

# Les Architectures Orientées Services

SOA

**Introduction aux AOS**

# Plan du cours

- Les AOS et leurs principes
- Le concept de service et ses caractéristiques
- Composition et orchestration de services
- Les éléments de base d'une AOS

## Les AOS

SOA signifie que les composants d'une application donnée agissent comme des **services** interopérables, et qui peuvent être (ré)utilisés indépendamment et combinés par d'autres applications.

Jeff Bezos, email à ses employés, 2002 :

*All teams will henceforth expose their data and functionality through **service** interfaces.*

*Teams must communicate with each other through these interfaces.*

*There will be no other form of interprocess communication allowed: no direct linking, no direct reads of another team's data store, no shared-memory model, no back-doors whatsoever. The only communication allowed is via **service interface** calls over the network.*

*It doesn't matter what technology they use. HTTP, Corba, Pubsub, custom protocols—doesn't matter. [Amazon CEO Jeff] Bezos doesn't care.*

*The teams must plan and design to be able to expose the interface to developers in the outside world. No exceptions. Anyone who doesn't do this will be fired.*

*Thank you; have a nice day!*

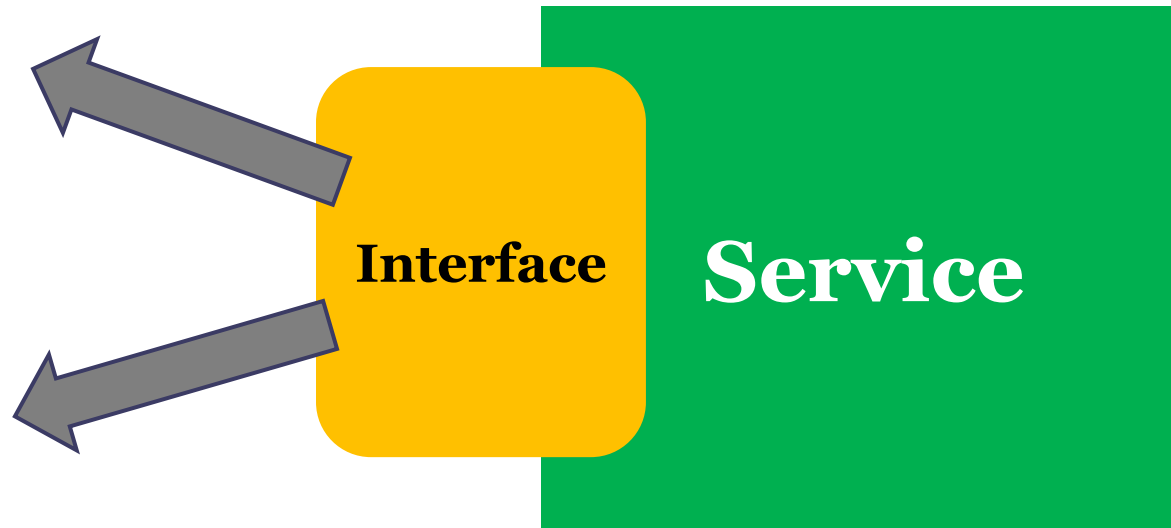
## Qu'est-ce qu'un service ?

- Un **service** est un composant logiciel autonome, exposant les fonctionnalités à forte valeur ajoutée d'un domaine métier.
- Un service expose un petit nombre d'**opérations** offrant un traitement de bout en bout indépendamment de tout système.
- A chaque service doit correspondre un contrat d'utilisation (**contrat de service**) qui permet à ses utilisateurs de comprendre son fonctionnement et usage métier et technique.
- Il est invocable à distance et localisable.

## Les AOS

Données et Activités → Services

Communiquer → Service Interfaces

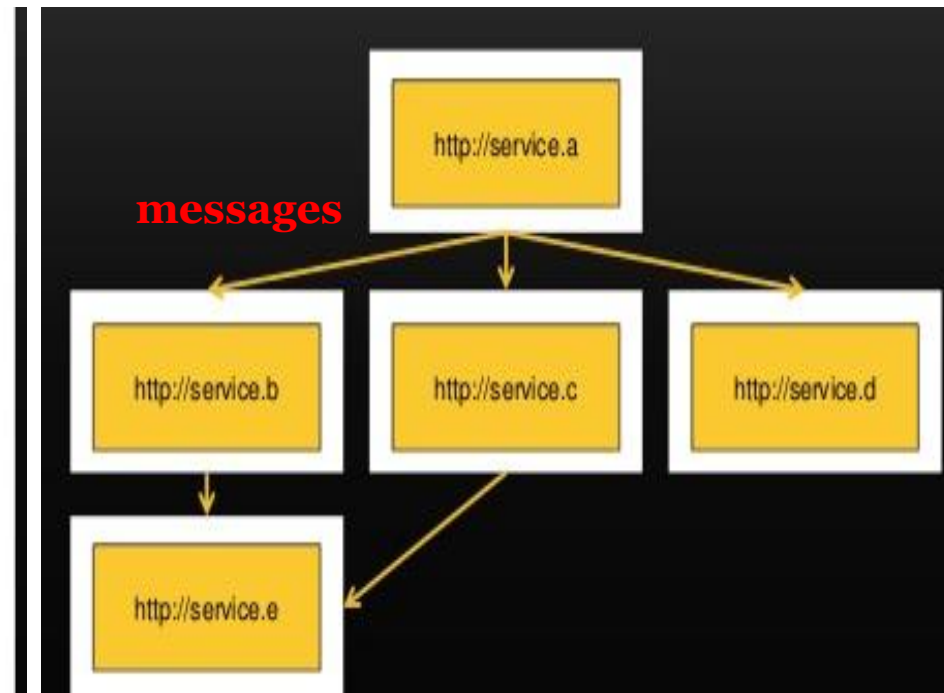
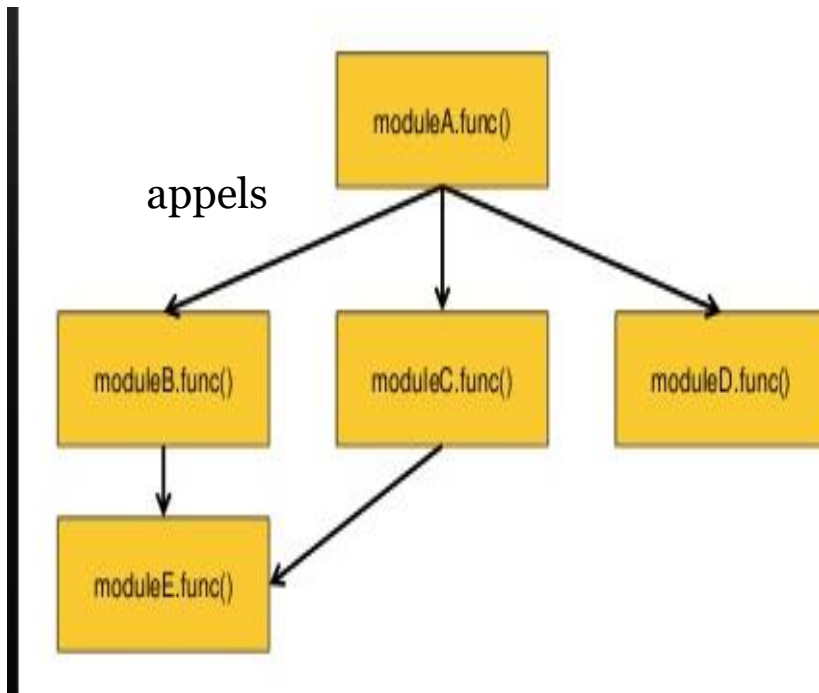


Un service est exposé via une **interface** qui décrit un message en entrée et un autre en sortie.

## Qu'est-ce qu'un service ?

Pour faciliter la réutilisation et l'interopérabilité, tout service SOA devra échanger et fournir le résultat de ses traitements dans des **messages** sous une forme normalisée, c'est-à-dire compréhensible par tous.

Format Pivot.



# Types de services

## 1. Service métier

- Participe à la réalisation d'un ou plusieurs processus métier.
- Ensemble cohérent de traitements métiers.
- Peut être : un service de calcul , un service de vérification, etc.

## 2. Service technique

- Correspond à l'exposition d'un composant technique.
- Donnent accès à des ressources techniques : BDD, imprimante, etc.

Un **service métier** peut s'appuyer sur un ou plusieurs **services techniques** pour exécuter ses traitements.

# La gestion du cycle de vie des services

- La gouvernance du cycle de vie des services est un élément clé d'une démarche SOA.
- 4 grandes phases :
  - Identifier les services à mettre en place (élaboration des contrats, etc.)
  - Développer les services (en vue la réutilisation)
  - Intégrer et déployer les services (règles, orchestration, etc.)
  - Exploiter et superviser les services (changement de versions, états, QoS)



# Principes SOA

- **Diviser pour régner**
  - Substituer la découpe strictement applicative par une structuration en composants plus réduits et potentiellement plus simples à faire évoluer.
- **Alignement métier**
  - Construire et organiser le système à partir des réalités métiers, qui doivent se retrouver dans ses constituants.
- **Neutralité technologique**
  - Assurer une indépendance totale entre les interfaces et les implémentations.
  - L'élément qui utilise un service ne doit pas être contraint ni par la technologie d'implémentation, ni par sa localisation (potentiellement distribué).

# Principes SOA

- **Mutualisation**

- Favoriser la réutilisation de services métiers par plusieurs lignes métiers ou applications.
- Permettre la construction de services de haut niveau par combinaison de services existants.

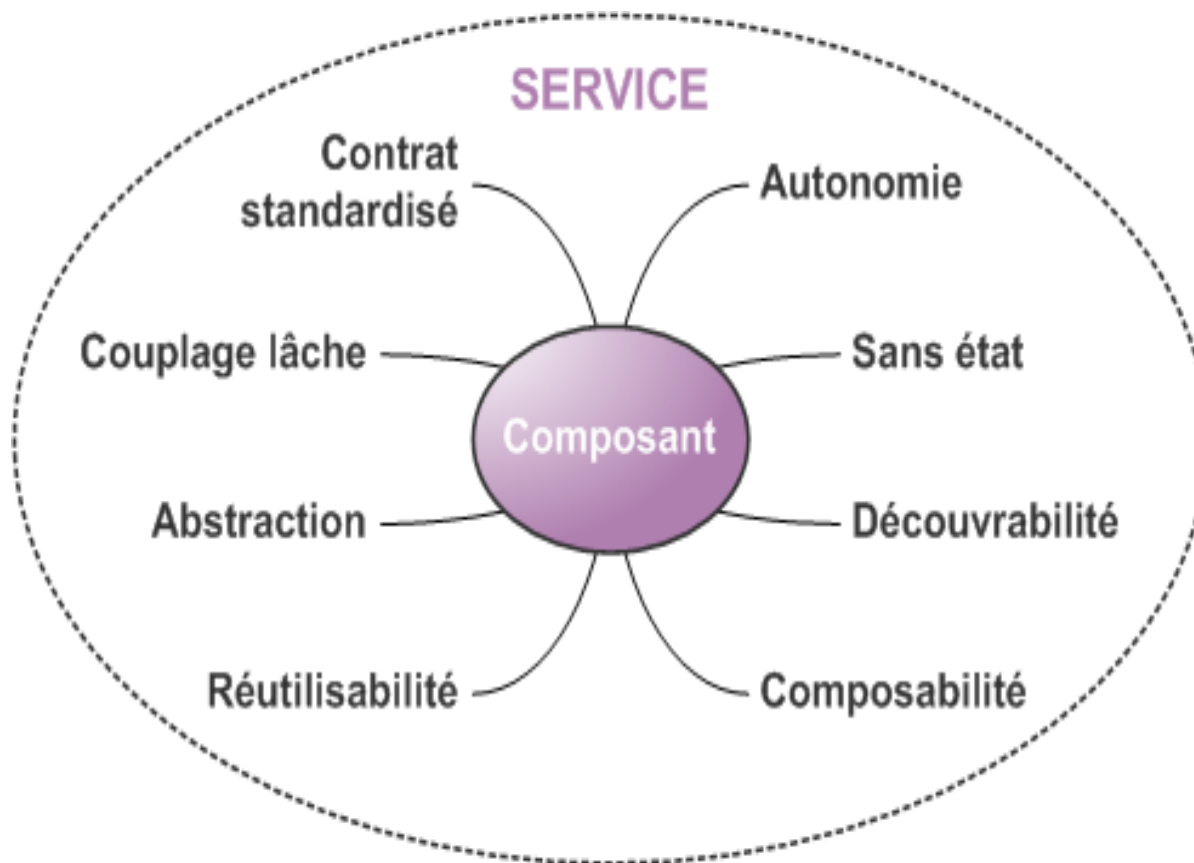
- **Automatisation des processus métier**

- Isoler la logique des processus métiers sur des composants dédiés qui prennent en charge les enchainements de tâches et les échanges de flux d'information.

- **Echanges orientés Document/Message**

- Les informations échangées par les services possèdent une structure propre, guidée par les besoins métiers.

## Caractéristiques d'un service



Ces 8 aspects sont issus du livre « SOA Principles of Service Design » de **Thomas Erl**, également auteur du site SOA Principles.

# Caractéristiques d'un service

## **Contrat standardisé**

Contrat entre le fournisseur de service et le consommateur de service qui décrit les entrées, les sorties, et le mode de fonctionnement.

Trois types de contrat sont à distinguer :

- Lié à la syntaxe du service (opération, messages d'entrée, messages de sortie, ...)
- Lié à la sémantique du service (définition de règles et de contraintes d'usage, ...)
- Lié à la qualité de service (temps de réponse attendu, procédures en cas de panne, temps de reprise après interruption,...)

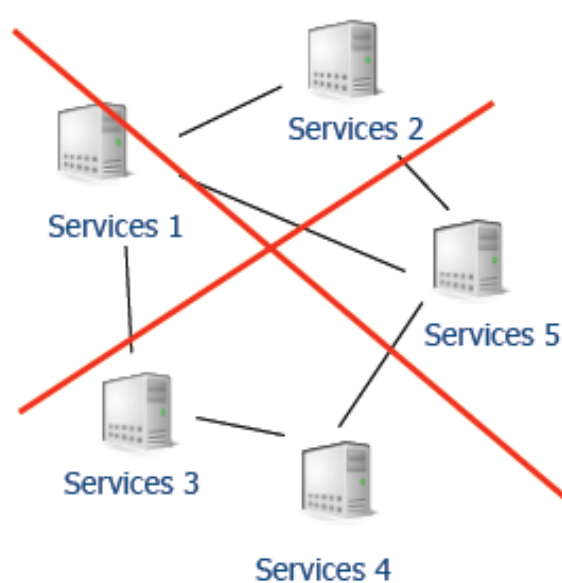
S'appuie sur des standards d'interopérabilité pour faciliter le dialogue (exemple : WSDL)

# Caractéristiques d'un service

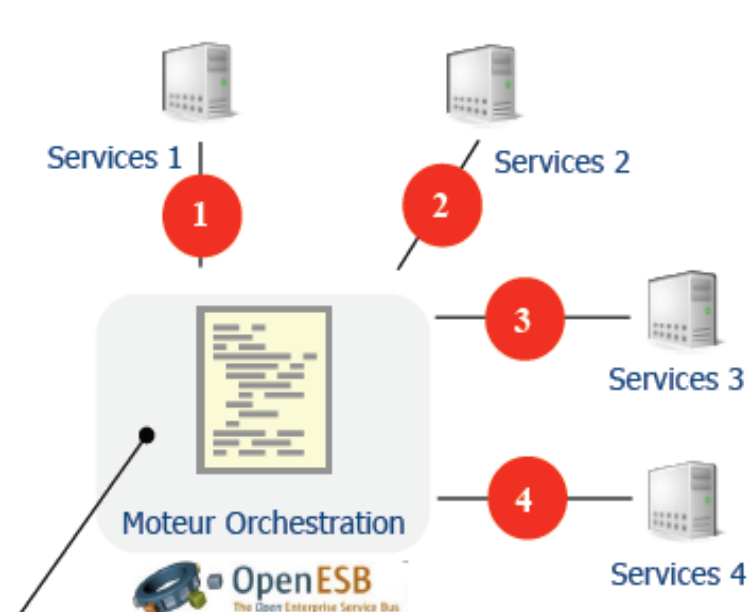
## Couplage lâche

L'échange entre le fournisseur de service et le consommateur doit se faire à travers des messages (couplage lâche vis-à-vis de son environnement).

L'utilisation d'un service BUS/hub évite que les services aient besoin de connaître les autres services.



Couplage Fort



Couplage Lâche

## Caractéristiques d'un service

### **Réutilisabilité/Découvrabilité**

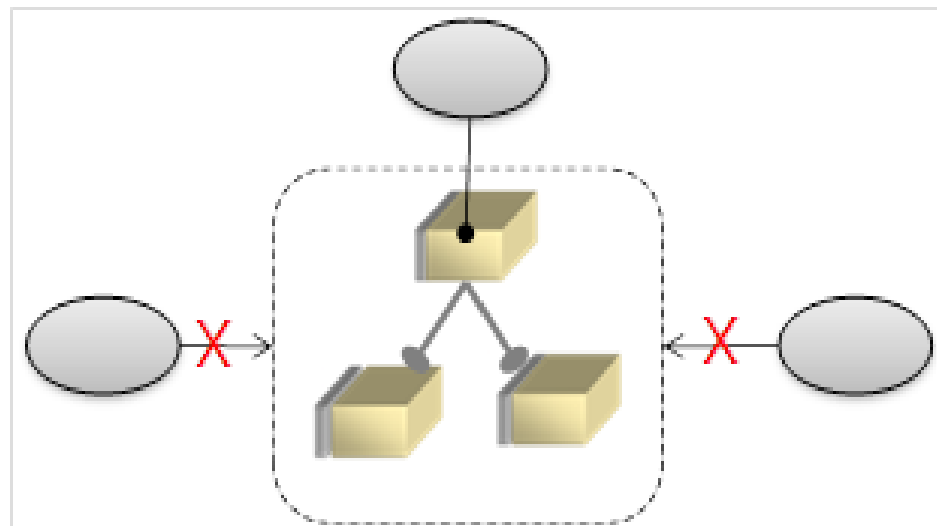
- La Découvrabilité passe par la mise en œuvre d'un référentiel de services qui vient outiller l'inventaire des services disponibles.
- Le fournisseur de services a la charge de déposer et de mettre à jour ses services dans le référentiel (repository).
- Le service est enrichi par un ensemble de métadonnées pour faciliter la recherche du consommateur de services.
- S'appuie sur des standards (UDDI).
- Un service doit être accessible depuis un annuaire (registry) de service ou un annuaire pour faciliter sa découverte.
- D'après la gouvernance SOA, un service est défini avec l'intention d'être réutilisé.
- La réutilisation est une condition première de l'agilité du SI indispensable à la réduction du time-to-market.

# Caractéristiques d'un service

## Autonomie

➤ Un service doit disposer :

1. de l'ensemble des informations nécessaires à son exécution.
2. ne doit dépendre d'aucun service externe (couplage lâche).



# Caractéristiques d'un service

## Sans état

- Les services ne sont pas des objets : ils ne conservent aucun état.
- Une seule instance du service est disponible.
- L'état est un attribut transmis par le consommateur en entrée de chaque opération du service.

### Interface

```
créer(typeFacture): typeFacture  
modifier(typeFacture): typeFacture  
emettre(typeFacture): typeFacture  
regler(typeFacture): typeFacture  
annuler(typeFacture): typeFacture
```



# Caractéristiques d'un service

## **Abstraction**

- Le contrat du service ne doit contenir que les informations pertinentes à son invocation.
- Fonctionnement du service dit en « boîte noire ».
  - Seul le contrat exposé au consommateur du service est connu
  - Le fonctionnement interne du service ne doit pas être visible
    - Logique métier
    - Implémentation
  - Le service encapsule les traitements et données existants ou à développer dans des boîtes noires et masque ainsi l'hétérogénéité du système d'information.

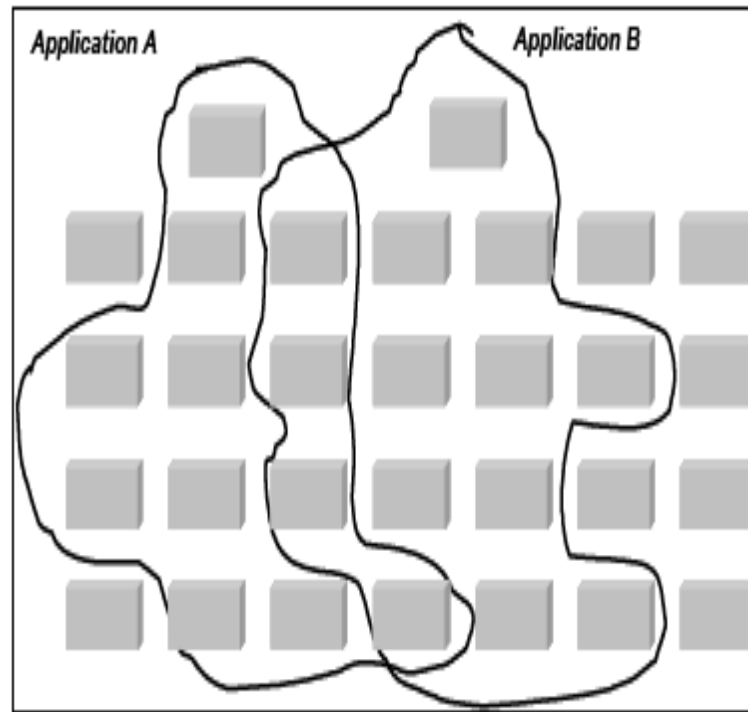
# Caractéristiques d'un service

## Composabilité

- Plusieurs services peuvent être composés pour former un service de plus haut niveau. Leur composition est appelée **Chorégraphie**.
- Un service doit fonctionner de manière modulaire et non pas intégrée.
- Assurer la décomposition d'un service complexe en sous services plus simples entre eux (garantie l'autonomie).
- S'inscrire dans une logique de composition de services à travers l'utilisation de l'orchestration (couplage lâche).
- **L'orchestration** favorise l'indépendance des services et assure que des services n'appellent pas directement d'autres services.

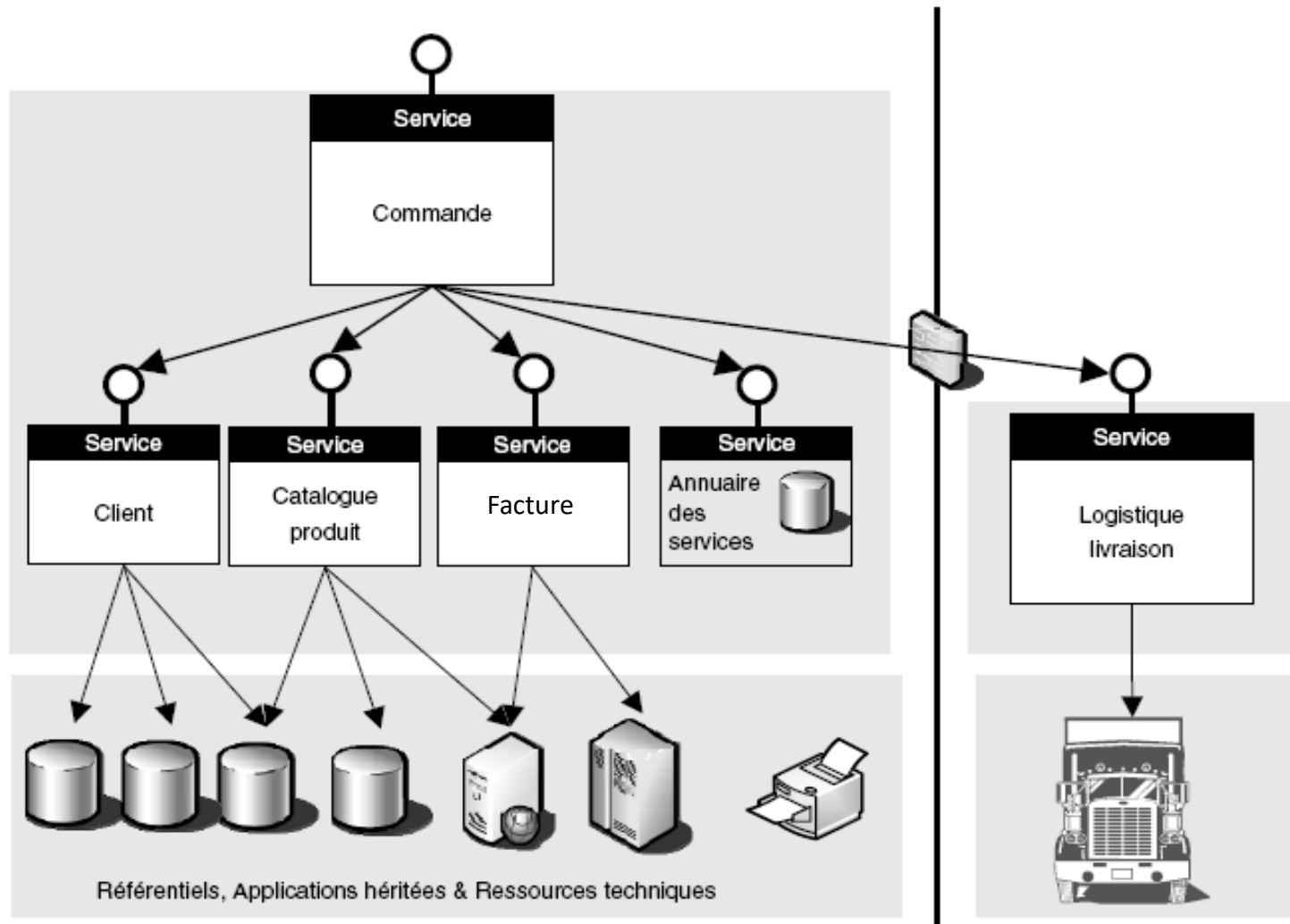
## Composition de services

- Implémentation d'une application dont la logique implique l'invocation d'opérations offertes par d'autres services.
  - Le nouveau service est appelé **service composite**
  - Les services invoqués sont des **composants de service**



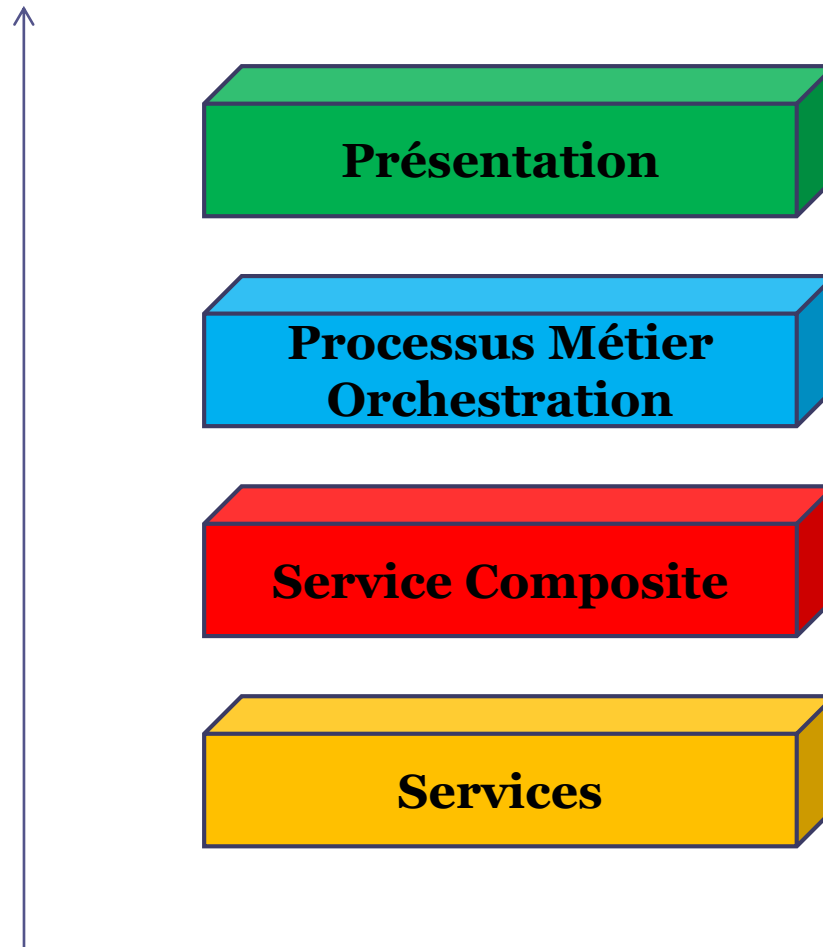
# Composition de services

Exemple: Composition – interne et/ou externe



# Hiérarchie de services

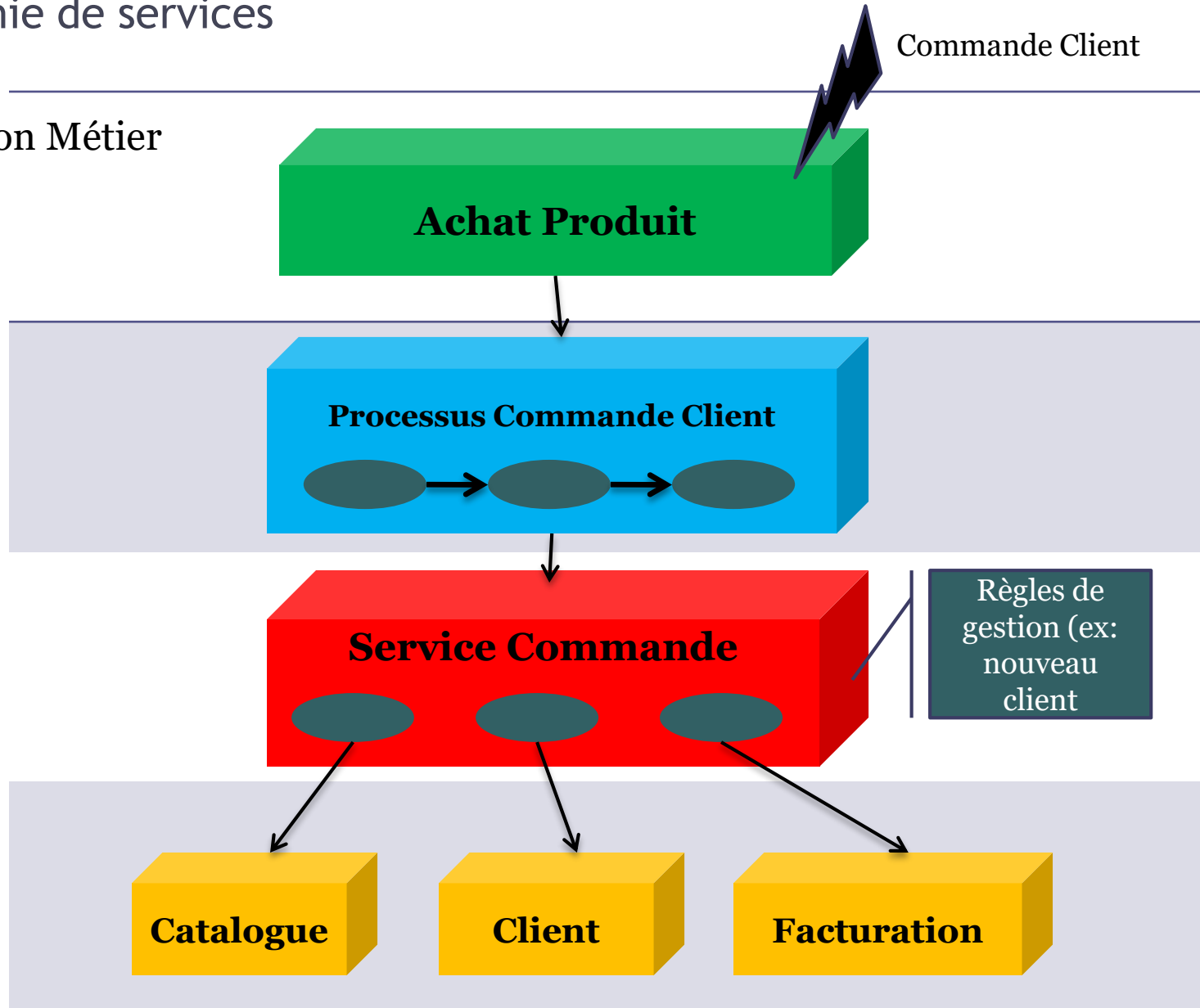
Les couches Logiques



# Hiérarchie de services

Solution Métier

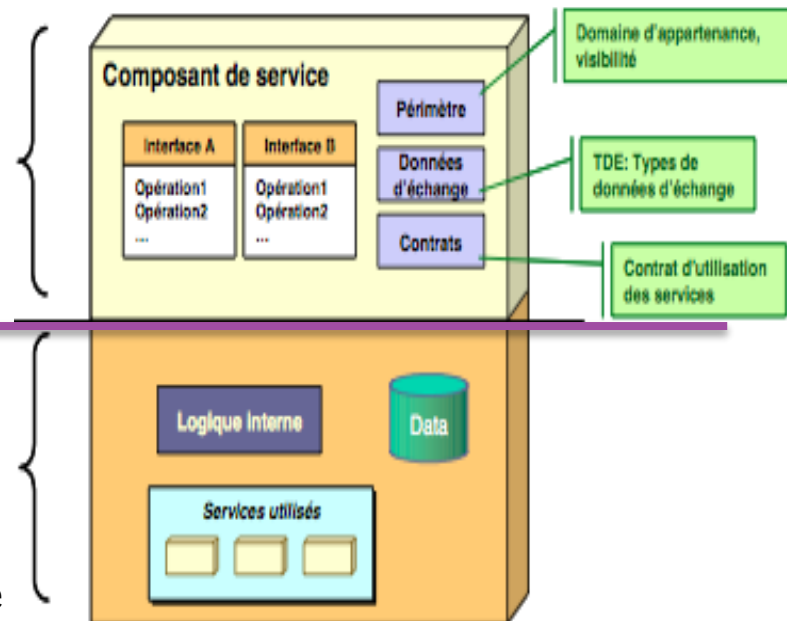
Commande Client



# Éléments de base SOA

## Composant de service

- Brique de base de l'architecture, composé d'une vue externe et interne
- La **vue externe** :
  - Expose la facette service proprement dite
  - Constituée :
    - d'un ensemble d'opérations de service regroupées en interfaces
    - appareillage pour les utiliser (types de data échangées, contrat de service, propriétés...)
  - Décrite par un fichier WSDL ou équivalent
- La **vue interne** :
  - Décrit le contenu du composant
  - Masquée aux consommateurs du composant
  - Contient des informations relatives à la logique interne (détail de traitement ou bases de données) + références vers les services utilisés par le composant



# Éléments de base SOA

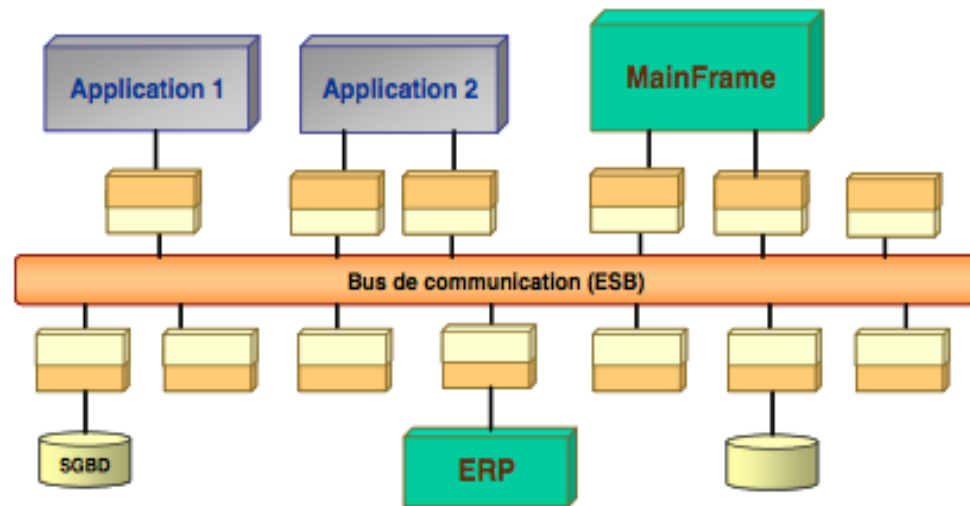
## Bus d'entreprise

Présence de plusieurs *participants* :

*Fournisseurs de service* : composants de service, deux familles:

- Composants qui prennent en charge l'implémentation des services
- Composants qui délèguent son implémentation à un tiers (ERP, application existante)

*Consommateurs de service* : applications, ou autres composants de service

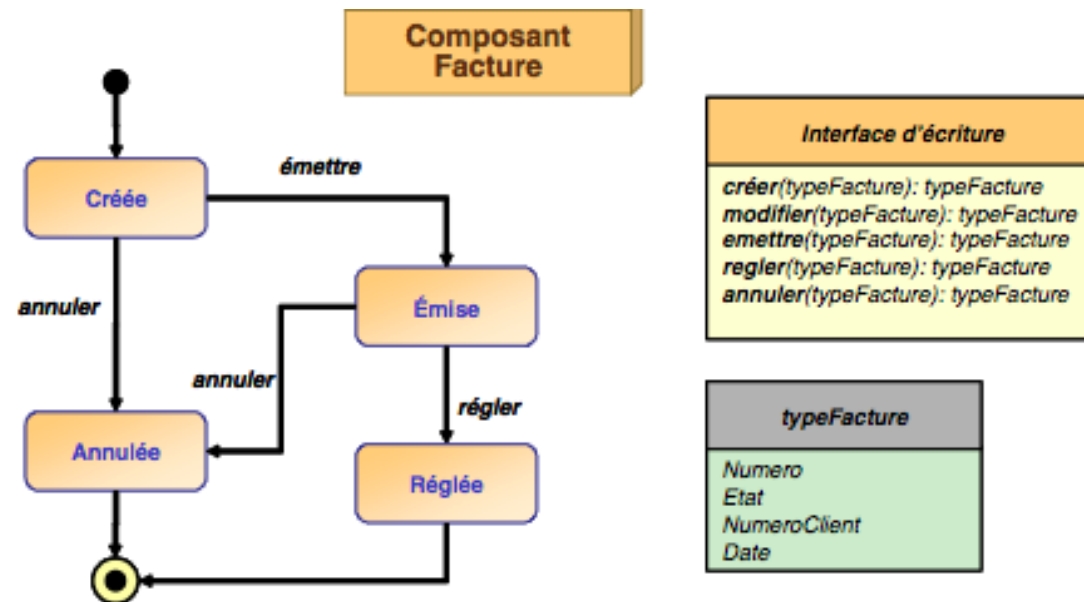




# Éléments de base SOA

## Contrat de service (Interface)

- Détaille les conditions d'utilisation du service sous forme de:
  - **Pré- et Post- conditions** : Détaillent les conditions d'utilisation sur les opérations de service
  - **Protocole d'utilisation**: les séquences valides d'invocation de ses opérations
  - **Contraintes** (QoS, SLA: *Service Level Agreement*, sécurité, fiabilité...)

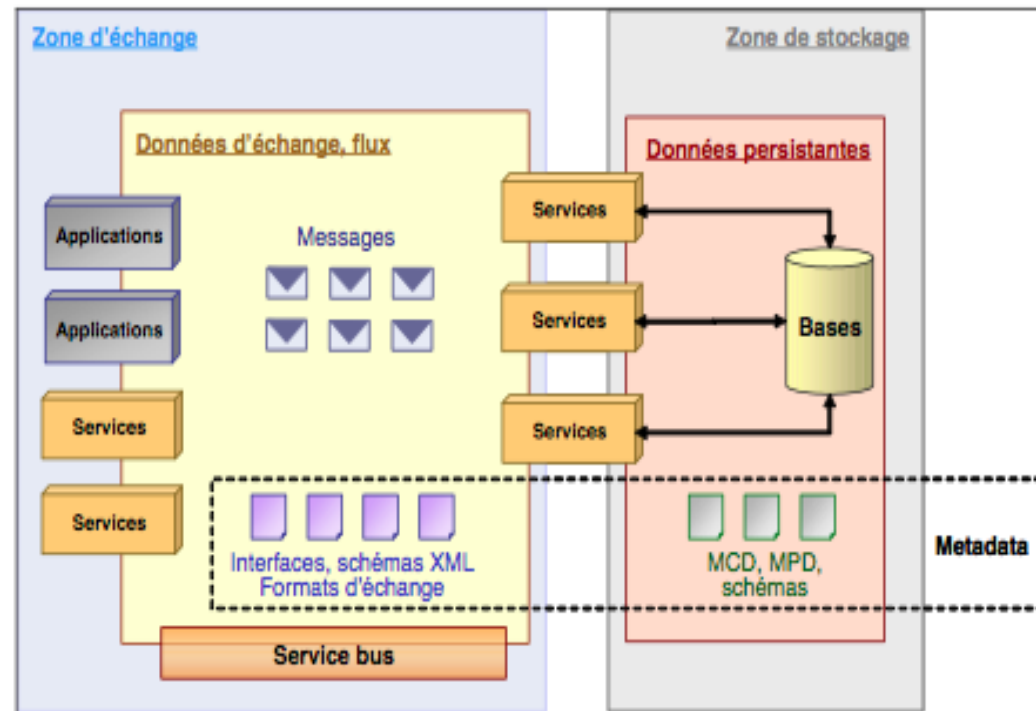


# Éléments de base SOA

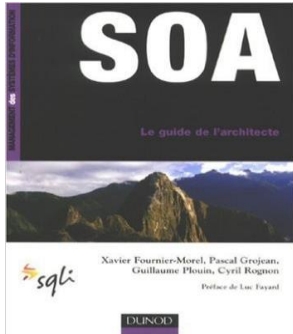
## Données d'échange (messages)

- Données d'échange:
  - Informations véhiculées entre les participants à travers l'invocation des opérations de service
  - **TDE** : Types de donnée d'échange : établissent la sémantique, structure et format de ces données, définis à l'aide de schémas XML par exemple.

- Données persistantes
  - Informations contenues et gérées dans les bases de données
  - Structurées de façon habituelle (SGBD relationnel, par exemple)

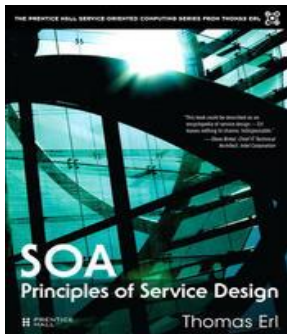


# Ressources



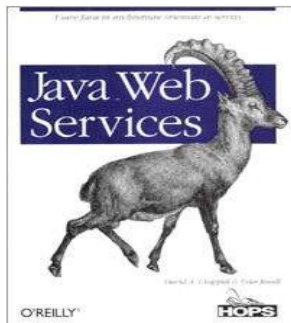
## **Le guide de l'architecte du SI**

- ✓ Auteur : Xavier Fournier-Morel, Pascal Grosjean, ...
- ✓ Éditeur : Dunod
- ✓ Edition : Octobre 2006 - 302 pages - ISBN : 2100499726



## **SOA Principles of Service Design**

- ✓ Auteur : Thomas Erl
- ✓ Éditeur : Prentice Hall Ptr
- ✓ Edition : Juillet 2007 - 608 pages - ISBN : 0132344823



## **Java Web Services**

- ✓ Auteur : David Chappell & Tyler Jewell
- ✓ Éditeur : O'Reilly
- ✓ Edition : Mars 2002 - 276 pages - ISBN : 0-596-00269-6

# Ressources

## **Engineering Long-Lasting Software: An Agile Approach Using SaaS and Cloud Computing**

- ✓ Auteur : Armando Fox and David Patterson
- ✓ Éditeur : Strawberry Canyon LLC
- ✓ Edition : Aout 2012 - 412 pages - ISBN : 0984881212

Livre blanc SOA : Architecture Logique : Principes, structures et bonnes pratiques Auteur: Gilbert Raymond.Version 2.

Cours – Mickael Baron – SOA et Microservices

- ✓ [http://mbaron.developpez.com/#page\\_soa](http://mbaron.developpez.com/#page_soa)

Blog – Xebia

- ✓ <http://blog.xebia.fr/>