المدرسة الدولية الخاصة للتقنيات
POLYTECH INTL

Informatique Répartie
- Remote Procedure Call -
(RPC)

Ichrak MEHREZ
(m_ichrak@hotmail.fr)

2018-2019

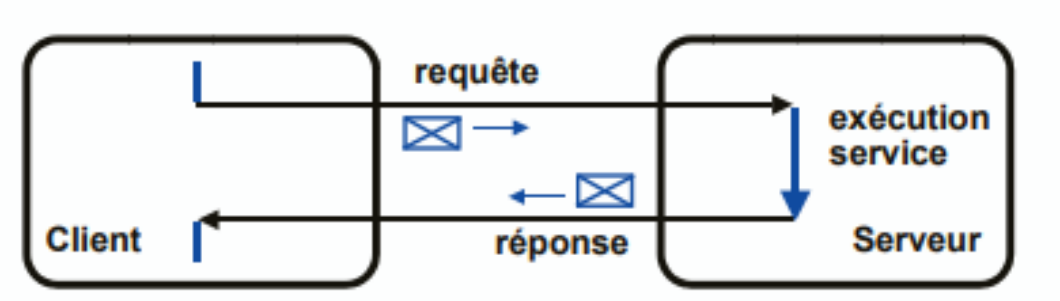
Rappel: Application répartie

Il s'git d'une application découpée en ***plusieurs*** unités

- Chaque unité peut être placée sur une machine différente
- Chaque unité peut s'exécuter sur un système différent
- chaque unité peut être programmée dans un langage différent

Modèle Client/Serveur

- Appel synchrone requête-réponse



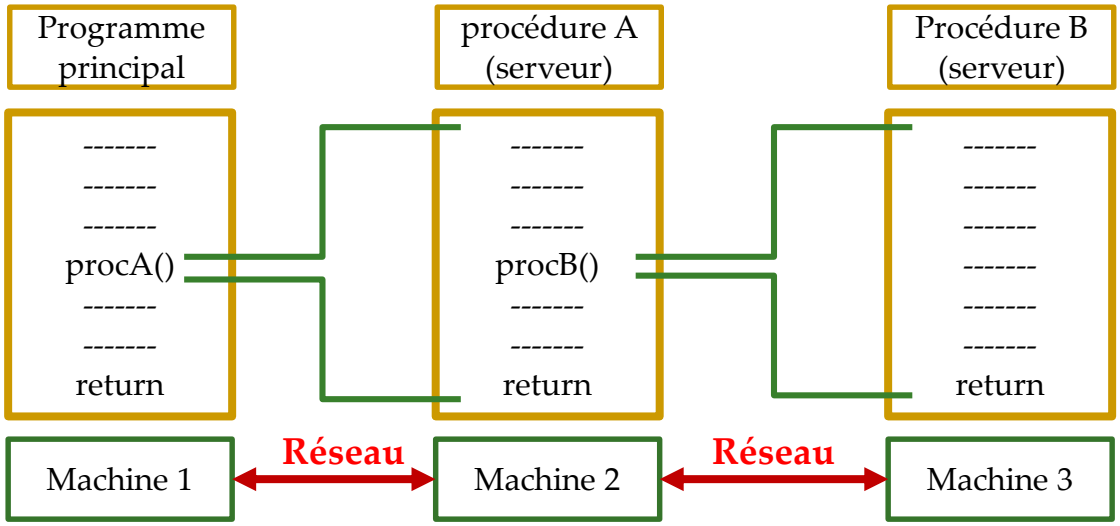
- Mise en oeuvre
  - Haut niveau : intégration dans un langage de programmation : ***RPC*** (construit sur sockets)
  - Bas niveau : utilisation directe du transport : sockets (construit sur TCP ou UDP)

Remote Procedure Call (RPC)

- Introduit par Birrell et Nelson (1984)
- Outil de haut niveau pour la réalisation du schéma client-serveur
- Garder la sémantique de l'appel de procedure locale ou Local Procedure Call (LPC)
- Fonctionnement Synchrone
- Communication transparente entre le client et le serveur

RPC: Principe

1. Un client a une description (ou *une interface*) de procédures disponibles chez un serveur distant
2. Le client invoque une procédure à distance
3. Le serveur exécute localement cette procédure;
4. Le serveur renvoie au client le résultat de la procédure



RPC: fonctionnement

- Utilisation d'un processus ***s*** (ou serveur) qui exécute la procédure ***P*** sur le site ***S***.

- Le processus client ***p*** reste bloqué pendant l'exécution de ***P*** est il est réactivé au retour de ***P***.

RPC: fonctionnement

- Deux procédures appelées ***talons***: le ***talon client***, sur le site appelant, fournit au processus client une interface identique à celle de la procédure appelée, ***le talon serveur*** , sur le site appelé, réalise pour le processus serveur une interface identique à celle d'un appel local.

- Chacun des talons, lié au programme du processus client ou serveur sur le site correspondant, fonctionne aussi comme un représentant de l'autre processus (client ou serveur sur ce site)

- En outre, les talons doivent détecter et traiter les erreurs (délais de garde)

RPC: fonctionnement

Talon client :

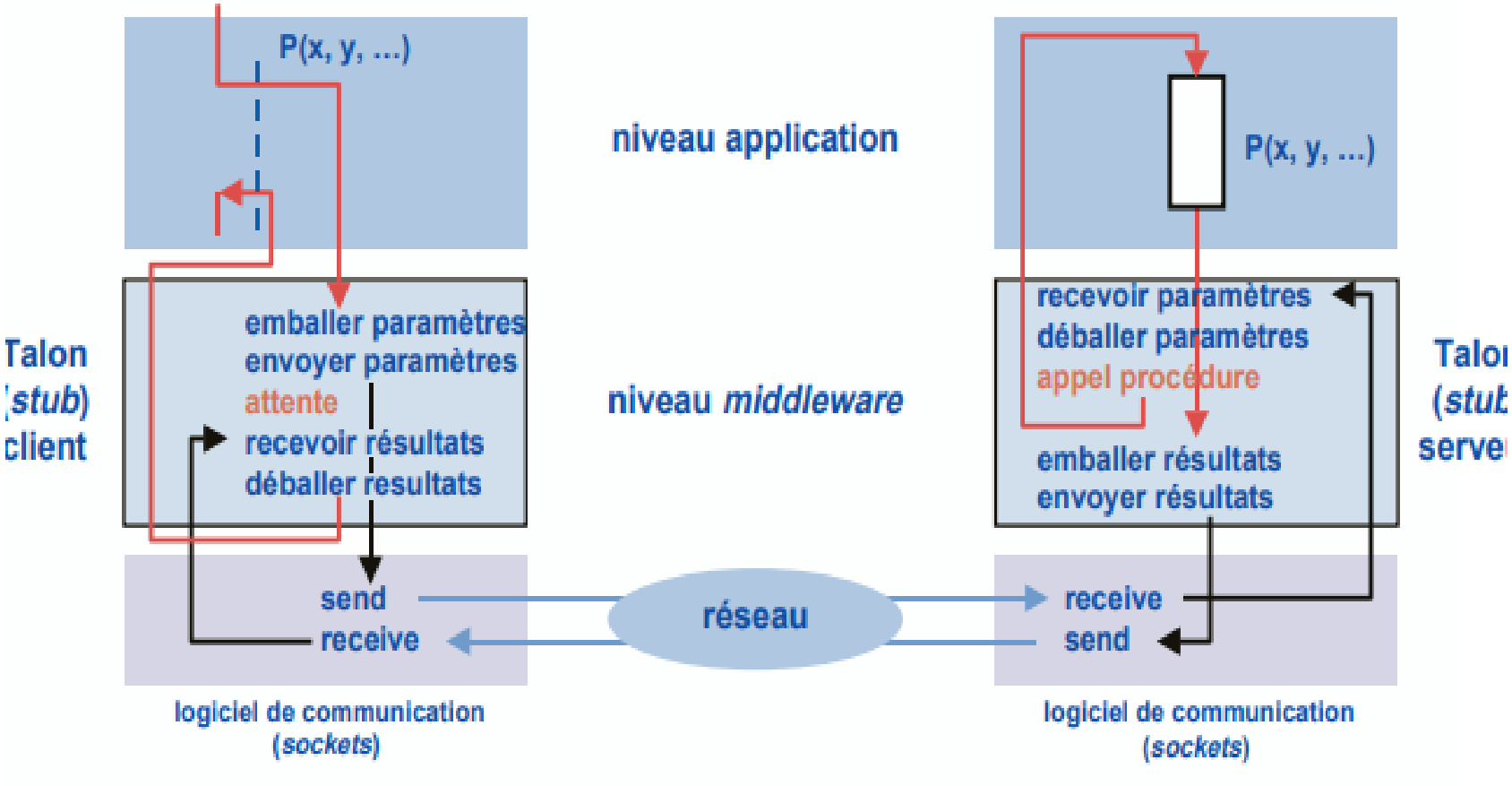
- Représente le serveur sur le site client
- Reçoit l'appel local
- ***Emballe*** les paramètres
- Crée un identificateur unique pour l'appel
- Exécute l'appel distant
- Met le processus client en attente
- Reçoit et ***déballe*** les résultats
- Exécute le retour vers l'appelant (comme retour local de procédure)

RPC: fonctionnement

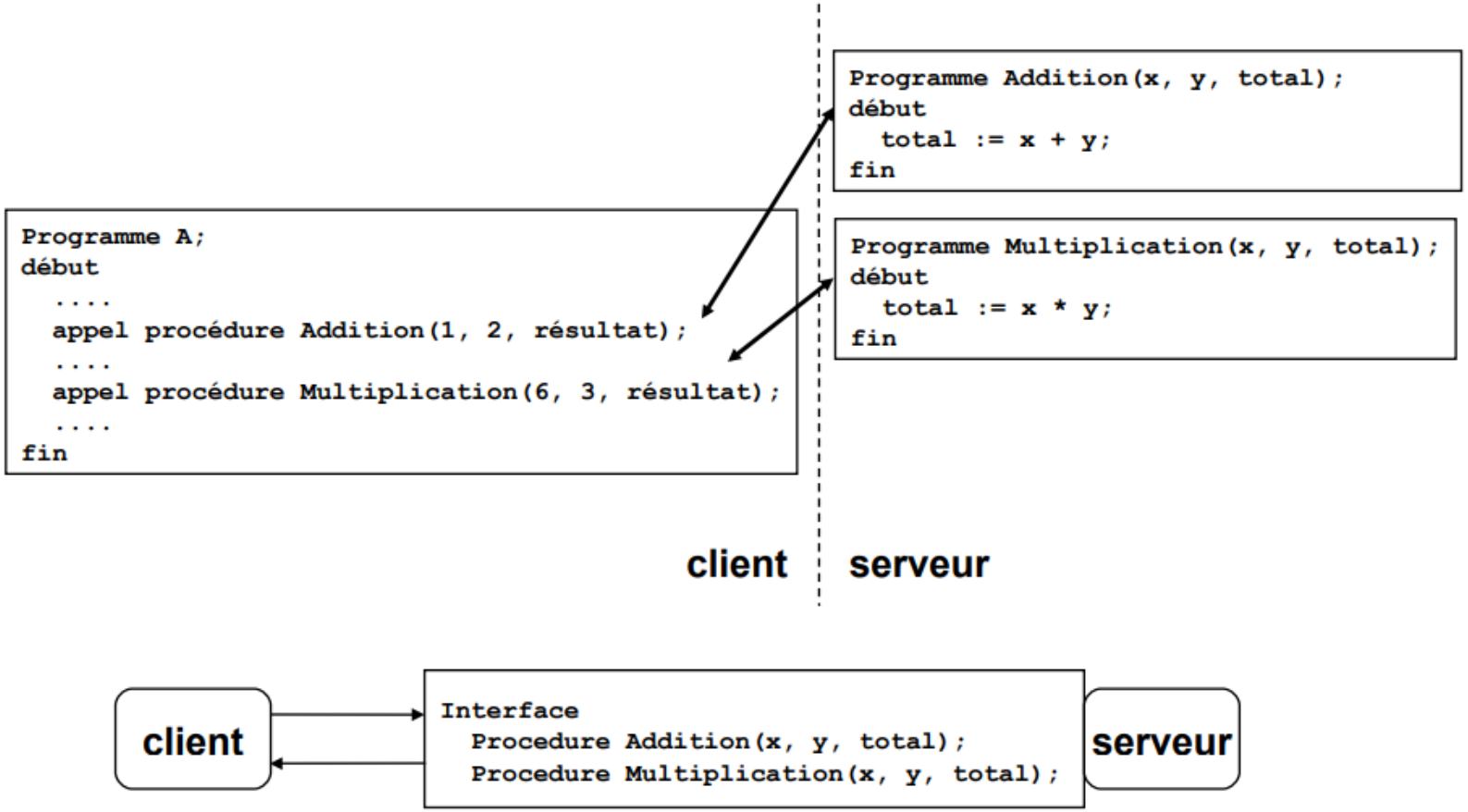
Talon Serveur :

- Représente le client sur le site serveur
- Reçoit l'appel sous forme de message et ***déballe*** les paramètres
- Crée ou sélectionne un processus (ou thread) et lui fait exécuter la procédure
- ***Emballe*** les résultats et les transmet à l'appelant

RPC: fonctionnement



Exemple: Calcul de fonctions mathématiques



Problèmes

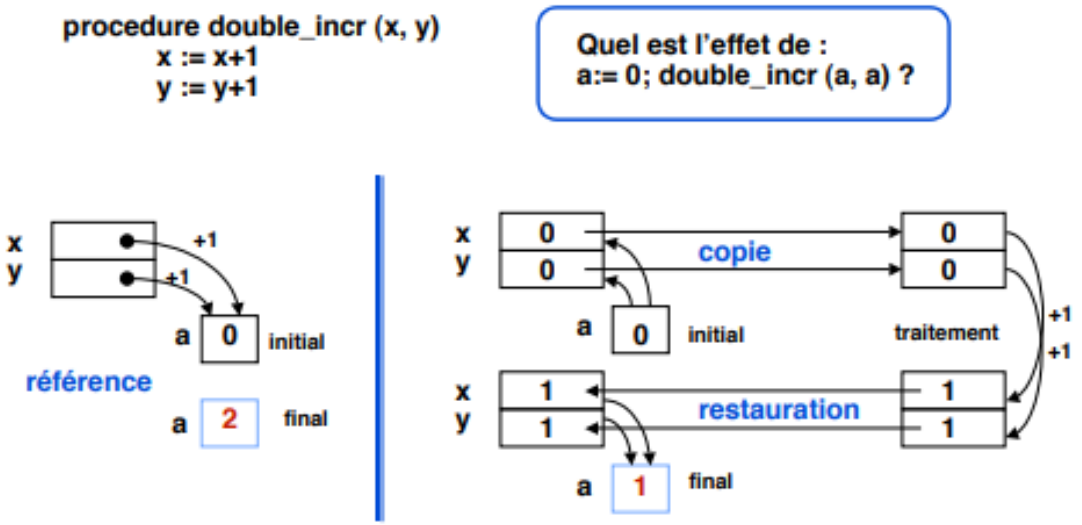
- Passage de paramètres
- Désignation et Liaison
  - Localisation (adresse serveur)
  - Procédure au sein d'un serveur

Problèmes: passage de paramètres

- Pas de passage de paramètres par adresse:
  - Impossible de passer des pointeurs (ou références). En effet, les espaces d'adressage du client et du serveur sont différents donc aucun sens de passer une adresse
- La procédure distante n'a pas un accès aux variables globales du client, aux périphériques d'E/S (affichage d'un message d'erreur!)
- Un appel de procédure obéit à un fonctionnement synchrone: une instruction suivant un appel de procédure ne peut pas s'exécuter tant que la procédure appelée n'est pas terminée

Problèmes: passage de paramètres

- Passage de paramètres par référence impossible
  - Passage par valeur
  - Passage par copie/restauration

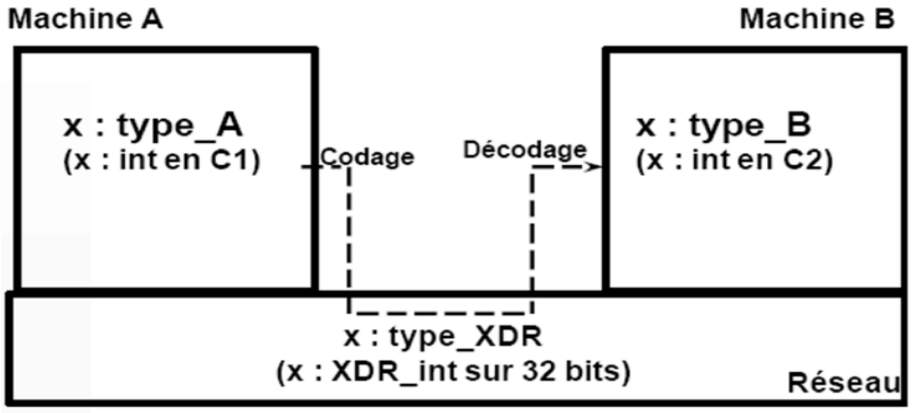


Problèmes: passage de paramètres

- Hétérogénéité des machines !!

  ➔ Deux solutions possibles

  - Codage/décodage de chaque type de donnée de toute architecture à toute autre architecture
  - Format universel intermédiaire (XDR, CDR, etc.)



Désignation et liaison

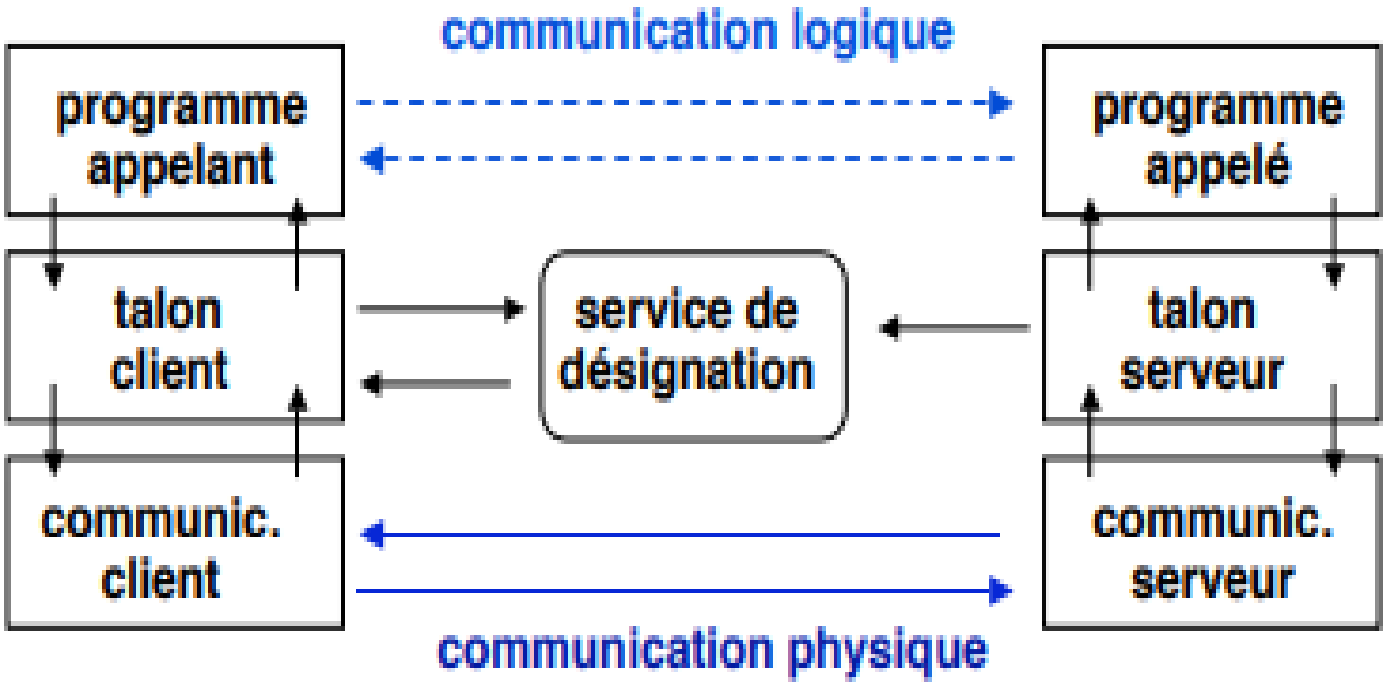
Objets à désigner

- Procédure appelée, site serveur
- Propriétés souhaitées : désignation indépendante de la localisation

Moment de liaison

- Liaison précoce (statique) ou tardive (dynamique)
- Statique : localisation du serveur connue à la compilation
- Dynamique : localisation non connue à la compilation
  - Désignation symbolique des services (non liée à un site d'exécution
  - Possibilité d 'implémentation ou de sélection retardée

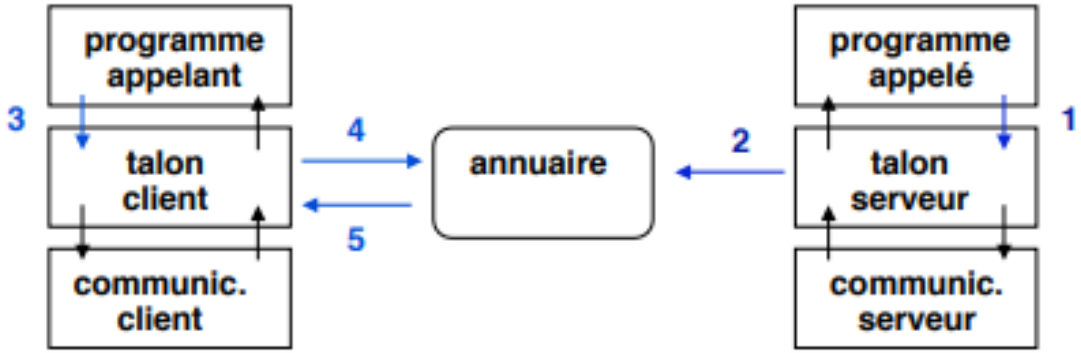
Désignation et liaison



Désignation et liaison

Désignation et liaison utilisant un annuaire

- 1, 2 : le serveur s'enregistre auprès de l'annuaire avec *<nom, adr. serveur, n° port>*
- 3, 4, 5 : le client consulte l'annuaire pour trouver *<adr. serveur, n° port>* à partir de *<nom>*
- L'appel peut alors avoir lieu



Désignation et liaison

Désignation et liaison utilisant le portmapper

- Si le serveur est connu (cas fréquent) : on peut utiliser un service de nommage local au serveur, le ***portmapper***
- Un service enregistre le n° de port de son veilleur auprès du portmapper et le veilleur se met en attente sur ce port
- Le portmapper a un n° de port fixé par convention (111)