

Paradigmes de Programmation & Introduction aux systèmes distribués

Par Mustapha Boushaba

mboushaba@cegepsth.qc.ca

Paradigmes de Programmation

- **Paradigme :**

"Modèle théorique de pensée qui oriente la recherche et la réflexion scientifiques" (Larousse).

.

- **Programmation:**

"Ensemble des activités qui permettent l'écriture des programmes informatiques. Elle représente usuellement le codage, c'est à dire la rédaction du code source d'un logiciel."

Paradigmes de Programmation

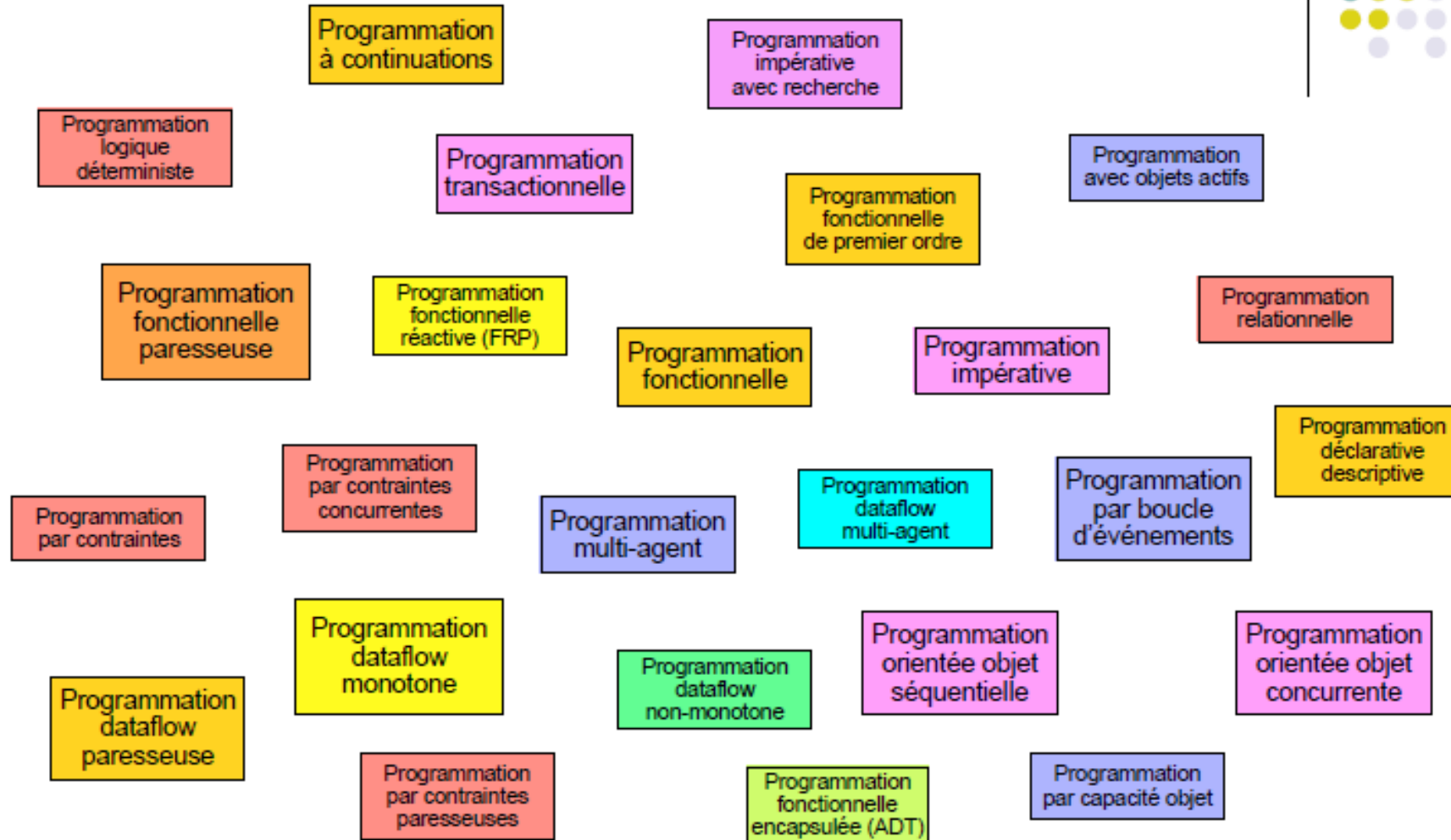
- Langage :

"Une abstraction des opérations réalisée par un ordinateur. Un langage est habituellement domine par un paradigme a partir duquel sa structure a été conçue. Souvent, des éléments issus de différents paradigmes y sont présents."

- Programmer:

"Utiliser un langage afin de se conformer de façon plus ou moins rigoureuse a un ou plusieurs paradigmes de programmation."

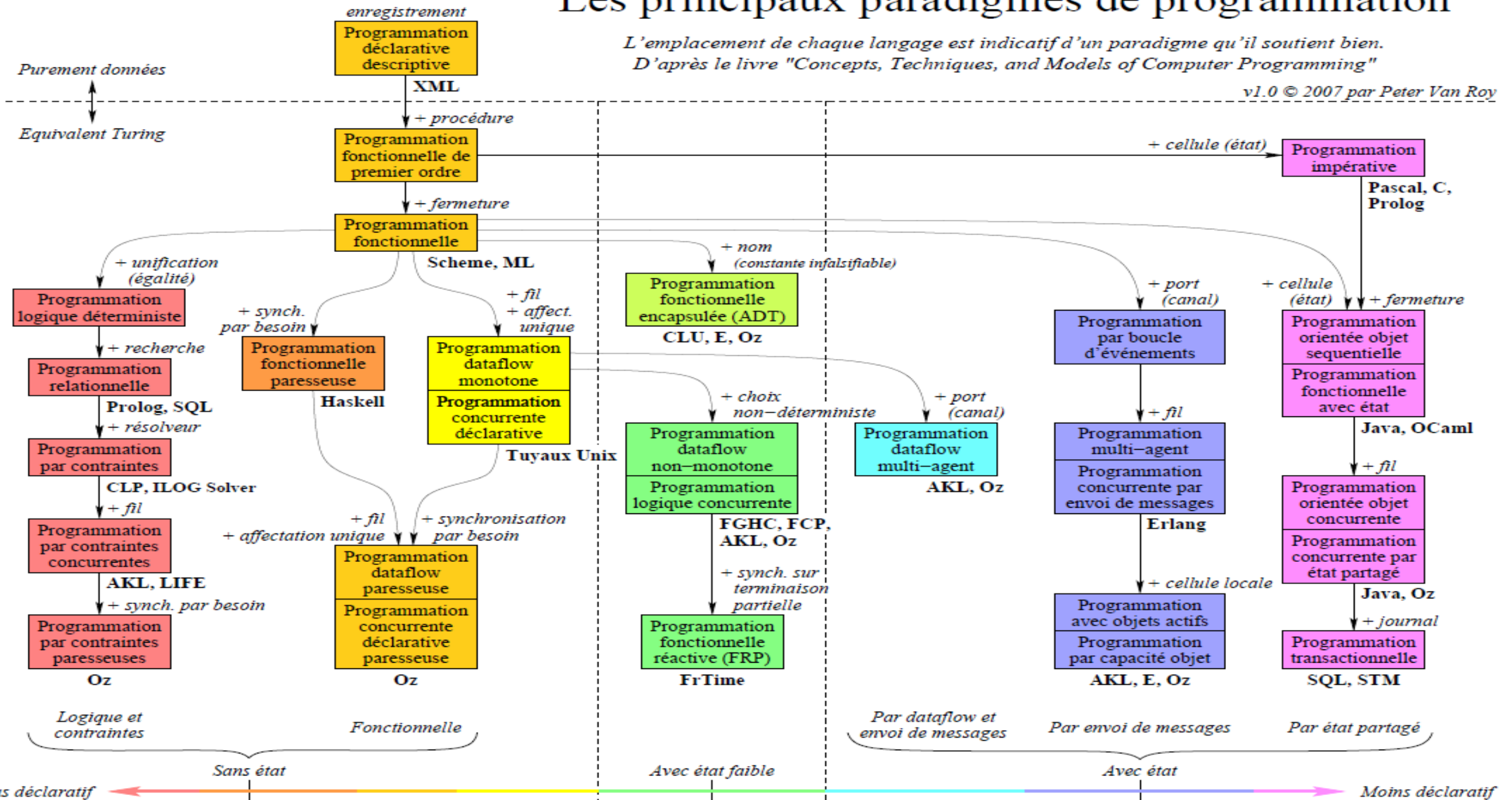
Il y a beaucoup de paradigmes



Les principaux paradigmes de programmation

*L'emplacement de chaque langage est indicatif d'un paradigme qu'il soutient bien.
D'après le livre "Concepts, Techniques, and Models of Computer Programming"*

v1.0 © 2007 par Peter Van Roy



- [https://fr.wikipedia.org/wiki/Paradigme \(programmation\)](https://fr.wikipedia.org/wiki/Paradigme_(programmation))
- (voir fichier doc distribué)

Introduction aux systèmes distribués

- Les systèmes informatiques ont fait l'objet d'une grande évolution
 - Depuis 1945: l'ère informatique moderne
 - Jusqu'en 1985 l'ère des grand calculateurs (volumineux et coûteux).
 - => conséquence: peu d'ordinateurs en compagnies, ordinateurs individuels (non connectés)
 - Au milieu des années 1980, deux avancées technologiques vont changer la situation:
 - **Développement et production en quantité de microprocesseurs puissants (8bits, 16bits etc..)**
 - Ordinateurs de grande taille mais à prix réduit. On est passé d'une machine de 10 million de dollars exécutant 1 instruction par seconde vers des machines qui exécutent 10 million d'instructions par seconde et qui coutent 1000 dollars

Introduction aux systèmes distribués

- **Développement des connexions à hautes vitesse**
 - Réseaux locaux (LAN)
 - Interconnexion d'un grand nombre de machines, transfert de l'informations entre 10 et 100 Million de bits/seconde
 - Réseaux étendus ou WAN
 - permettent à des millions de machines se trouvant partout dans la terre de se connecter à des vitesses variant de 64 Kbps (kilobits par seconde) à gigabits par seconde
- **Résultat: maintenant il est possible et facile même à mettre en place des systèmes composés d'un grand nombre de processeurs connectés par un réseau à grande vitesse de calcul. => généralement appelés systèmes distribués**

Qu'est-ce qu'un système distribué?

- Diverses définitions des systèmes distribués dans la littérature
- Selon Tanenbaum,

Un système distribué est un ensemble d'ordinateurs indépendants qui apparaît à un utilisateur comme un système/ordinateur unique et cohérent

Cette définition a plusieurs aspects importants:

- Les composants (à savoir, des ordinateurs) sont autonomes.
- les utilisateurs (qu'ils soient des personnes ou des programmes) pensent qu'ils ont affaire à un seul système. Cela signifie que l'un ou l'autre des composants autonomes doivent collaborer.

Avantages des systèmes distribués par rapport aux systèmes centralisés

- Économique
 - Excellent rapport performance/prix des microprocesseurs
 - Loi de Grosch : La puissance de calcul d'un processeur est proportionnelle au carré de son prix. N'est plus applicable
- Puissance de calcul
 - Puissance de calcul supérieure observée dans un système pluri-processeur par rapport à un système à un seul processeur

Avantages des systèmes distribués par rapport aux systèmes centralisés

- Distribution naturelle de certaines applications
 - travail coopératif assisté par ordinateur (TCAO): exemple, ensemble d'employés qui collaborent pour produire des rapports conjoints
 - jeux coopératifs assistés par ordinateur
- Haute disponibilité
 - Un problème d'une machine n'affecte pas les autres
- Évolution progressive:
 - Une entreprise dont la charge de travail augmente risque de voir la capacité de son mainframe atteinte. Elle n'a que de le remplacer par un autre de plus grande capacité (si elle existe) ou d'ajouter un deuxième mainframe. Les deux peuvent causer des grands problèmes dans les opérations de l'entreprise.

Avantages des systèmes distribués par rapport aux systèmes centralisés

- À long terme , la principale force des systèmes distribués sera le grand nombre d'ordinateurs personnels et la nécessité aux gens de travailler ensemble et de partager des informations d'une manière pratique, sans être restriction de position géographique des personnes, des données et des machines .

Avantages des systèmes distribués Par rapport à des postes de travail indépendants

- partage de données entre les utilisateurs
 - exemple : système de réservation chez les compagnies aériennes
- partage de périphériques coûteux
 - exemple : imprimante laser couleur, périphériques d'archivage massif (exemple, jukeboxes optique)
 - https://en.wikipedia.org/wiki/Optical_jukebox
- Facilité des communications entre personnes
 - courrier électronique
 - communication asynchrone
 - modification possible des documents échangés
- flexibilité (distribution de la charge entre les machines disponibles de manière à rentabiliser l'action)

Inconvénients des systèmes distribués

- Logiciel
 - On ne dispose que de peu d'expérience dans la conception, la mise en œuvre et l'utilisation de logiciels distribués. Quels types de systèmes d'exploitation, des langages de programmation, et des applications sont appropriées pour ces systèmes?
 - la distribution doit elle être transparente aux utilisateurs ?
- Réseau de communication
 - Saturation du réseau: penser à changer vers des types plus avantageux (fibre optique)
 - perte de messages
- Sécurité
 - facilité de partage de données rend risque de produire des intrusions et des violation des données confidentielles. Une protection est obligatoire.

Exigences des systèmes distribués

- Le besoin d'utiliser un système distribué est souvent dérivé d'exigences non fonctionnelles soit :
- Extensibilité: les systèmes distribués permettent facilement une expansion si nécessaire ;
- Ouverture – les composantes des systèmes distribués possèdent des interfaces bien définies ce qui leur permet d'être facilement extensibles et modifiables. Les services web sont un exemple de système distribué qui possède une grande ouverture ;
- hétérogénéité – les composantes peuvent être écrites en différents langages sur différentes machines. Exemple, les éléments d'un serveur peuvent être programmés en C++ et s'exécuter sous Unix, alors que le client peut être en Java et s'exécuter sous Windows ;

Exigences des systèmes distribués

- Accès aux ressources et partage – les systèmes distribués fournissent un moyen de partager les ressources, c'est-à-dire à la fois le matériel, le logiciel et les données ;
- Tolérance aux pannes – les systèmes distribués peuvent être plus tolérants aux pannes que les systèmes centralisés, car ils permettent de répliquer facilement les composantes.

Champs d'application

- Ingénierie simultanée
 - Coopération d'équipes pour la conception d'un produit
 - Production coopérative de documents
 - Partage cohérent d'information
- Gestion intégrée des informations d'une entreprise
 - Intégration de l'existant
- Contrôle et organisation d'activités en temps réel
 - Workflows
 - Travail coopératif
- Centres de documentation, bibliothèques
 - Recherche, navigation, visualisation multimédia

Source: http://www.i3s.unice.fr/~nlt/cours/licence/pdist/pdist1_slides.pdf

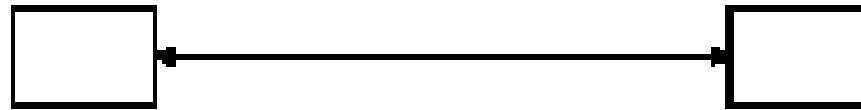
Besoins des applications

- Ouverture: Interopérabilité, portabilité, fédération ; réutilisation de l'existant
- Coopération, coordination, partage
- Vision commune cohérente d'informations partagées (globalement, par groupes).
Interaction en temps réel, support multimédia
- Transparence: Accès (mobilité des usagers avec préservation de l'environnement),
localisation (de l'information, des services, ...)
- Qualité de service: Disponibilité, délais, coûts, qualité de perception, .. avec niveau garanti
- Sécurité: Authentification, intégrité, confidentialité, ...
- Évolutivité: Reconfiguration, gestion dynamique de services

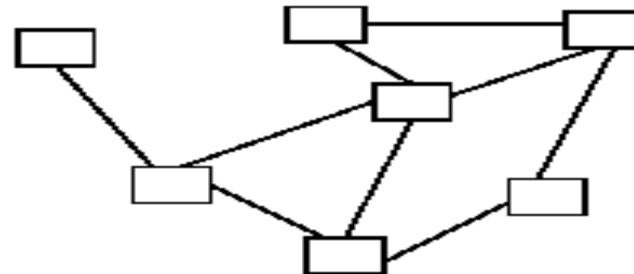
La communication en réseaux

La communication de données Vs. Réseautage

- **La communication de données:** deux noeuds qui communiquent.



- **Réseautage:** deux ou plusieurs noeuds qui communiquent.



Modèle Simple de Communications

- Les agents (utilisateurs, programmes, capteurs, etc.) veulent transmettre une information m .
- L'information prend plusieurs formes, il faut alors la convertir en données:
 - **données:** une représentation de faits, de concepts ou d'instructions, formalisée d'une façon convenable pour le traitement par des machines ou des humains.
 - **information:** l'interprétation (le sens) attribuée aux données.
- Les données doivent être converties pour la transmission de l'information en une forme convenable.

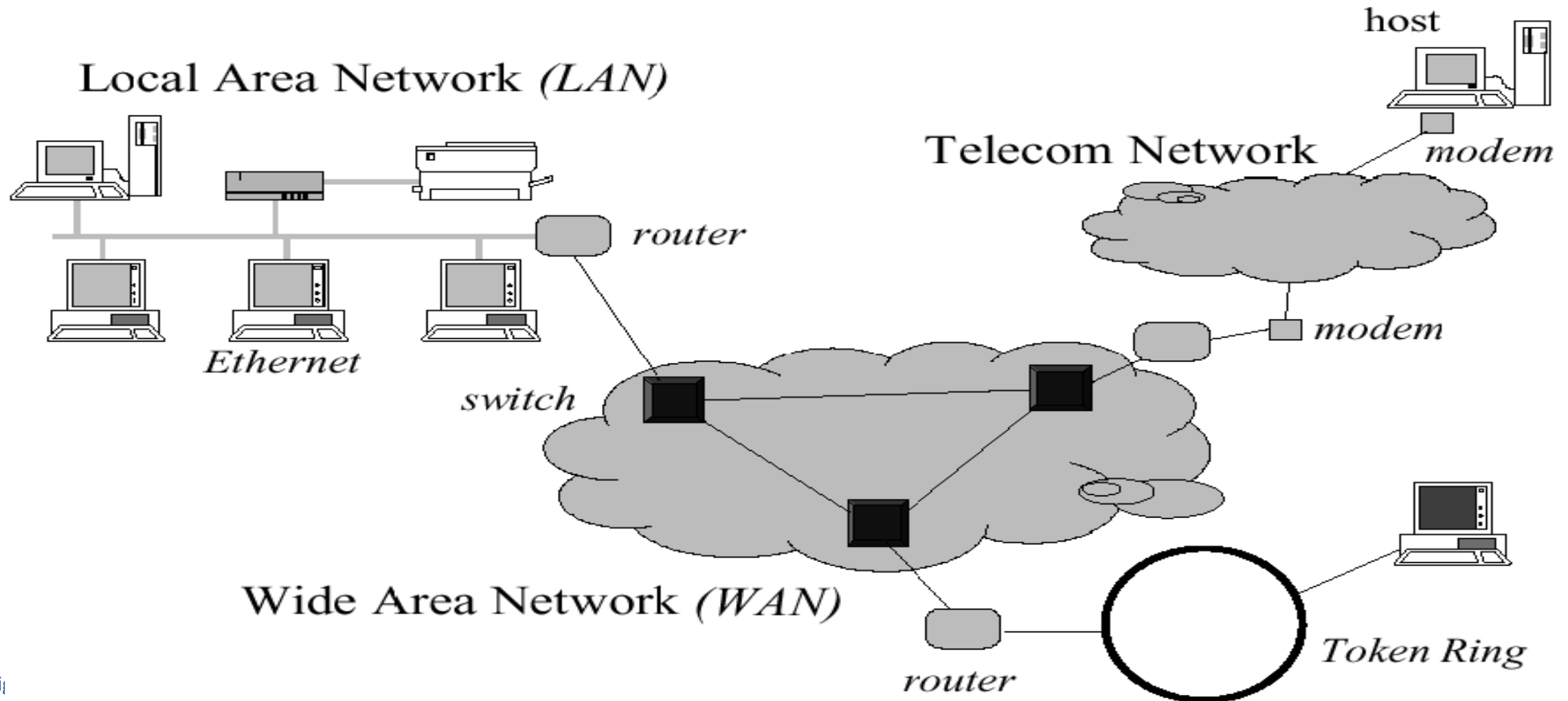
Taches Principales

- Génération de signaux
- Transmission de signaux
- Synchronisation
- Adressage et routage
- Établissement/rupture de lignes de communication
- Détection d'erreurs et correction
- Contrôle de flux
- Codage de messages
- Protection

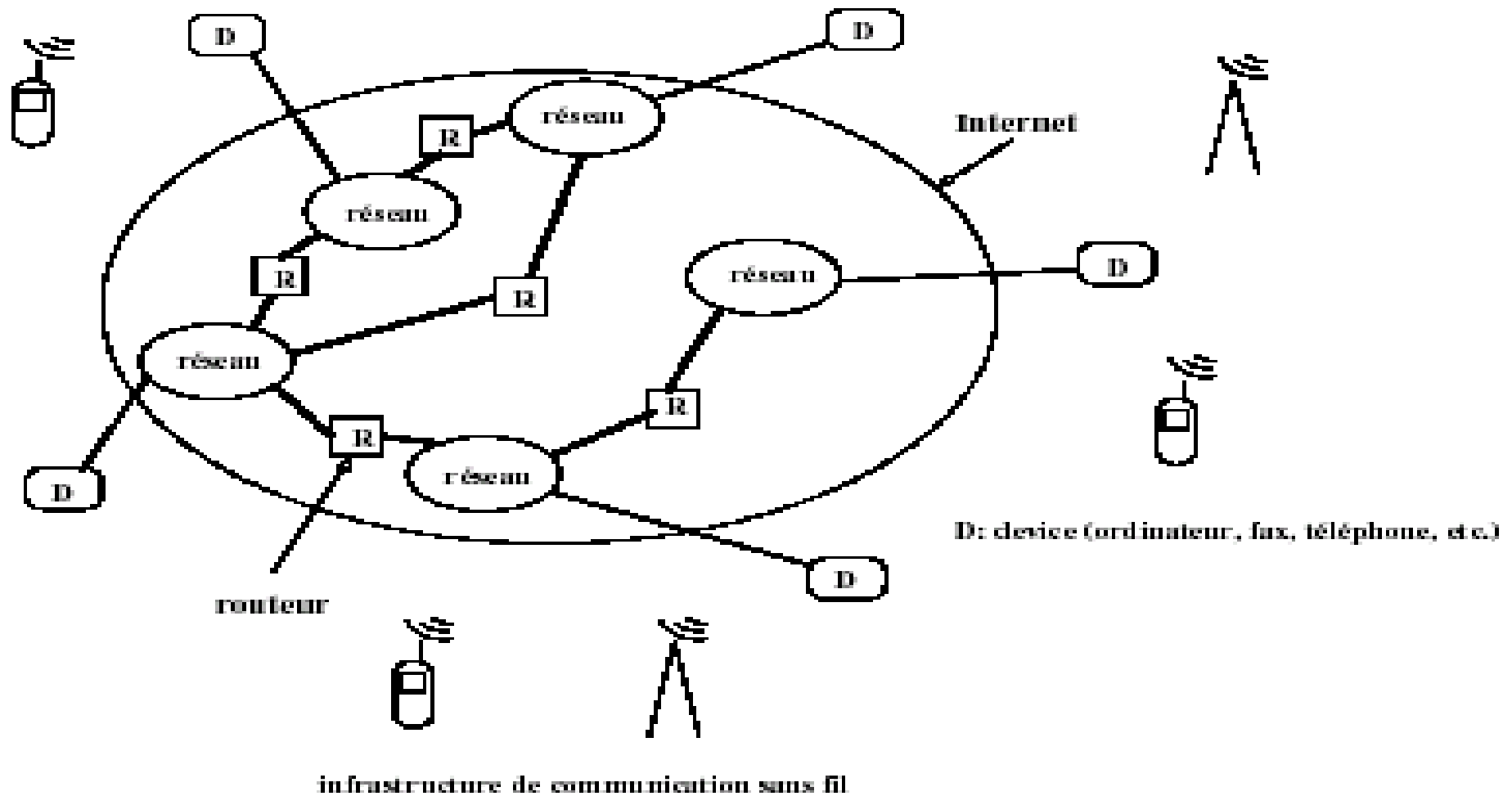
Réseaux d'ordinateurs et systèmes distribués

- Réseau d'ordinateurs
 - Un réseau d'ordinateurs est un ensemble d'ordinateurs autonomes interconnectés.
- Système distribué
 - Un système distribué est un système qui apparaît à l'utilisateur comme une infrastructure à un seul processeur virtuel unique. L'utilisateur n'a pas connaissance de la multiplicité des ordinateurs

Un exemple de Réseau



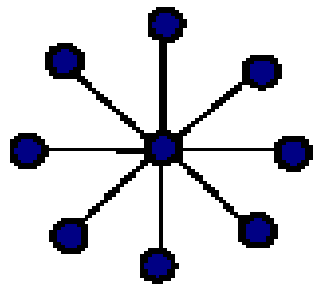
Réseau Mondial



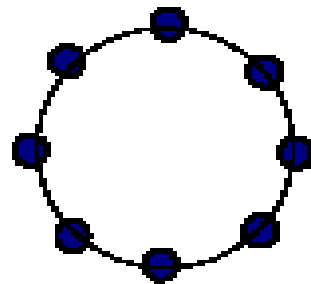
Communiquer - comment ?

- Broadcast - diffusion
- Point à point
- Multicast

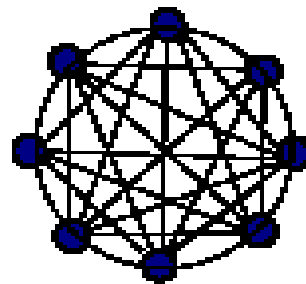
Topologies pour réseaux point-à-point



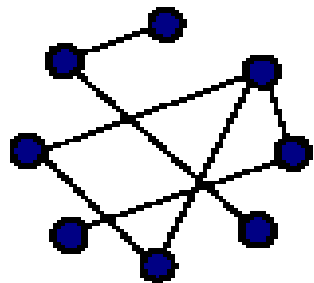
étoile



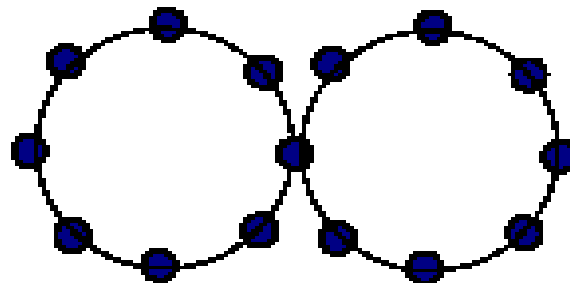
anneau



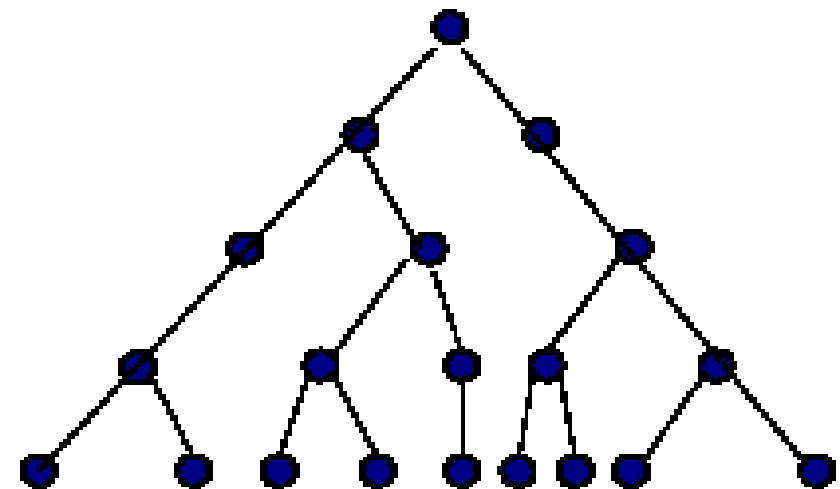
maillage régulier



maillage irrégulier



double anneau



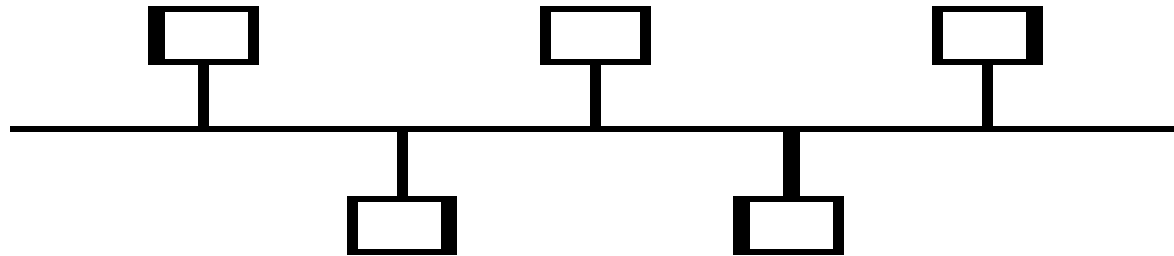
arbre

L'importance des Topologies

- La structure des topologies influence largement la performance d'un réseau:
 - Chemins les plus courts?
 - Distribution de la charge dans le réseau par différent cheminement?
 - Nombre de bonds ou arcs à traverser?
 - Complexité pour calculer les cheminement les plus favorables?

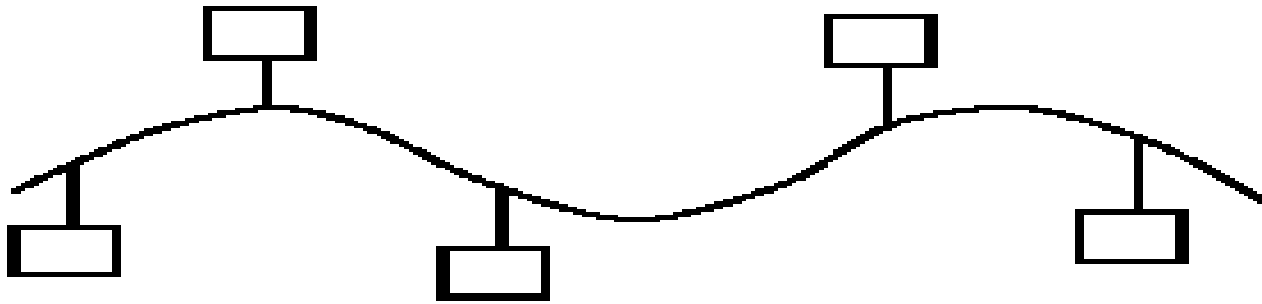
Types de Réseaux

- Réseaux locaux (LANs)
 - Réseau à diffusion
 - Normalement privé
 - Réseau d'ordinateurs
 - Distance typique: 10 km
 - Performance typique: 10 MBit/sec - 1 GBit/sec (ou plus)



Types de Réseaux

- MAN
 - Réseau à diffusion dans les villes
 - Haute performance
 - Distance typique: 100 km
 - Performance typique: 100 MBit/sec - 1 GBit/sec (ou plus)



Types de Réseaux

- WAN
 - Réseau à grande distance, géographiquement étendu
 - Utilisation des réseaux de télécommunication publics
 - Liaison point-à-point
 - Distance typique: 1000 km

Architecture de Protocole

- La tâche de communication divisée en plusieurs modules
- Exemple
 - Le transfert de fichiers peut utiliser 3 modules
 - Application de transfert de fichier
 - Module du service de transport/communications
 - Module d'accès au réseau

Couche d'accès réseau

- Échange de données entre la machine et le réseau
- La machine source fournit l'adresse de destination
- Peut exiger des niveaux de service
- Dépend du type de réseau utilisé (LAN, commutation de paquets, ...)

Couche de Transport

- Échange de données fiable
- Indépendante du réseau utilisé
- Indépendante de l'application

Couche Application

- Supporte des applications différentes
 - E-mail, transfert de fichier, etc.

Couches: Pourquoi?

Complexité

- Structure explicite permet l'identification et les relations entre les différents éléments du système complexe
 - Modèle à couche
- Séparation entre l'implémentation et la spécification
 - Cacher l'information
 - Changer l'implémentation d'un service fourni par une couche est transparent pour le reste du système
 - E.g., changer la procédure de communication n'affectera pas le reste du système

Couches OSI

- Application
- Présentation
- Session
- Transport
- Réseau
- Liaison
- Physique

TCP/IP Protocol Architecture

- Développé par DARPA (US Defense Advanced Research Project Agency) pour son réseau à commutation de paquets (ARPANET)
- Utilisé par Internet
- 5 couches
 - Application
 - Transport
 - Couche Internet (Internet layer)
 - Liaison (Network Access layer)
 - Physique

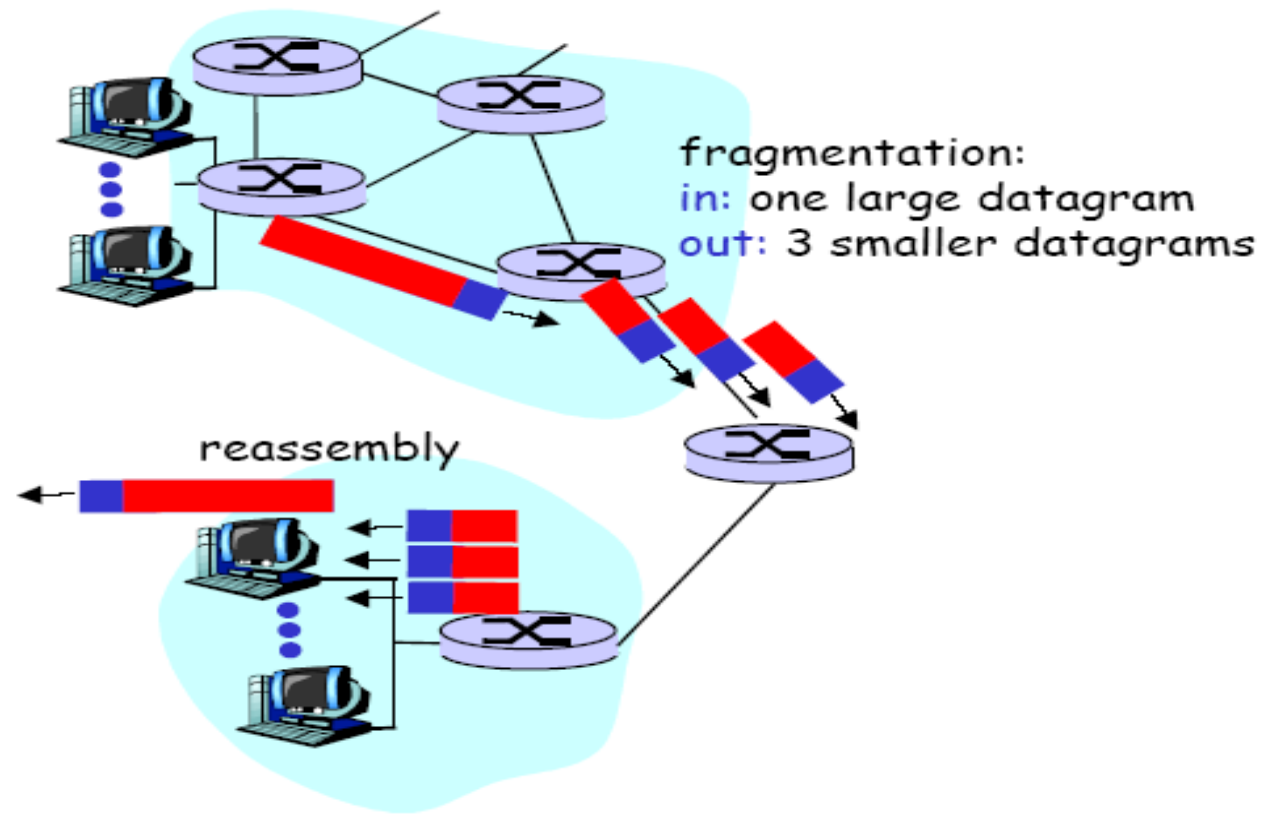
Exigences d'adressage

- Deux niveaux d'adressage requis
- Chaque machine a besoin d'une adresse réseau unique
- Chaque instance d'application s'exécutant sur une machine a besoin d'une adresse unique dans la machine
 - Point d'accès au service (service access point): SAP

Segmentation (Fragmentation)

- Les blocs de données sont d'une taille déterminée
- Les messages de la couche application peuvent grands
- Les paquets réseaux peuvent être petits
- Segmentation consiste à couper des blocs de données qui sont grands en plus petits blocs
 - Fragmentation dans le modèle TCP/IP
 - Les trames Ethernet ne peuvent pas dépasser 1526 octets

Segmentation (Fragmentation): Example



Contrôle de flux

- Fait par le receveur
- Limiter la quantité ou le taux de données
- Techniques
 - Plusieurs techniques utilisées (sans entrer dans les détails)

Contrôle d'erreur

- Protéger contre les pertes et les dommages
- Détection d'erreur
 - La source insert des bits de détection d'erreur
 - La destination vérifie ces bits
 - Si OK, confirme
 - Sinon, détruire le paquet
- Retransmission
- Fait à différents niveaux

Adressage

- Une adresse unique pour chaque machine et chaque routeur (par interface)
- Adresse niveau réseau
 - Adresse IP (TCP/IP)
 - NSAP: Network Service Access Point (OSI)
- Processus dans la machine
 - Numéro de port (TCP/IP)
 - Service Access Point (OSI)

Adressage

- Une adresse IP est un entier écrit sur quatre octets, elle peut donc prendre des valeurs entre 0 et $2^{32} - 1$. Pour plus de commodité, on note les adresses en donnant les valeurs de chaque octet séparés par des points ; par exemple, 1100000010101000000000100001101 s'écrit : 11000000 10101000 00000001 00001101. devient 192.168.1.13.
- Une adresse IP est constituée de deux parties : l'adresse du réseau et l'adresse de la machine, elle permet donc de distinguer une machine sur un réseau. Deux machines se trouvant sur un même réseau possèdent la même adresse réseau mais pas la même adresse machine.

(source https://fr.wikibooks.org/wiki/R%C3%A9seaux_TCP/IP/Adressage_IP_v4)

Adressage

classe	premiers bits	premier octet	masque
A	0	0-127	255.0.0.0
B	10	128-191	255.255.0.0
C	110	192-223	255.255.255.0
D	1110	224-239	
E	1111	240-255	

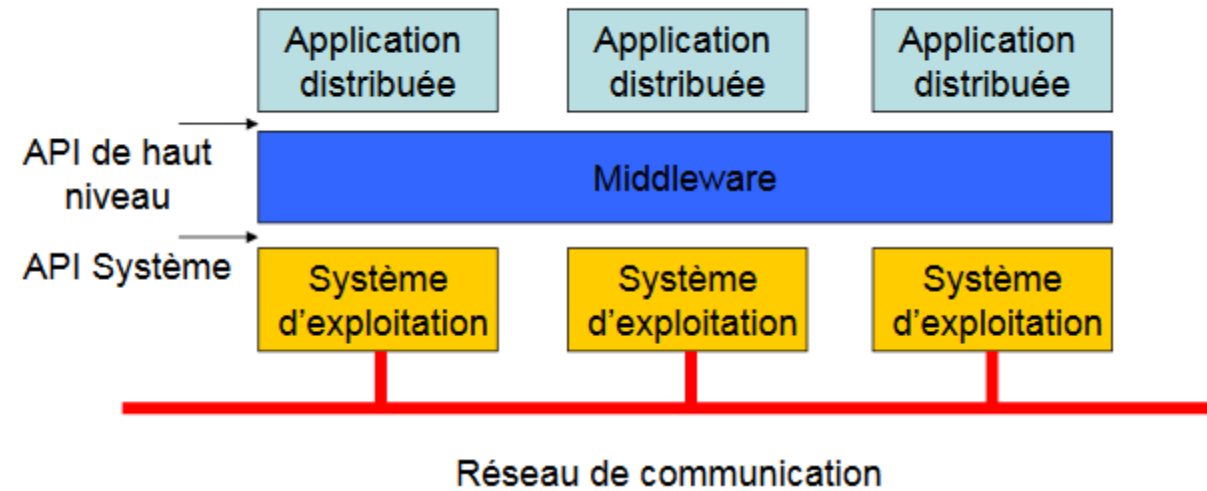
Adresses réseaux et adresse de diffusion

IP (classe)	masque	adresse réseau	adresse de diffusion
10.10.10.10 (A)	255.0.0.0	10.0.0.0	10.255.255.255
192.168.150.35 (C)	255.255.255.0	192.168.150.0	192.168.150.255

Adresses déconseillées et réseaux privés

classe	réseau privé
A	10.0.0.0
A	127.0.0.0
B	de 172.16.0.0 à 172.31.0.0
C	de 192.168.0.0 à 192.168.255.0

Middleware



Middleware

- Plusieurs applications distribuées:
 - Domaine médicale
 - L'environnement
- Chaque Application doit prendre en considération:
 - Distribution inhérente des ressources
 - Disponibilité des ressources et des informations
 - Contrainte de la Qualité de service demandé par les applications.
 - Limitation des ressources (bande passante)
 - Coopération entre les autres applications (doivent partager les ressources disponible)
- Nécessité d'un besoin pour contrôler pour les fonctionnalités du réseau et des capteurs individuels pour le mieux servir les exigences des applications.
- Solution Middleware

Qu'est ce qu'un middleware

- Logiciel situé entre l'application et le système d'exploitation
- C'est une "colle" entre l'application système d'exploitation, qui est utilisé pour envoyer des messages aux composantes à distance, collaborant avec l'application distribuée.
- Capable de produire des abstractions des fonctionnalités des niveaux plus bas (comme la cohérence de réseau) et en fournissant une interface de coordination évoluée au programmeur d'applications.
- Rend la conception et l'implémentation des applications facile

Applications et services

Middleware

Network operating System

Hardware

Middleware

- Les composants logiciels du middleware assurent la communication entre les applications quels que soient les ordinateurs impliqués et quelles que soient les caractéristiques matérielles et logicielles des réseaux informatiques, des protocoles réseau, des systèmes d'exploitation impliqués.

Caractéristiques d'un middleware

- Extensible (scalable) face aux contraintes des ressources.
- Générique au différentes applications avec une interface commune
- Adaptatif pour configurer sa structure
- Réflexif: doit prendre conscience de ce qui se passe dans l'environnement d'exécution pour bien fonctionner en offrant ainsi à l'application la possibilité d'être réactive.