esprit
Se former autrement
HONORIS UNITED UNIVERSITIES

Introduction

Module Services Web

A.U 2022-2023

United Nations Educational, Scientific and Cultural Organization
UNESCO Chair "Project-based learning" ESPRIT School of engineering, Tunisia
EUR-ACE® Délivrée par la Commission des Titres d'Ingénieur
CONFÉRENCE DES GRANDES ÉCOLES
cdio CONCEIVE DESIGN IMPLEMENT OPERATE

Objectifs du module

- Sensibiliser l'apprenant des défis de l'interopérabilité
- Maîtriser les concepts liés de services web et technologies liées
- Construire et déployer des services web et leurs clients
- Sécuriser un service web REST

Plan du module

Charge horaire: 21H

Pré-requis : Java, HTML, protocoles web

Objectifs:

- Services web
- WS REST
- Sécuriser un WS REST
- GraphQL

Chapitre 1

Introduction aux services web

Contexte

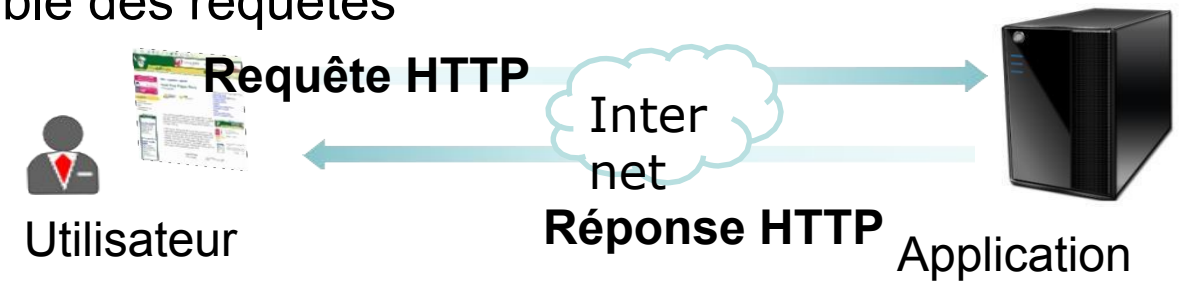
Human-centric web

- Le Web centré utilisateur implique que l'humain est l'acteur principal

pour l'initialisation de l'ensemble des requêtes

B2C

Business ToConsumer

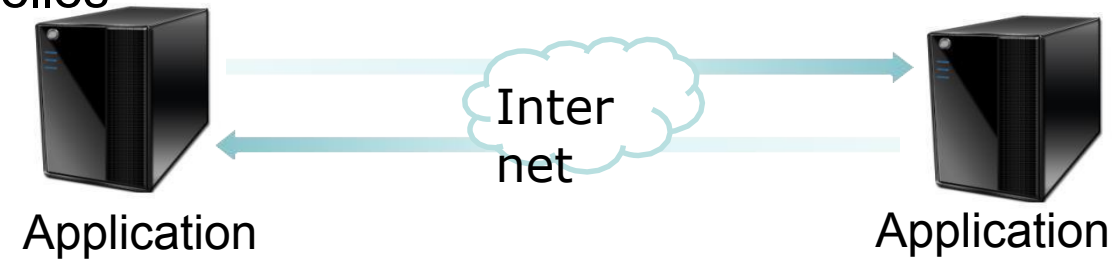


Application-centric web

- Le Web centré application a pour objectif de permettre à des applications de différentes organisations de communiquer entre elles

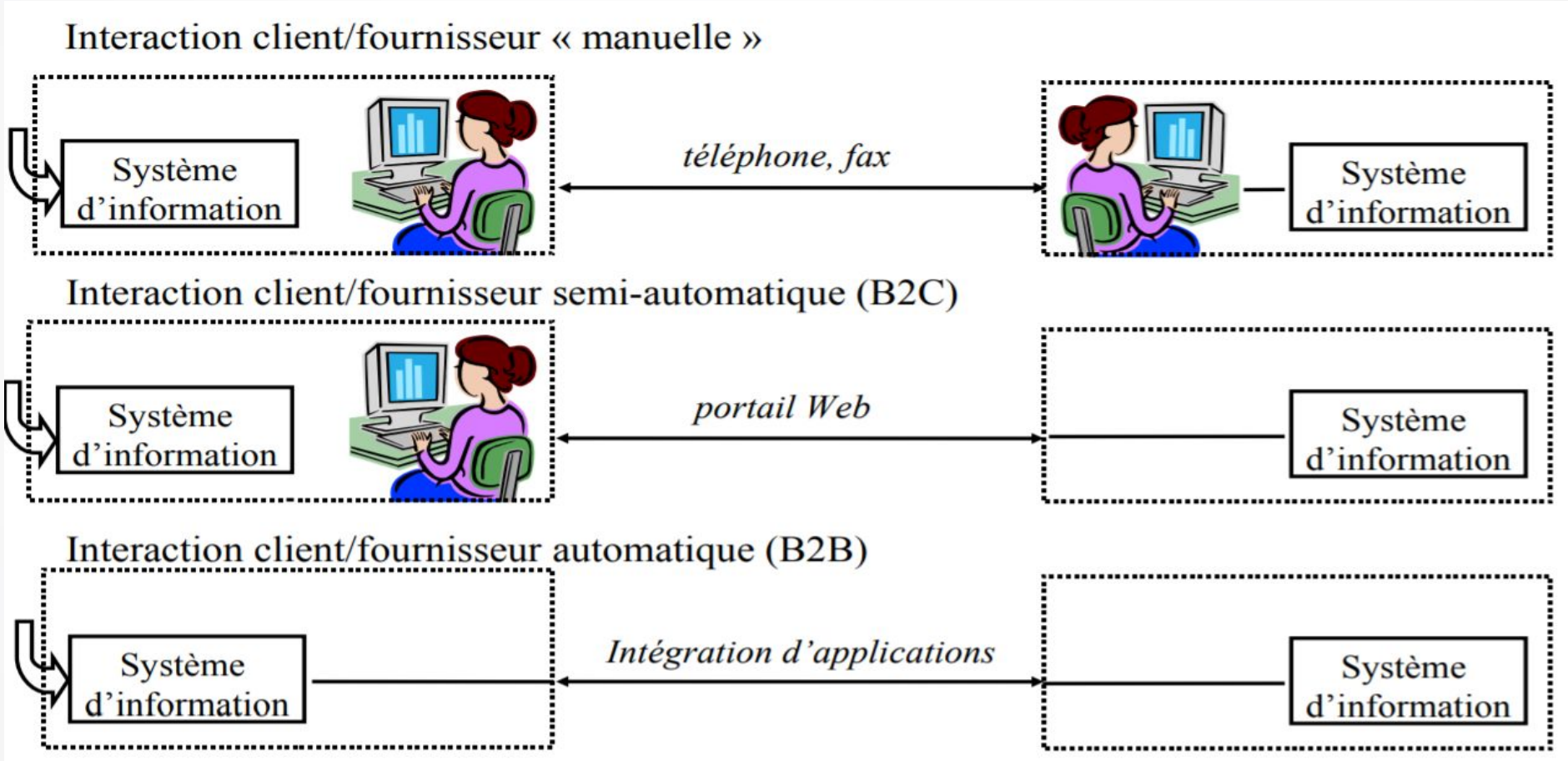
B2B

Business To Business



Contexte

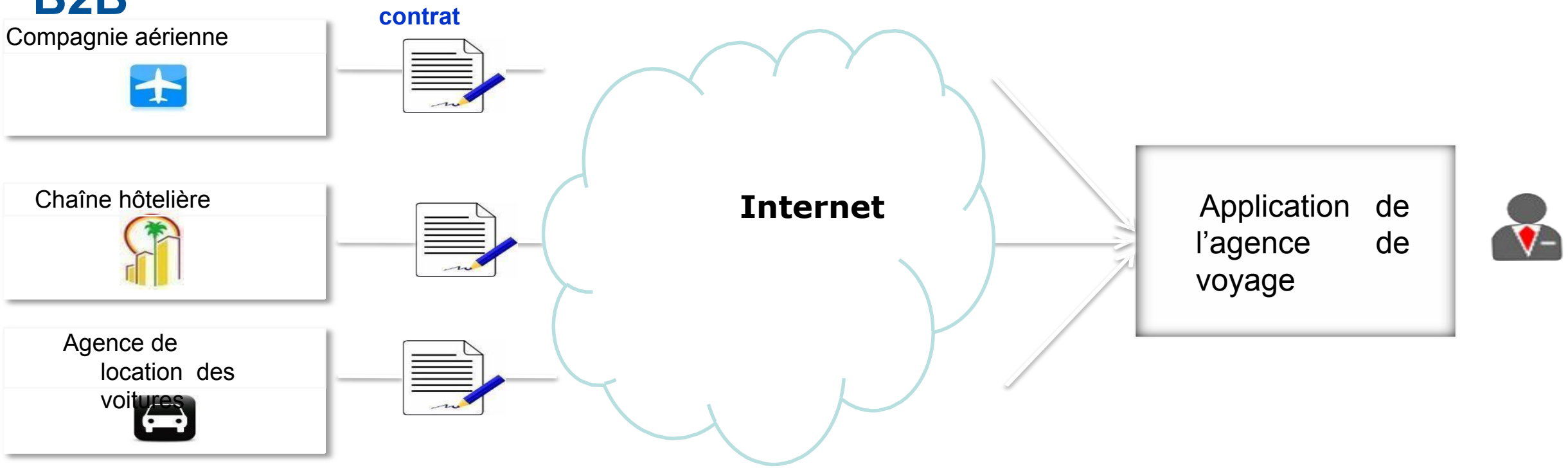
Historique



Source: https://www.ibisc.univ-evry.fr/~tmelliti/cours/CPAR/cours6.pdf

Contexte

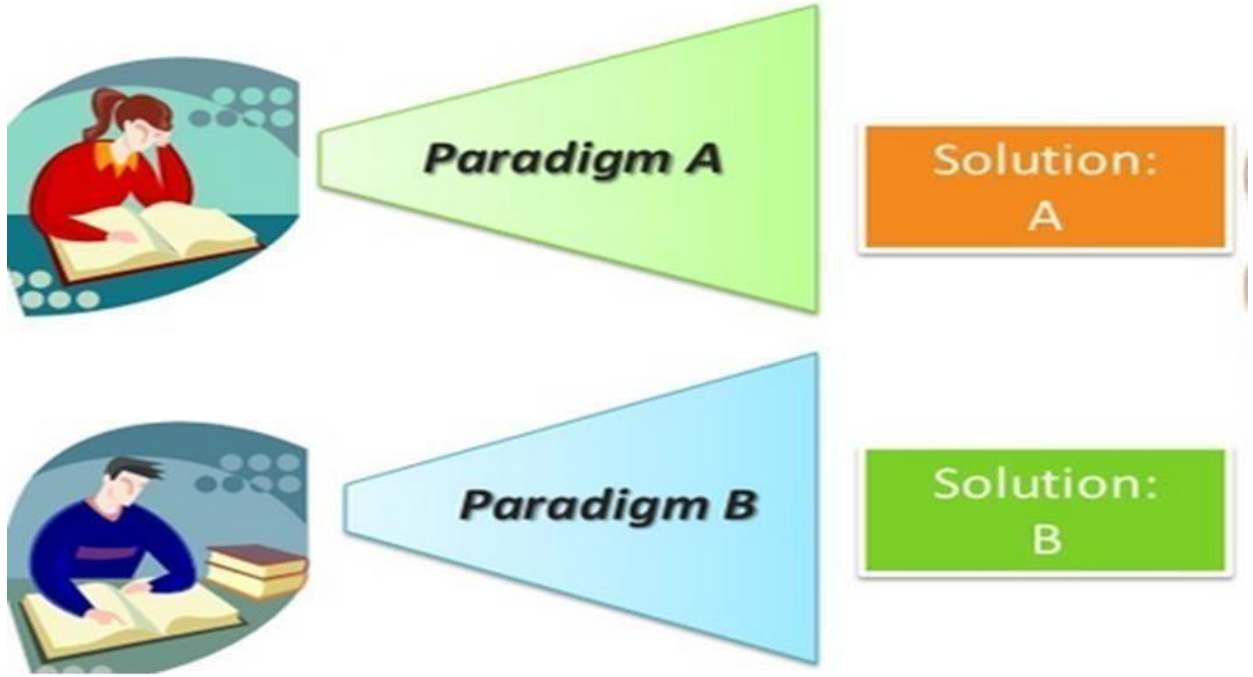
Exemple B2B



Evolution des paradigmes

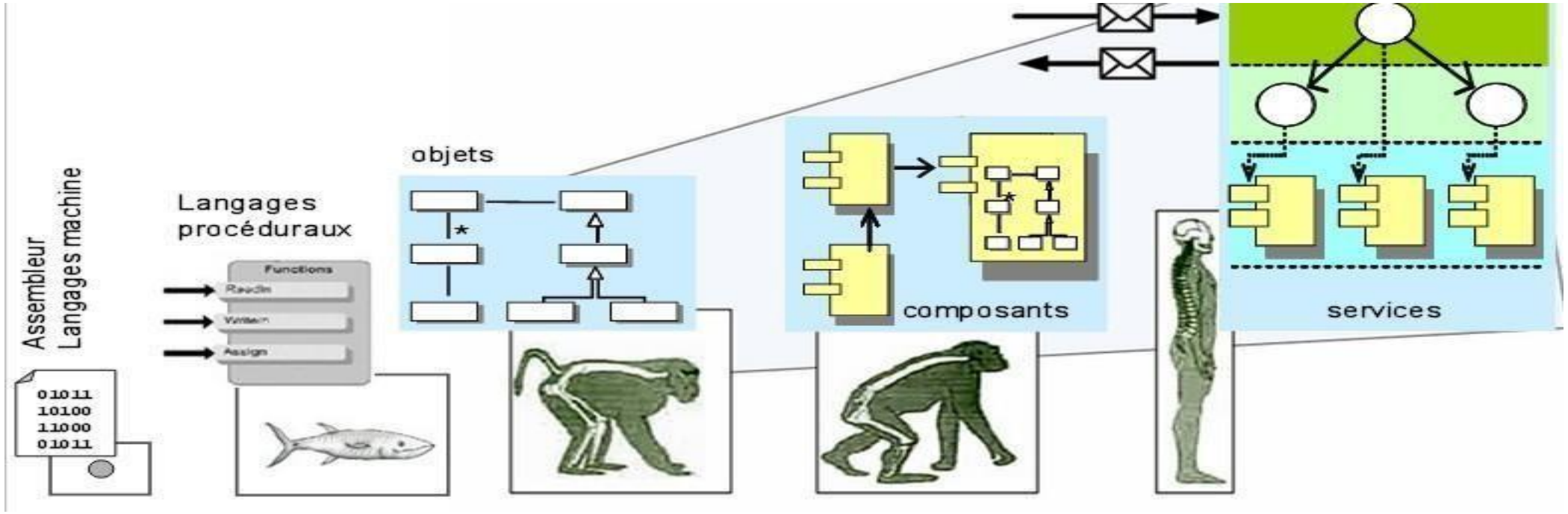
Le terme de **paradigme** est employé pour exprimer la façon dont un système a été conçu et pensé dans ses grandes lignes. [1]

Objectif:
Développer une application de facturation



Evolution des paradigmes

•Les révolutions informatiques coïncident généralement avec un changement de paradigme
•Niveau d'abstraction grandissant avec l'évolution des paradigme



Source : école Hes SO

Paradigme procédural

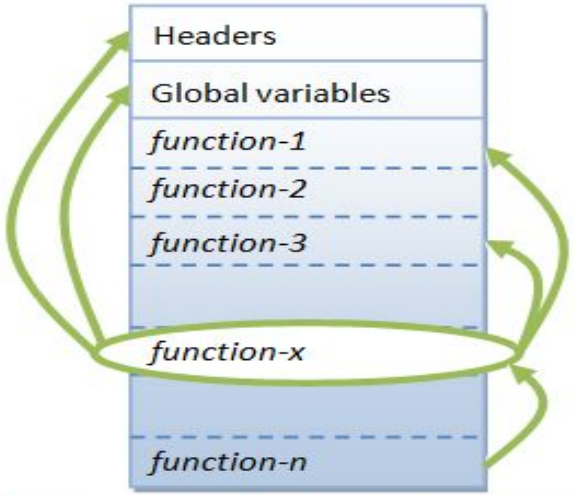
- Le programme est une liste des tâches et des opérations à exécuter.
- Un Système informatique **désordonné**

Limites

Tend à générer du code "Spaghetti " Maintenance complexe

Modularité et abstraction absente

Réutilisation ardue



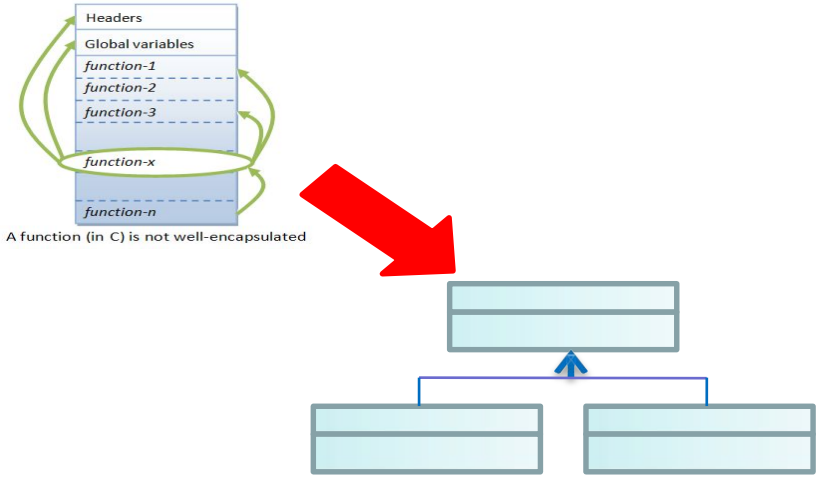
A function (in C) is not well-encapsulated

Paradigme objet

- L'idée est de concevoir les programmes non plus commedes lignes de codes qui s'exécutent séquentiellement, mais comme des **objets qui dialoguent**
- Ses principes incluent l'abstraction encapsulation,dedonnées, polymorphisme et héritage.

Limites

- Réutilisation difficile Couplage rend difficile la fort -> maintenance

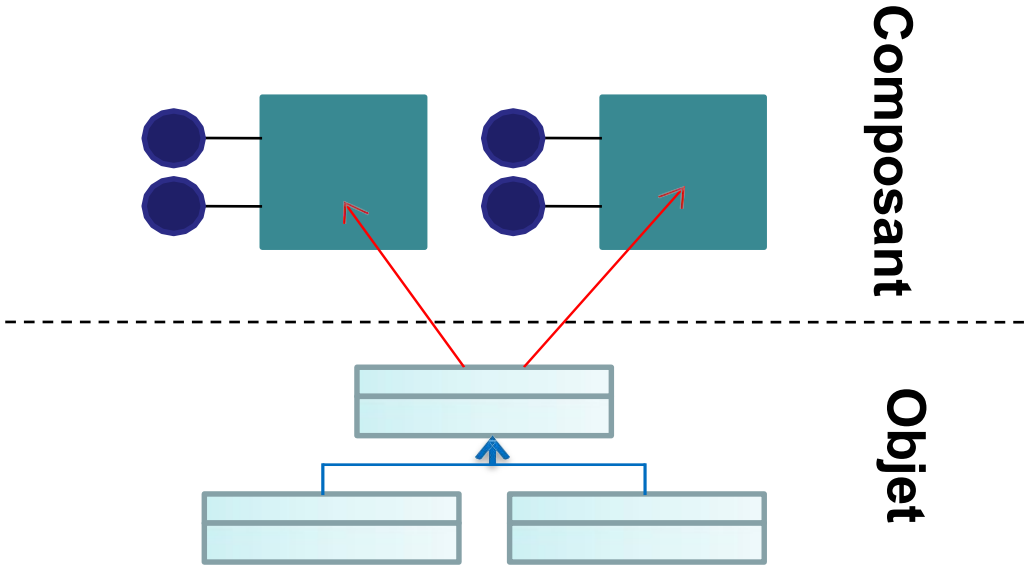


Paradigme composant

- Construire une application composée par un ensemble de briques de base **configurables** et **réutilisables**.
- Il s'agit d'externaliser le code fonctionnel d'une application afin de le rendre **réutilisable** dans d'autres applications

Limites

⊖ Interopérabilité entre composants hétérogènes

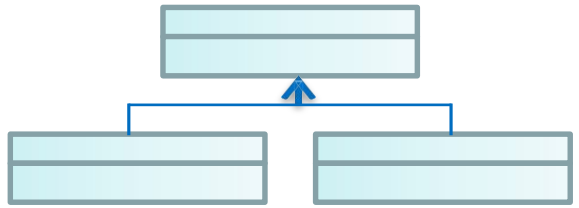


Paradigme service

•Prise en charge de la diversité et de l'hétérogénéité des logiciels, en termes de langages de programmation, de systèmes de technologies de conception (et de réalisation) ou de plates-formes d'exécution

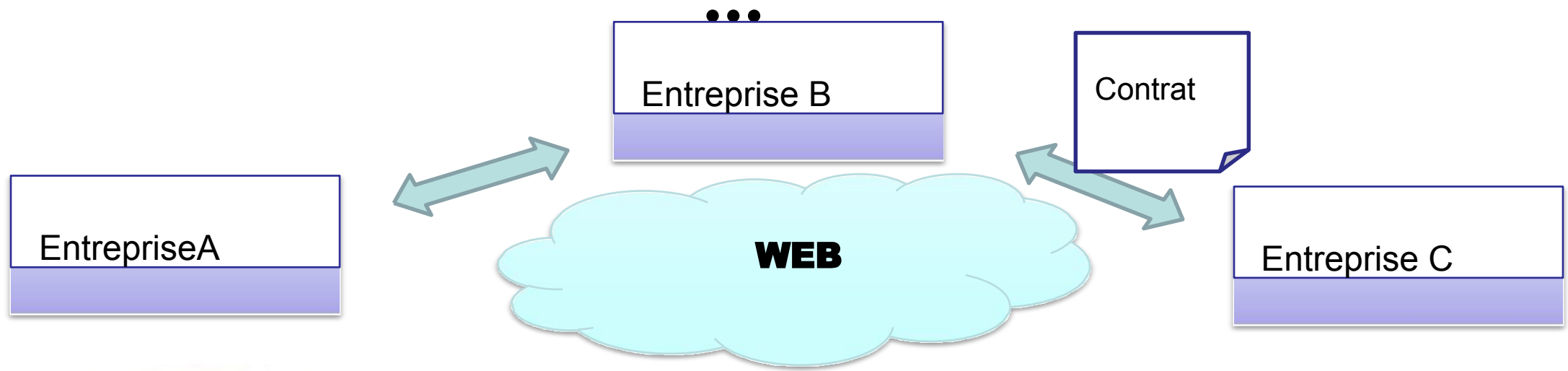
•Le paradigme service permet de:

- réduire le **couplage**
- améliorer la **réutilisation**
- augmenter l'**abstraction**



Service Composant Objet

Besoins

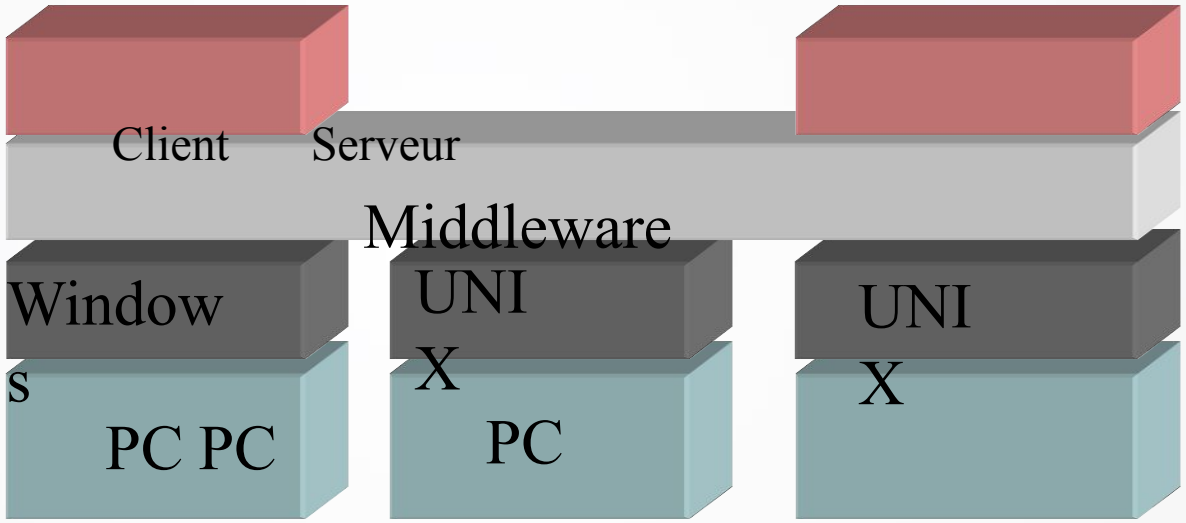


- Langage commun
- Protocole commun
- Contrat
- Middelware

Middleware

Middleware (intergiciel)

- Un logiciel servant d'intermédiaire entre d'autres logiciels; ou
- Un intermédiaire de communication entre des applications complexes et **distribuées** [3]



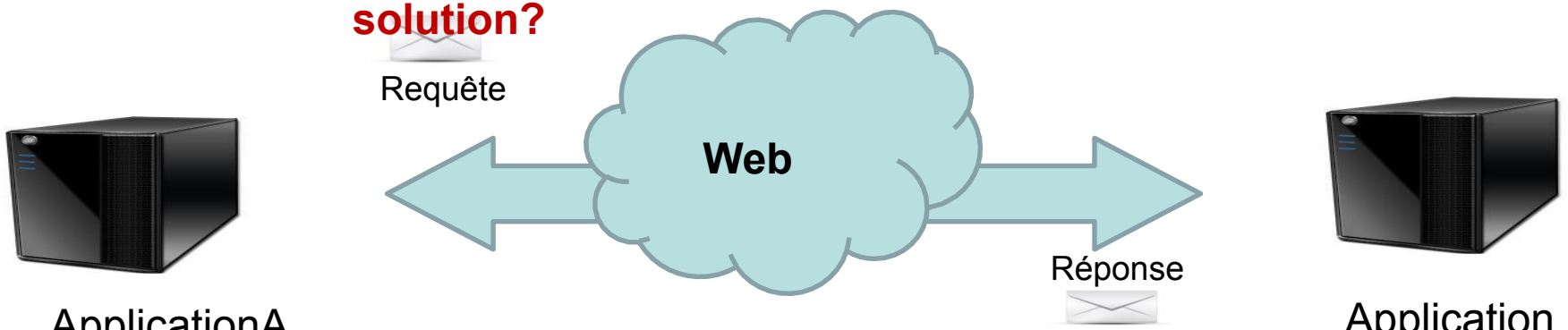
Rôles de base d'un middleware:

- Résoudre l'**intéropérabilité** : Unifier l'accès à des machines distantes
- Résoudre l'**hétérogénéité** : Etre indépendant des systèmes d'exploitation et du langage de programmation des applications

Middleware

- Solutions existantes:
  - DCOM,
  - .NET Remoting
    - Middlware .Net
  - RMI (Remote Method Invocation)
    - Middlware Java permet de faire communiquer des *objets java* **distribués** sur le réseau
  - CORBA (Common Object Request Broker Architecture)
    - Permet de faire communiquer des objets écrits dans des langages différents (C++, Java, Smalltalk)
  -
  -

Quels sont les atouts d'une meilleure solution?



Services web

- Service web = service + web
- “A Web service is a software system designed to support interoperable **machine-to-machine** interaction over a **network**”.

W3C – 2004

- Un SW est un **programme informatique**, permettant la communication et l'échange de données entre applications et systèmes hétérogènes dans des environnements distribués.
- Les services Web interagissent à travers l’échanges de messages

Services web

❑Il existe deux grandes familles de services web:

- Les services web **étendus** utilisant les standards:
  - SOAP pour la communication ;
  - UDDI (annuaire) pour la publication ;
  - WSDL (contrat) pour la description

- Les services web **REST** utilisant :
  - Directement HTTP au lieu d'une enveloppe SOAP ;
  - Un URI pour nommer et identifier une ressource ;
  - Les méthodes HTTP (POST, GET, PUT et DELETE) pour effectuer les opérations de base CRUD

Avantages des services Web ?

- Offrir une technologie adaptée aux applications B2B;

- Rendre possible et plus facile l'interconnexion et l'interaction des systèmes et composants hétérogènes;
- Utilisés par le Web Sémantique pas seulement le web interactif

- Garantir l'interopérabilité et donner lieu à des **systèmes plus ouverts** que ceux utilisant des protocoles tels que **RPC**, **DCOM**, **RMI**… ;
- Réutilisables dans un environnement ouvert;

- Garantir un couplage lâche.

Intégration, Interopérabilité –

- **Interoperability** means that two (or more) systems work together unchanged even though they weren't necessarily designed to work together.

- Réfère à la capacité d'un système à coexister et à coopérer avec d'autres systèmes éventuellement **hétérogènes** selon un schéma ouvert d'interconnexion.

✔Langages de programmation différents,
✔SGBD différents,
✔ Architectures différentes,
✔etc.

Intégration, Interopérabilité –

- **Integration** means that you've writtensome custom code to connect two (or more)systems together.
  - *[Bobby Wolf – IBMArchitect]*

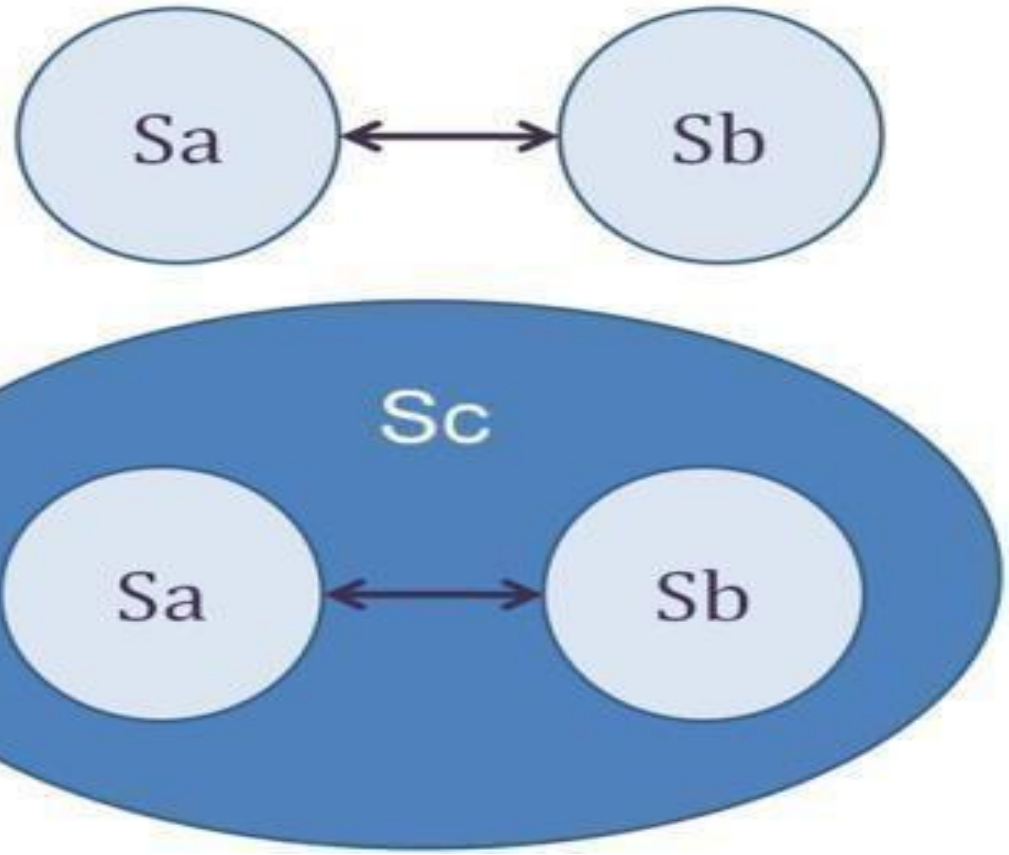
- Lorsqu'on parle d'intégration, nous pensons au processus qui fait que différents systèmes d'information **apparaissent comme un seul**.

Intégration, Interopérabilité

Two heterogeneous Systems Sa and Sb

One « integrated » system Sc

(Does not mean that Sa and Sb have been « merged »)

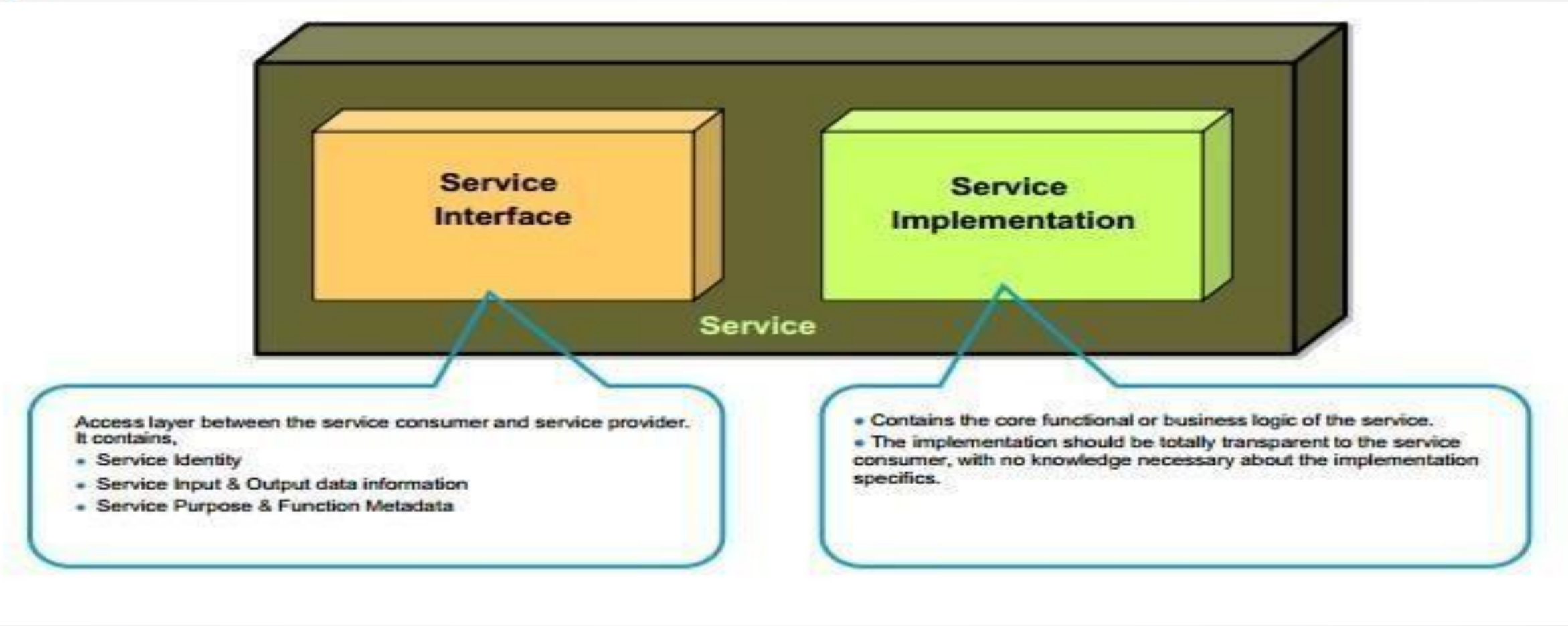


Interoperabilité

Integration

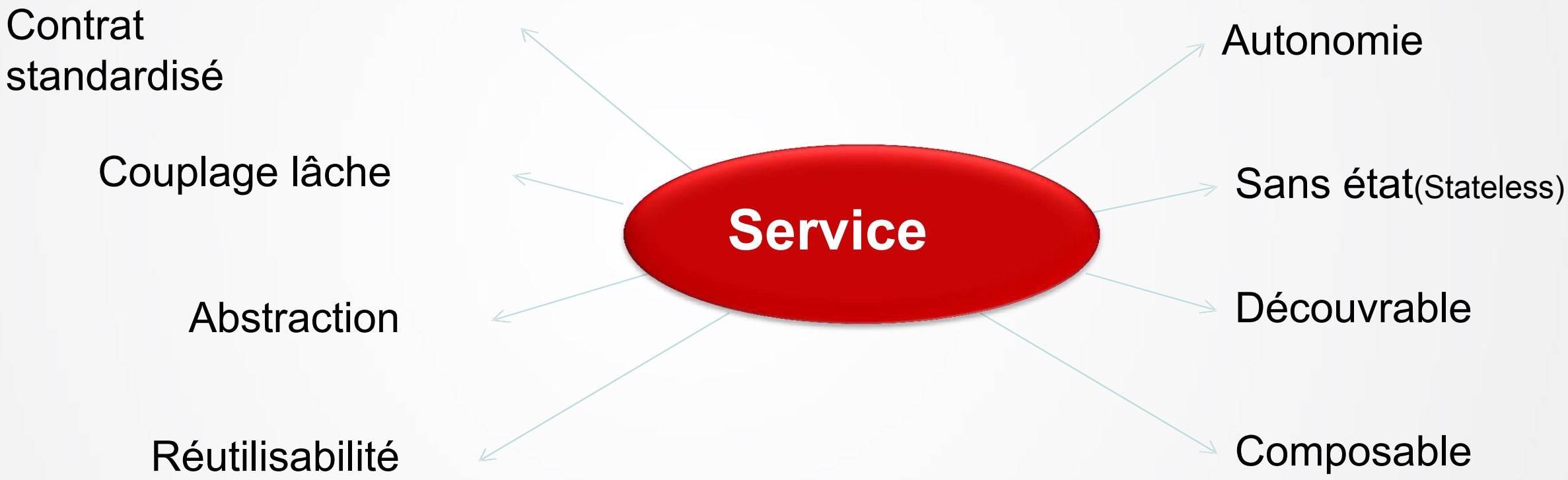
Source: http://modelseverywhere.wordpress.com/2010/11/04/model-driven-integration/

Anatomie d'un service



Source:
http://e-technologymanagement.com/tm/articles/soa.pdf

Caractéristiques d'un service



Contrat standardisé & Abstraction

Contrat standardisé

L'ensemble des services d'un même Système Technique sont exposés au travers de contrats respectant les mêmes règles de standardisation.

Abstraction

Le service fonctionne en « **boîte noire** »

- Seul le contrat du service (informations nécessaires pour l'invocation) est exposé au consommateur du service .

- Le fonctionnement interne du service (sa logique métier et son implémentation) ne sont **pas visibles** .

Couplage lâche

Dépendance faible entre le consommateur et le service

- **Dépendance du contrat** et non pas de l'implémentation;
- Echange à travers des **messages;**
- Orchestration qui assure l'indépendance des services vu qu'elle leur permet de communiquer pour réaliser un processus, sans avoir à se connaître

Autonomie & Stateless

Autonomie

- Un service ne doit être dépendant d'aucun contexte ou service externe:
  - Son comportement est indépendant du contexte fonctionnel et technique dans lequel il a été invoqué.

Stateless

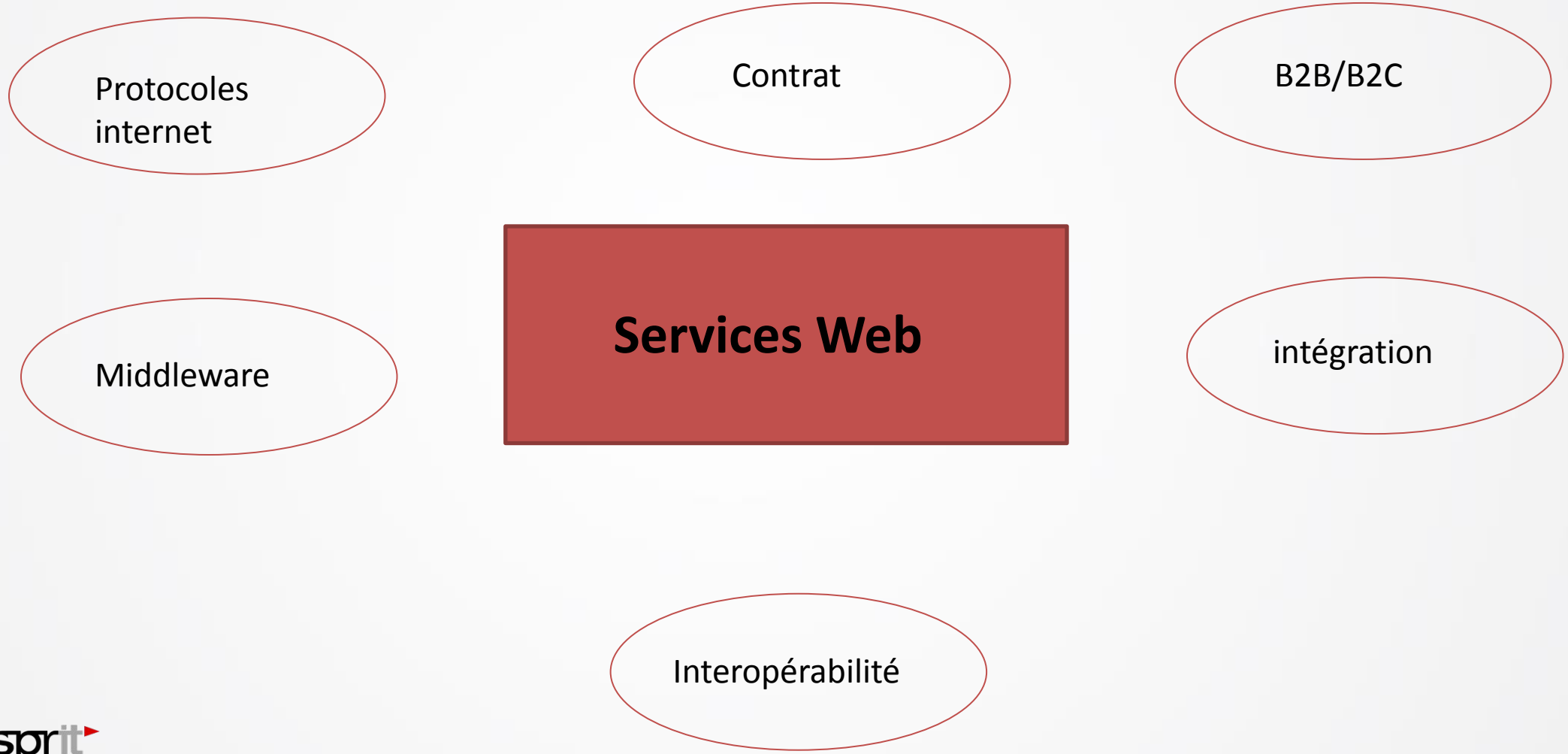
- Un service ne stocke pas les informations des clients:
  - Ne stocke pas de données;
  - Ne fait référence à aucune transaction passée.

Composable

Un service peut participer à des compositions de services

- Un ensemble de services peuvent être composés à travers leur orchestration pour répondre à un besoin complexe;
- **Avantage:**
  - Apport de valeur ajoutée (répondre à un nouveau besoin complexe)

A retenir


Protocoles internet
Contrat
B2B/B2C
Middleware
Services Web
intégration
Interopérabilité

Références

1 http://fr.wikipedia.org/wiki/Paradigme

2 http://design-patterns.fr/introduction-a-la-programmation-orientee-objet

3 http://fr.wikipedia.org/wiki/Middleware

4 http://blog.xebia.fr/2009/04/29/soa-du-composant-au-service-lautonomie

5 https://www.ibisc.univ-evry.fr/~tmelliti/cours/CPAR/cours6.pdf

6 http://adslbox.free.fr/rapports/rapport-gl-service-oriented-architecture.pdf

7 http://deptinfo.unice.fr/~baude/WS/cours_SOA_AO+FB.pdf