

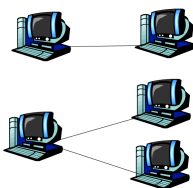
IN606 – Fiche de révision

Classification des réseaux

Name	Abréviation	Couverture	Débit	Exemples
Réseaux Personnels	PAN	10-100m	quelques Mbits/s	Bluetooth, Homeref, ...
Réseaux Locaux	LAN	100-1000m	quelques dizaines de Mbits/s	Ethernet, Token Ring, Wifi, HipperLan
Réseaux Métropolitains	MAN	la taille d'une ville	quelques dizaines Mbits/s	FDDI, DQDB, ATM, WiMax, ...
Réseaux Locaux	WAN	Mondiale	quelques Mbits/s	RNIS, IP, ATM, WATM, GSM, ...

Topologies

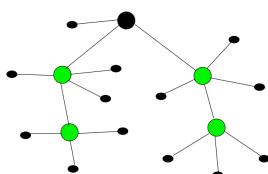
- Logique: le mode d'échange des messages dans le réseau
- Physique: raccordement des machines → deux types de liaisons: point-à-point ou multipoints



Topologies – Physique

- Bus
- Etoile
- Anneau

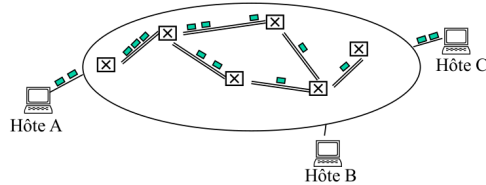
Topologies construites : dérivés des réseaux en étoiles



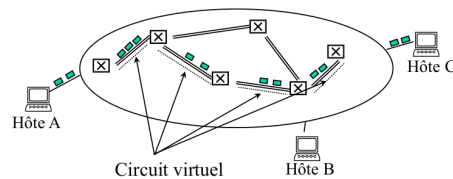
Modes de mise en relation

Deux modes de fonctionnement pour transiter les informations

- Mode non connecté (Datagramme)
 - Une seule phase (Transfert des données)
 - Simple
 - Plusieurs chemins possibles



- Mode connecté (Circuit virtuel, CV/circuit physique)
 - Etablissement d'une connexion
 - Transfert des données
 - Libération de la connexion
 - Service fiable
 - Complexe
 - Chemin dédié
 - Circuits commutés (SVC)
 - ou Circuits permanent (PVC)



- Technique de commutation la manière d'interconnecter 2 correspondants
- Le fonctionnement d'un nœud (routeur/switch)
- Nombre de liens = $N(N+1)/2$ (N: nombre de nœuds)
- Temps de traversée du réseau $T_p : T_p = (L + pH) \frac{1+N}{D}$

L : longueur de message

N : nombre de nœuds

p : nombre de paquets

H : entête protocole

D : débit

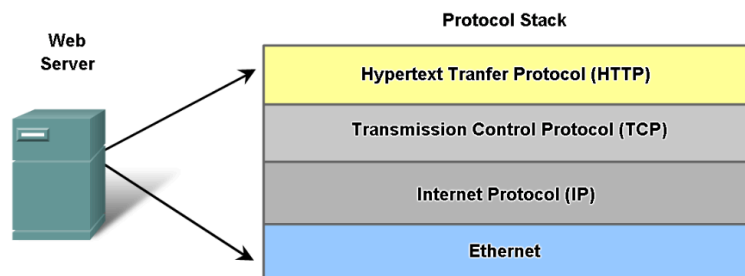
- Techniques de commutation
 - Commutation de circuit ($p = 1, N = 0$)
 - Commutation de messages ($p = 1, N > 0$)
 - Commutation de paquets
 - Commutation de trames
 - Commutation de cellules

Le réseau Coeur

- Réseau maillés de routeurs
- **La question fondamentale :**
Comment les données sont transférées à travers le réseau ?
 - **Commutation de circuits :**
circuit dédié par appel : réseau téléphonique
 - **Commutation de paquets :**
données envoyées sur le réseau par “morceaux”

Fonction du Protocole dans le réseau de communication

- L'importance des protocoles et comment ils sont utilisés pour faciliter la communication sur les réseaux de données
 - Un protocole est un ensemble de règles prédéterminées
- Les protocoles réseau sont utilisés pour permettre aux périphériques de communiquer avec succès
- Les suites du protocole et les normes de l'industrie
 - Une norme est un processus ou un protocole qui a été approuvé par l'industrie des réseaux et ratifié par un organisme de normalisation
- Interactions entre protocoles

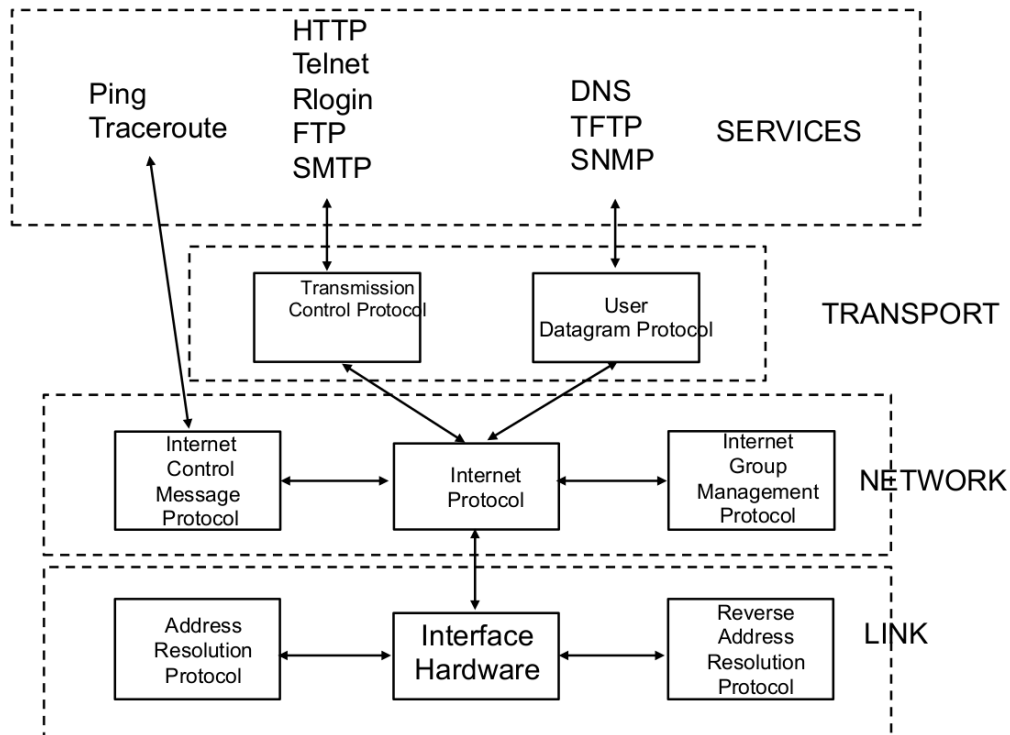


- Indépendance du matériel (dispositif)
 - Beaucoup de divers types de dispositifs peuvent communiquer en utilisant les mêmes ensembles de protocoles. C'est parce que les protocoles spécifient les fonctionnalités du réseau, pas la technologie sous-jacente à la charge cette fonctionnalité.

Les couches de protocoles : modèle OSI et modèle TCP/IP

Couches OSI	Fonctions	Modèle TCP/IP
Application	Applications réseaux : transfert de fichiers.	Application
Présentation	Formatage et cryptage des données	
Session	Etablissement et maintien des sessions	
Transport	Transport de bout en bout, fiable et non fiable	Transport
Réseau	Envoi et routage des paquets de données	Internet – Interface
Liaison	Transfert des unités d'informations et contrôle d'erreurs	Matériel
Physique	Transmission des données binaires	

Application	File Transfer	Mail	Terminal Emulation	File Transfer	Client Server	Réseau Mgmt
Présentation	FTP	SMTP	TELNET	TFTP	NFS	SNMP
Session						
Transport	TCP			UDP		
Réseau	ARP --- RARP	ARP --- RARP --- IP	IP		IP --- ICMP	ICMP
Liaison	Ethernet – Token ring – Starlan – Arcnet – FDDI – SMDS					
Physique	TP – STP – FO – Satellite – Micro-ondes					



SMTP	TELNET	FTP	HTTP
DNS	RIP	OSPF	SNMP
PING	NFS	X11	
TCP	UDP	SPX	
IPv4 - IPv6	ICMP - IGMP	IPX	
ARP - RARP	ATM	SLIP PPP	
		RTC ISDN	

Les couches et les données Encapsulation/Décapsulation

Lors du passage d'une couche à l'autre des **en-têtes** (Headers) sont utilisés pour encapsuler le message.

Couche	État	En-tête courante
Application	Message	M
Transport	Segment	$M-H_t$
Réseau	Datagram	$M-H_t-H_r$
Liaison	Frame	$M-H_t-H_r-M_l$
Physique		----

Applications réseau

Processus : Programme s'exécutant sur un hôte (host).

- Sur le même hôte, deux processus interagissent en utilisant la **communication interprocessus**.
- Processus sur deux machines différentes communiquent par message à travers un protocole de la **couche application**

Processus distribués :

- E-mail, Web, telnet
- S'exécutant sur les terminaux (hosts)
- Échangent des messages pour implémenter l'application

Agent utilisateur : Interfaces avec l'utilisateur au dessus et le réseau en dessous.

- Implémentations interface utilisateur et le protocole du niveau application
 - Web : navigateur
 - E-mail : lecteur de mail
 - Streaming audio/vidéo: lecteur multimédia

Protocole de la couche application :

- Définissent des messages échangés par les applications et les actions prises
- Utilisent les services de communication fournis par les protocoles de la couche inférieure (**TCP**, **UDP**)

Protocole de la couche application

- Types de messages échangés, exemple → Messages de demande et de réponse
- La syntaxe adoptée par les différents types de message : soit les différents champs qu'il contiennent et leur délimitation
- La sémantique des différents champs, c'est à dire le sens des informations qu'ils contiennent
- Les règles utilisées pour déterminer quand et comment un processus doit envoyer ou répondre à un message

Protocoles domaines publics :

- Définis dans les RFCs
- Permettent l'interopérabilité
- **HTTP**, **SMTP**, ...

Protocoles propriétaires :

- **KaZaA**, ...

Paradigme Client-Serveur

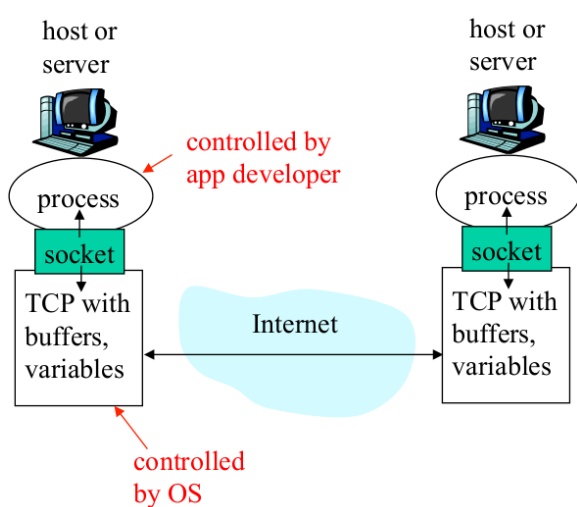
Client

- Initialise le contact avec le serveur
- Demande de service au serveur
- Web: client implémenté dans le navigateur, e-mail dans lecteur de mails

Serveur

- Fournit le service demandé par le client
- Web server envoie la page Web demandée, mail server délivre e-mail

Processus de communication à travers le réseau



- **API** → le choix du protocole de transport
- La possibilité de définir quelques paramètres

Processus d'adressage :

- Un **processus local** à besoin d'identifier le **processus distant**
- Chaque host à une unique adresse **IP**
- Plusieurs processus peuvent être exécutés sur la même machine. L'identificateur inclut l'**IP** et le numéro de **port** associé à l'hôte.

Services nécessaires à une application

Data loss (transfert fiable)

- Certaines apps (audio, ...) tolèrent la perte de paquets
- D'autres apps (file transfer, telnet, ...) exigent 100% transfert des paquets

Contraintes de temps

- Certaines apps (Internet telephony, interactive games) demandent un bas délai

Bandwidth (débit)

- Certaines apps (multimédia, téléphonie par internet, ...) exigent un débit minimal disponible
- Autres apps ("elastic apps, comme ftp et web") peuvent s'adapter aux débits disponibles

Fonctionnement de quelques applications de réseau

Application	Data loss tolerant	Bandwidth	Time Sensitive
File transfer – Mails – Web documents	Non	Elastic	Non
Realtime audio/ video	Oui	audio : 5kbps-1Mbps	100's msec
Stored audio/video	Oui	10kbps-5Mbps	few secondes
Interactive games	Oui	10kbps-5Mbps	100's msec
Instant messaging	Non	few kbps up	Oui et Non

DNS : Domain Name System

Le **DNS** est simplement un surnom utilisé pour désigner des les adresse **IP** → google.com au lieu de 001011100...

DNS :

- **Datatable distribuée** implémentée en hiérarchie dans plusieurs name servers
- **Protocole couche application** host, routeurs, name servers communiquent pour résoudre noms (translation adresse/ name)

Pourquoi le DNS est décentralisé ? Problèmes classiques de débits, scalabilité et de maintenance de la base de données.

Name serveur locaux :

- Chaque ISP a son **local default serveur**
- Host **DNS** consulte en premier le name serveur local

Name server de source autorisée :

- Pour un hôte : stocke son adresse IP, nom
- Peut accomplir la translation name/address IP

Electronic Mail

