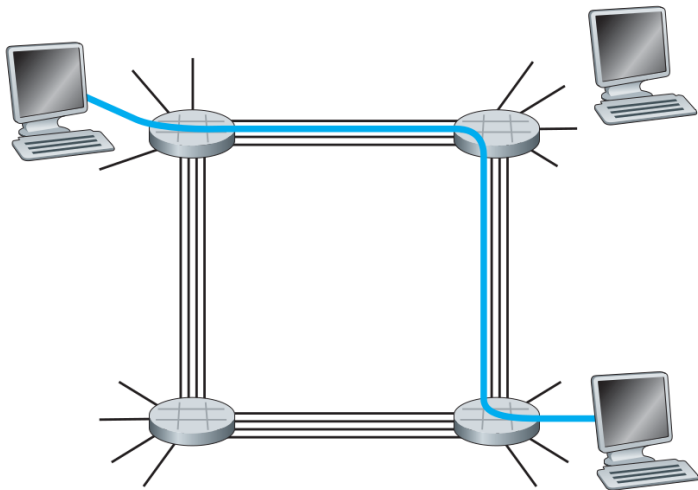
# Les protocoles nous permettent de partager les ressources du réseau informatique entre les nœuds

- 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

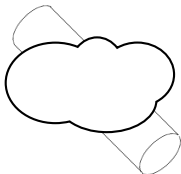


Client

- 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



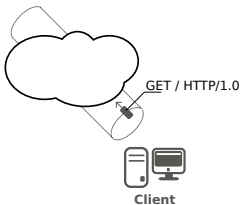
Client

- 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



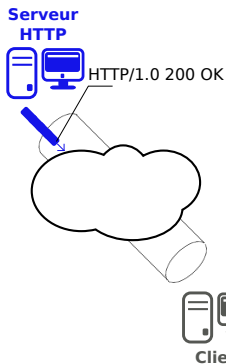
# Une requête GET nous permet de solliciter un contenu

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 envoie 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

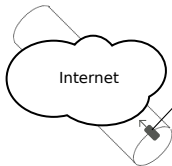


Client

- 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" />

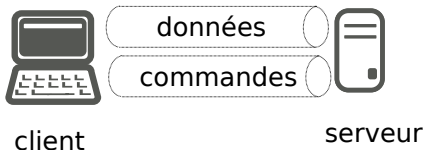
...

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

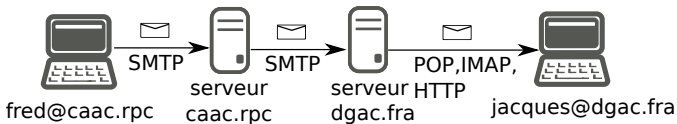
## Le *File Transfert Protocol* nous permet de stocker ou partager de fichiers sur le réseau

- 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

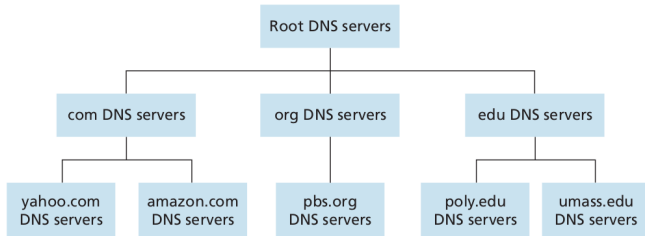
- Il y a trois composants principaux :
  - l'agent client : les applications qui nous permettent de rédiger, envoyer, recevoir et lire les e-mails ;
  - le serveurs d'e-mails : jouent un rôle de relai entre l'émetteur et le destinataire d'un message ;
  - *Simple Mail Transfert Protocol* (SMTP) crée 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)
<b>A</b>	identifiant IP	le nom d'un nœud
<b>NS</b>	le serveur qui répond pour ce domaine	le nom du domaine
<b>CNAME</b>	le vrai nom	un alias du nœud
<b>MX</b>	le serveur SMTP	le domaine de l'email