

# Intergiciels et applications communicantes

Philippe Quéinnec

Informatique et Réseaux 2e année  
ENSEEIH

5 février 2018

Inspiré de cours de G. Padiou, Ph. Mauran et S. Krakowiak. Certains dessins en sont issus.



# Première partie

## Introduction



# Plan

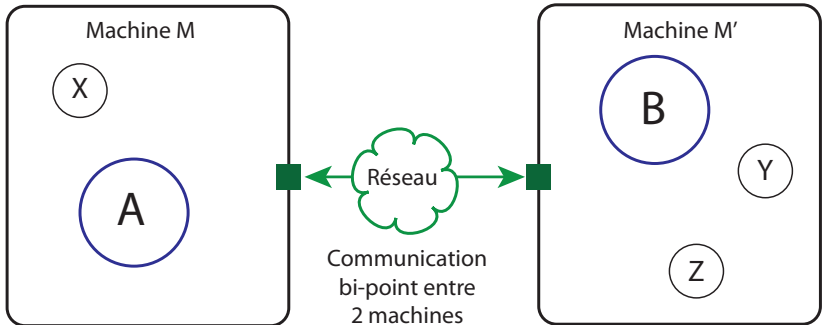
- 1 La communication à distance
- 2 Les protocoles
- 3 Les intergiciels (middleware)



# La communication à distance

## Communication au niveau réseau

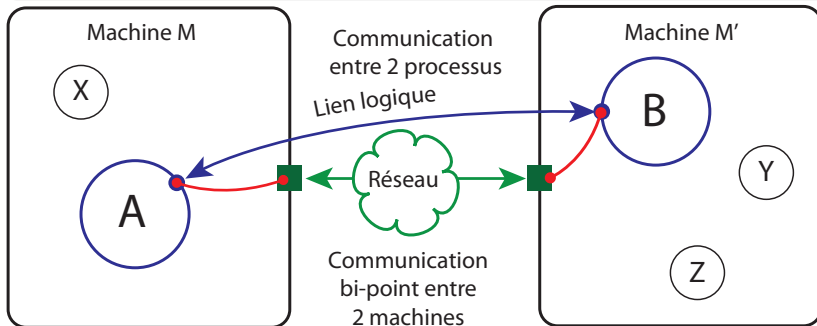
Les ingénieurs réseaux font communiquer des machines



# La communication à distance

## Communication au niveau intergiciel

Les ingénieurs informaticiens font communiquer des applications



## La communication à distance : objectifs

Faire communiquer des **processus** par échange d'information

Pas de mémoire partagée, communication « à distance »

- Exploiter les réseaux de communication pour faire communiquer des ordinateurs
- Avantage : le partage de ressources et l'échange d'informations
- Comment ? définition de protocoles de communication entre processus distants
- Difficulté : hétérogénéité du matériel, des systèmes d'exploitation, des programmes applicatifs écrits dans différents langages : C, C++, Java...



# Plan du cours

- 1 La communication à distance entre applications
- 2 Communication par socket
- 3 Appel de procédure et appel de méthode à distance
- 4 Intergiciel asynchrone : communication par messages
- 5 Tolérance aux fautes
- 6 Mémoire virtuelle répartie, systèmes de fichiers répartis



## Distinction service / serveur

### Service

Un service est une description / interface / spécification.

### Serveur

Un serveur est une réalisation / implantation / concrétisation d'un service

### Besoins

- Trouver les services existants
- Pour un service donné, trouver les serveurs qui le réalisent
- Dialoguer avec un serveur donné (= protocole)





## Service/serveur – exemple

### Je veux manger

- Quels sont les services offerts dans un village  $\Rightarrow$  boulangerie, bistrot. . .
- Où trouver une boulangerie?  $\Rightarrow$  place Dupuy
- Comment acheter une baguette  $\Rightarrow$  protocole d'interaction

### Service de nommage internet (DNS)

- Service de conversion de noms symboliques en adresses IP numériques
- À l'N7, serveurs sur les machines 147.127.80.123, 147.127.176.22, 147.127.16.11



# Plan

- 1 La communication à distance
- 2 Les protocoles
- 3 Les intergiciels (middleware)



# Les protocoles de communication

Grande variété de mise en œuvre

## Beaucoup de paramètres

- Point à point ou diffusion
- Synchronisation émetteur-récepteur :
  - Envoi asynchrone : l'émetteur ne se bloque pas
  - Envoi synchrone : l'émetteur attend un acquittement
  - Gestion de tampons en émission et/ou en réception
- Délais de transmission (non) bornés
- Fiabilité : pertes possibles, duplications, erreurs...
- Désignation des entités distantes (processus)



# Les protocoles de communication (suite)

## Mais aussi selon le niveau d'abstraction

- Simple échange d'un message
- Transaction de messages, par exemple :  
question, réponse [, acquittement ]
- Flots de messages sur une liaison pré-établie
- Appel procédural à distance, appel de méthode à distance
- Publication et abonnement
- etc



# Plan

- 1 La communication à distance
- 2 Les protocoles
- 3 Les intergiciels (middleware)



# Les intergiciels et leur rôle

## Définition

- Couche logicielle entre système d'exploitation et applications
- Service implantant un modèle d'interaction entre processus
- API **normalisée** associée au service de communication implanté

## Quelques points communs

- Besoin d'une interface avec la couche « transport »
- Besoin de certains services de base : service de nommage
- Nécessité de traiter l'hétérogénéité entre les processus



# Les sources d'hétérogénéité

## Deux processus communicants peuvent

- exécuter un programme écrit dans des langages différents
- être exécutés par des systèmes d'exploitation différents
- s'exécuter sur des architectures de machine différentes

## Deux approches de solution

- Définition d'un langage commun (IDL/Interface Description Language) et instanciation dans différents contextes d'exécution → normes RPC, CORBA
- Définition d'un environnement de développement et d'exécution portable et exécution dans n'importe quel contexte → environnement Java