

Développement d'applications réparties

Emna Ben Abdallah

emnabenabdallah@ymail.com

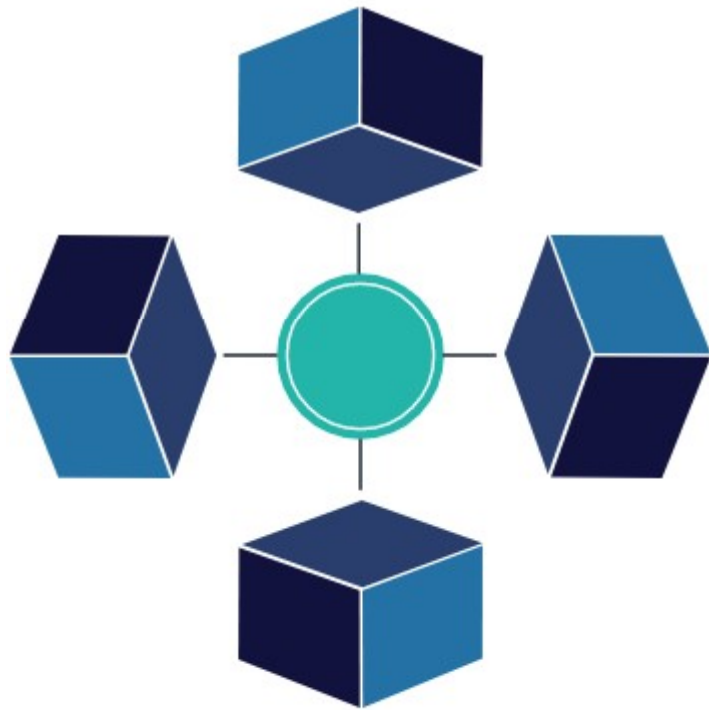


Chapitre0 – présentation du cours

- Objectifs du cours
 - Concevoir et développer des applications qui peuvent s'exécuter sur plusieurs machines

Chapitre0 – présentation du cours

- Plan du cours
 - Chapitre 1 : Introduction aux systèmes répartis
 - Chapitre 2 : Le langage java
 - Chapitre 3 : Les sockets JAVA
 - Chapitre 4 : Les objets répartis – Java RMI



Chapitre 1 : Introduction aux systèmes répartis



Plan

- Application et système
- Systèmes répartis
- Applications réparties
- Phases construction d'une AR
- Intergiciel (Middleware)



Application et système

- **Système** : fonctionne sur un ordinateur unique [Neumann]. Un système est un ensemble de matériels et logiciels opérant en vue d'un objectif précis, généralement l'exécution cohérente d'une ou plusieurs application(s) et sa mise à la disposition d'un ou plusieurs utilisateur(s) finaux.
- **Application** : ensemble d'entités logicielles, qui permet à un ordinateur ou à un système informatique d'assurer une tâche ou une fonction particulière.
- ✓ Les énormes capacités des micro-ordinateurs ont permis la réalisation d'applications monolithiques où toutes les tâches sont traitées sur la même machine.

Application et système

■ **Application**

- Capacités de traitement importantes, mais besoins toujours plus importants (en puissance de calcul, en espace de données, en partage de l'information,...).

Comment tirer parti au mieux de l'infrastructure matérielle et réseau aujourd'hui à notre disposition ?



Réparties ?



- Répartie \neq Centralisée
- La **répartition** est un état de fait pour un nombre important d'applications
 - Développement des réseaux (Internet, réseaux sans fil)
 - Intégration d'applications existantes initialement séparées
 - Pénétration de l'informatique dans des nouveaux domaines d'application
 - Intégration d'objets du monde réel (informatique omniprésente (ubiquitous computing))
 - Intégration entre informatique et télécommunications
- Le parallélisme est la réponse aux besoins croissants des applications
 - Puissance de traitement
 - Gestion de grandes masses de données
 - Intégration et mise en commun de ressources



Systèmes répartis

- **Système réparti (distribué)** : Ensemble de machines autonomes connectées par un réseau, et équipées d'un logiciel dédié à la coordination des activités du système ainsi qu'au partage de ses **ressources**.
- Accès commun aux ressources (calcul, stockage, impression,...)
- Partage d'information (travail coopératif, algorithmes répartis,...)

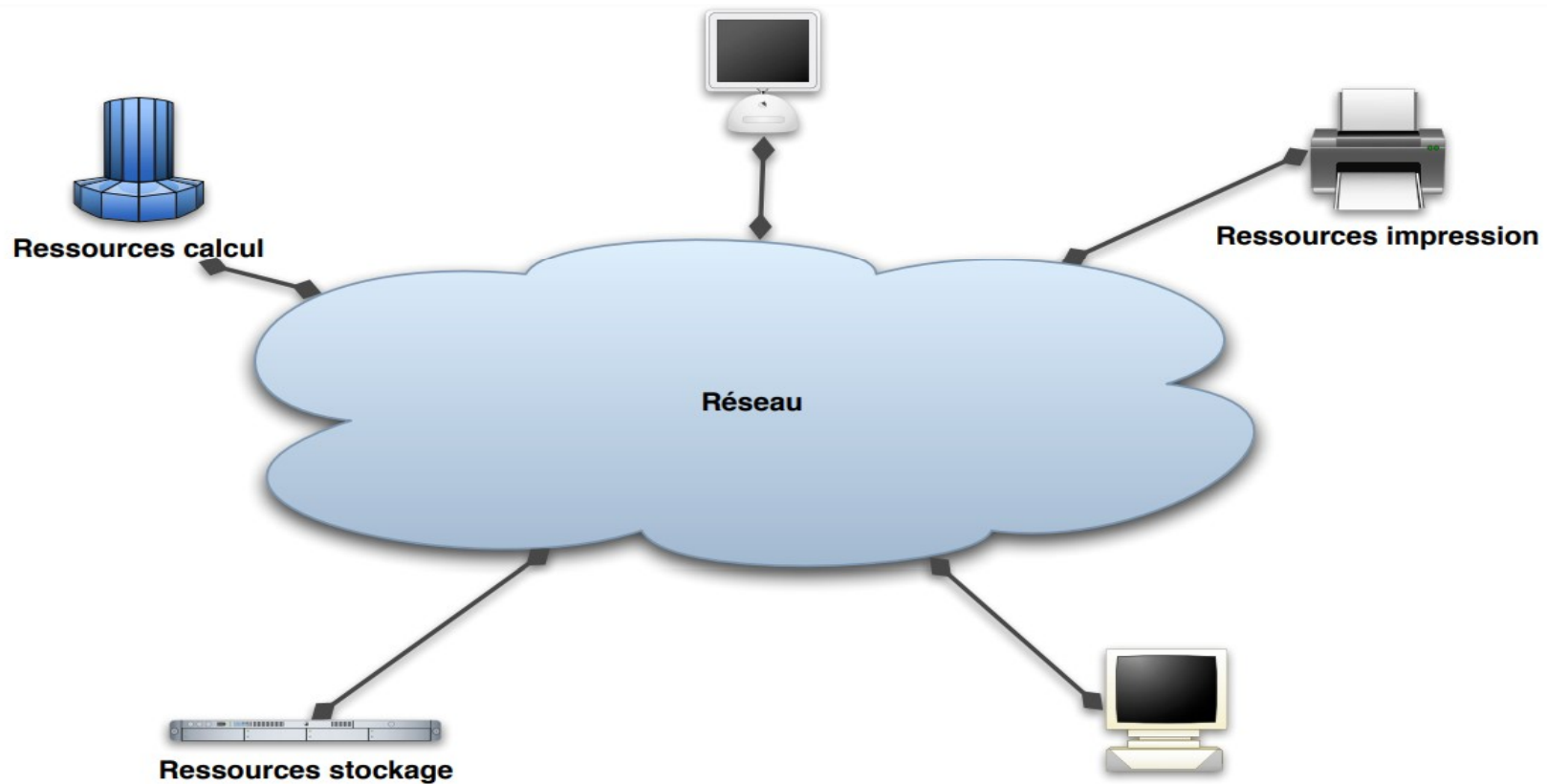
Réparti = distribué ?

Distribué → fabricants d'ordinateurs

Réparti → Génie logiciel

Systèmes répartis

- Architecture de haut niveau



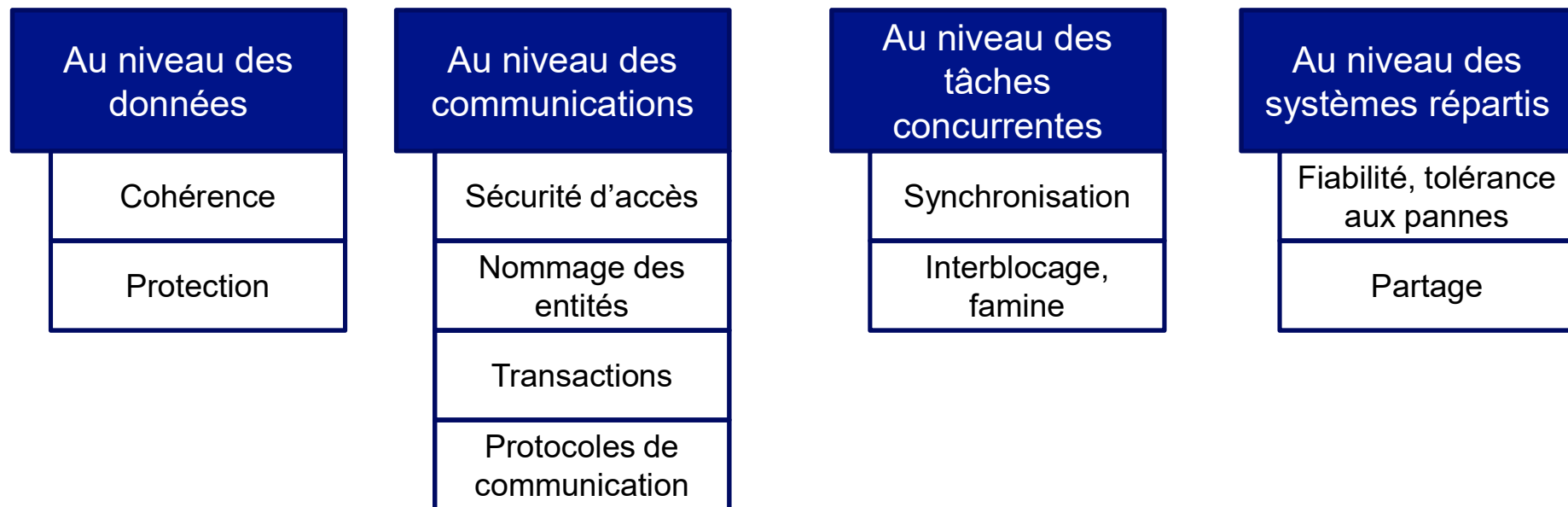


Systèmes répartis

- Catégories des systèmes réparties
 - **Multi-processeurs à mémoire partagée**
 - Plusieurs processeurs **homogènes**
 - Un seul système d'exploitation
 - ✓ Communication rapide et fiable, faible latence
 - ✗ Coût élevé, peu versatile
 - **Multi-ordinateurs à mémoire répartie.**
 - Plusieurs processeurs homogènes ou **hétérogènes**
 - Plusieurs systèmes d'exploitation
 - ✓ Coût minimal, très versatile
 - ✗ Communication lente et non-fiable, latence élevée et variable

Systèmes répartis

■ Problématiques



Systèmes répartis

■ Avantages et inconvénients

Avantages

- ✓ Fiabilité et disponibilité
- ✓ Performances
- ✓ Gain en termes de maintenance, d'évolution

Inconvénients

- ✗ Difficulté de réalisation
- ✗ Pas de vision globale
- ✗ Administration lourde

Systèmes répartis

■ Exemples d'utilisation

- Base de données réparties (banques, réservation, entreprises)
- Systèmes de fichiers répartis (NFS, AFS, ...)
- Calculs parallèles gourmands en temps de calcul (traitement d'image, conception CAO, chimie, biologie, physique, ...)
- Navigation aérienne
- Travail coopératif: courrier électronique, messageries
- Web, eCommerce
- Enseignement à distance
- Systèmes embarqués (civiles et militaires)

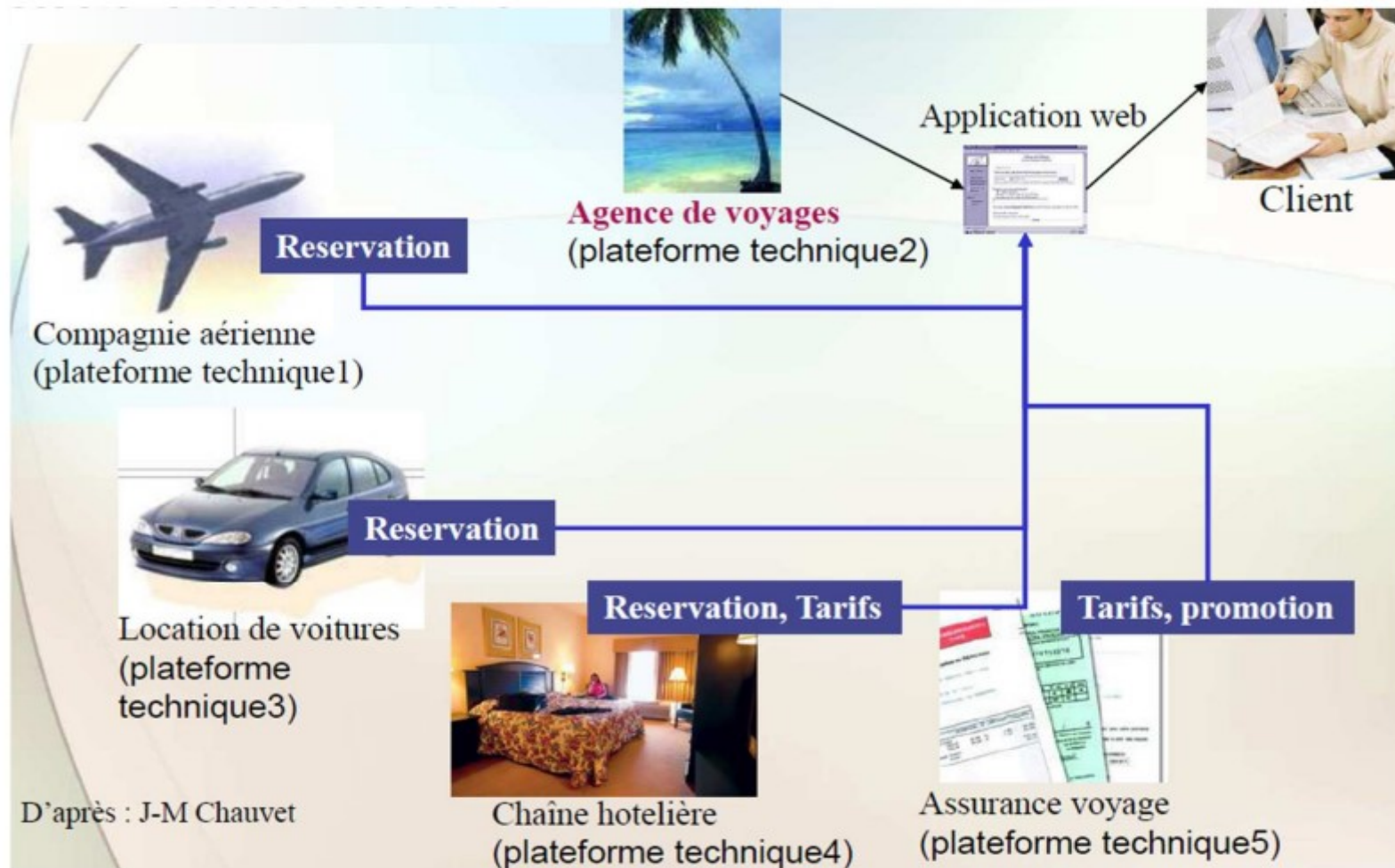
Application répartie

- **Application répartie** : Est une application découpées en plusieurs unités (composants) connectées via un réseau où :
 - Chaque unité ou ensemble d'unités peut être placée sur une machine différente.
 - Chaque unité peut s'exécuter sur un système différent.
 - Chaque unité peut être programmé dans un langage différent
- Application répartie = Traitements coopérants sur des données réparties
- Catégories des applications réparties :
 - Les applications réparties « centralisées ».
 - Les applications réparties « distribuées ».

Applications réparties vs systèmes répartis : Les applications réparties sont constituées d'entités logicielles interconnectées et les systèmes répartis servent à partager des ressources.

Application répartie

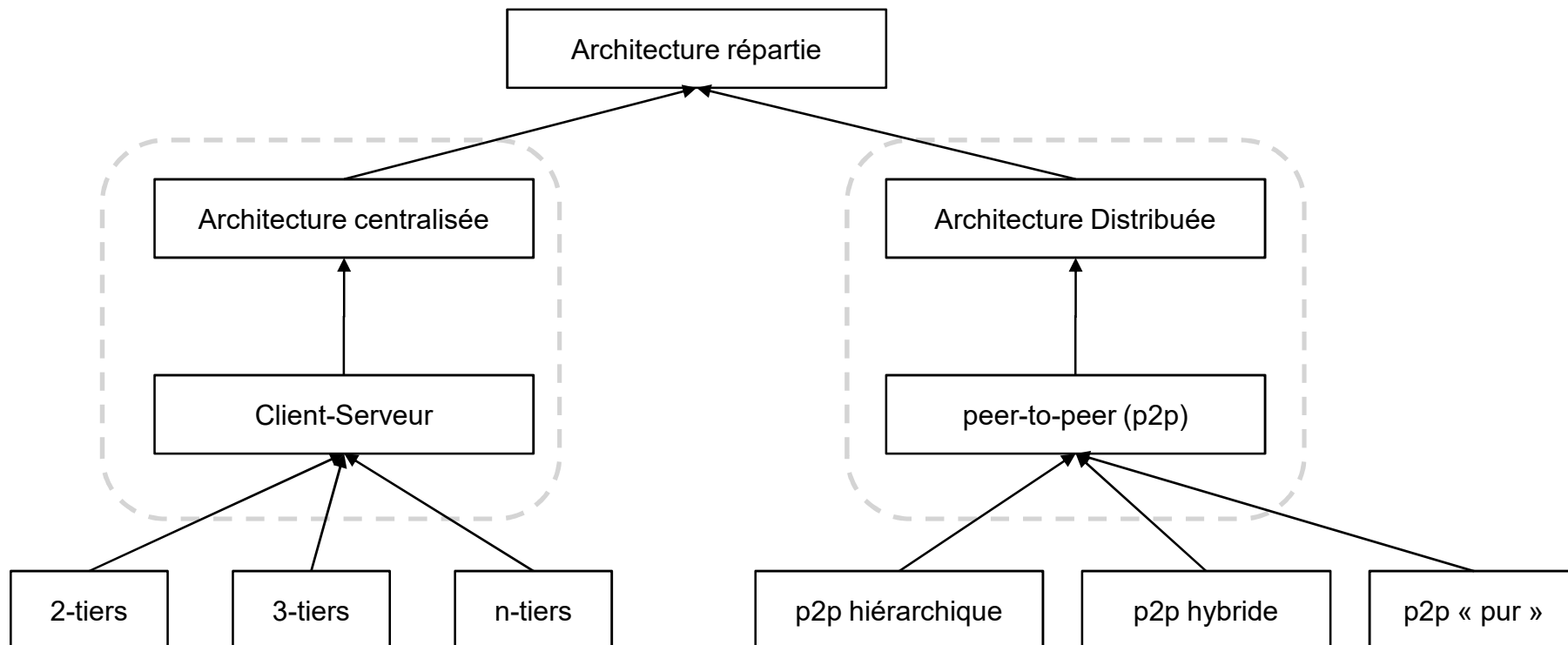
■ Exemple



Applications réparties

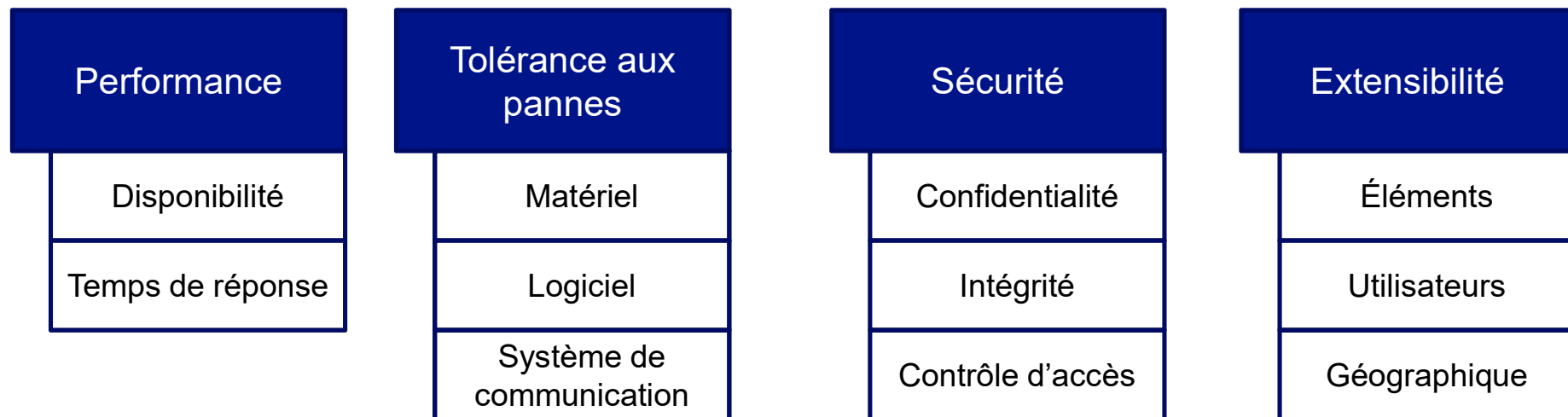


■ Architectures réparties



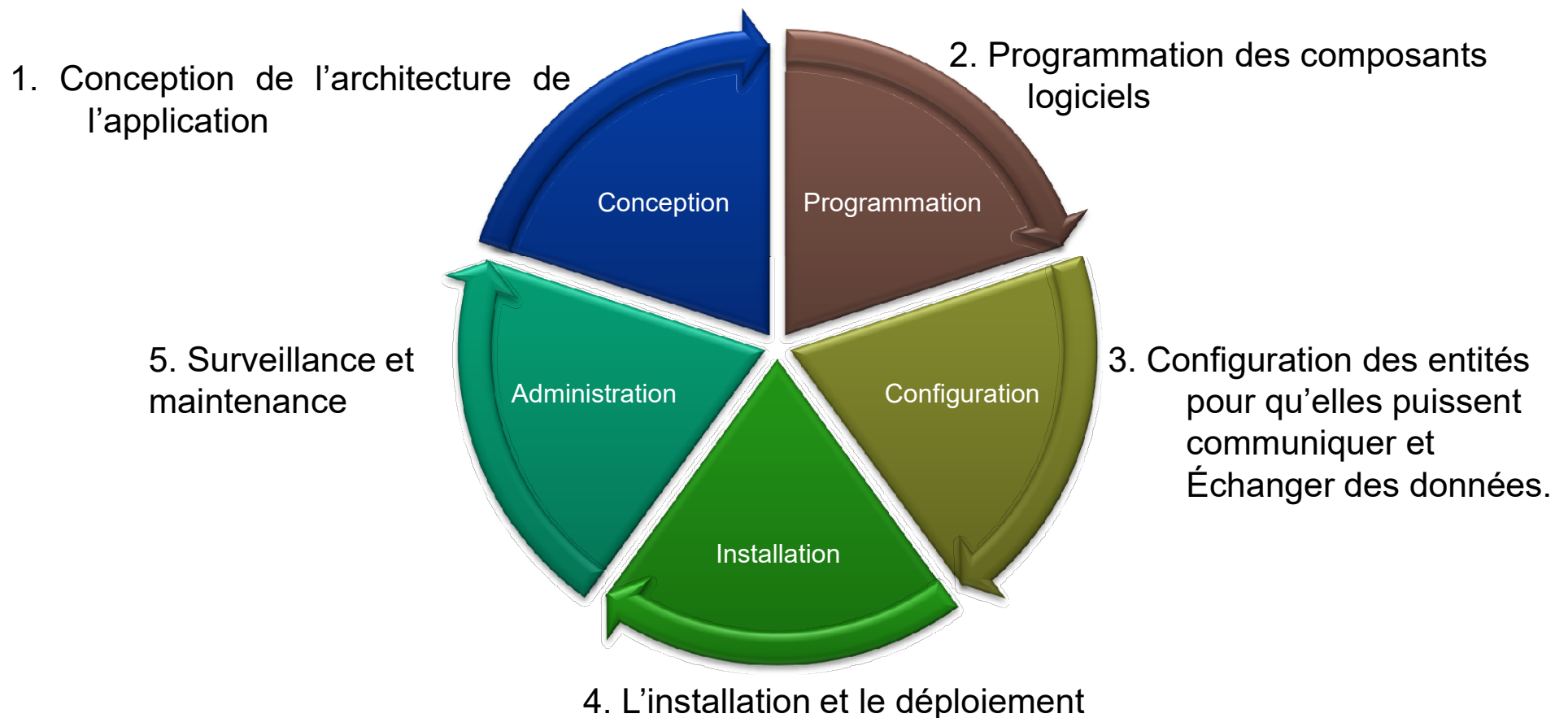
Applications répartis

- Problématiques
 - Les qualités de services (QoS)



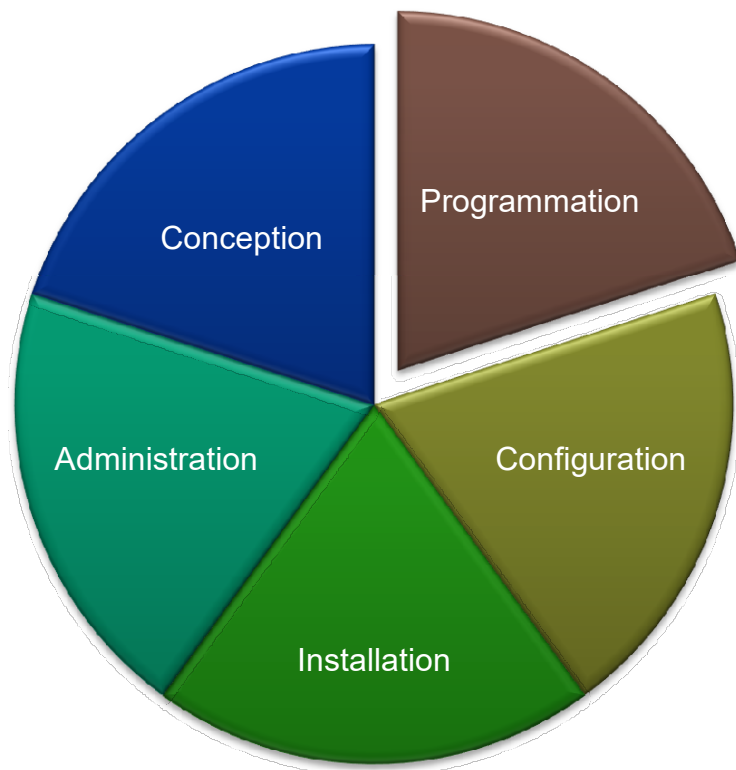
Phase de construction d'une AR

- Les phases de construction d'une AR sont :



Phase de construction d'une AR

- Les phases de construction d'une AR sont :



2. Programmation des composants logiciels

- Utilisation de mécanismes de communication des composants répartis, appelés **Intergiciel ou middleware**, tels que : Socket, RPC, RMI ou CORBA
- Programmation selon le modèle d'exécution adopté



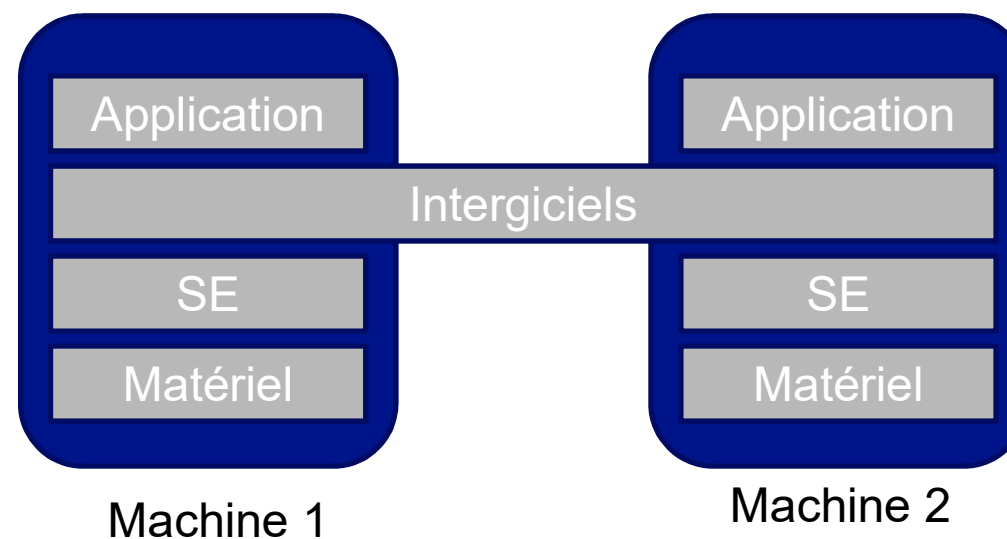
Intergiciel (Middleware)

- Dans un système réparti, l'interface fournie par les systèmes d'exploitation et de communication est encore trop complexe pour être utilisée directement par les applications.
 - Hétérogénéité (matérielle et logicielle)
 - Complexité des mécanismes (bas niveau)
 - Nécessité de gérer la répartition

 - **Solutions :**
 - Introduire une couche intermédiaire (répartie) entre les niveaux bas (système et communications) et le niveau haut (application)
- ➔ **Intergiciel (Middleware)**

Intergiciel (Middleware)

- **Middleware** : est une couche intermédiaire entre le système d'exploitation et l'application afin d'en améliorer sa mise en œuvre.



- **Objectif** : Unifier, pour les applications, l'accès et la manipulation de l'ensemble des services disponibles sur le réseau, afin de rendre l'utilisation de ces derniers presque **transparente**.



Intergiciel (Middleware)

- Types de middleware
 - Type “Remote Procedure Call” (RPC)
 - Exp. Java/RMI
 - Type “Message Oriented Middleware” (MOM)
 - Exp. Java/JMS
 - Type “Object Request Brocker” (ORB)
 - Exp. OMG/CORBA ou Microsoft/COM/DCOM

Intergiciel (Middleware)

■ Avantages et inconvénients

Avantages
✓ Utilisation simple
✓ Multi-plate-forme
✓ Intéropérable
✓ Evolutif

Inconvénients
✗ Performances relativement mauvaises
✗ Fonctionnalités relativement limitées



Fin chapitre