

This is the html version of the file <https://moodle.polymtl.ca/mod/resource/view.php?id=255464>.
Google automatically generates html versions of documents as we crawl the web.

LOG8430 : Architecture logicielle et conception avancée

Yann-Gaël Guéhéneuc
Fabio Petrillo

Chargé de Cours

Introduction à aux architectures orientées services (SOA)

(Contributeurs : Occello, 2007; Khomh, 2013-2014;
Soh, 2014; Guéhéneuc, 2015, Petrillo, 2016)

This work is licensed under a Creative
Commons Attribution-NonCommercial-
ShareAlike 3.0 Unported License

Page 2

1. Contexte d'intégration

- Les systèmes logiciels reflètent très souvent **l'organisation de l'entreprise (Conway's Law)**
- Processus métiers des entreprises sont de plus en plus **multi-départementaux**
- Quels problèmes ?
 - **Redondance** dans les systèmes logiciels
 - **Coûts** considérables dans la gestion des flux entre départements et dans **l'intégration** de leurs systèmes logiciels

1. Contexte d'intégration

- Développements coûteux
- Interconnexions redondantes (point à point)
- Grande complexité
- Maintenance difficile
- Réutilisation difficile (couplage fort)

3

Page 4

1. Contexte d'intégration

- Les entreprises doivent s'adapter en permanence aux variations des marchés
 - Fusions, Acquisitions, etc.
- Les variations du marché ont un impact sur les systèmes des entreprises
 - Leurs systèmes logiciels ne doivent pas être un frein à ces changements

Service-oriented architecture

- A service-oriented architecture (SOA) is a **style of software design** where **services** are provided to the other **components** by application components, through a communication **protocol over a network**.
- Principles of service-oriented architecture

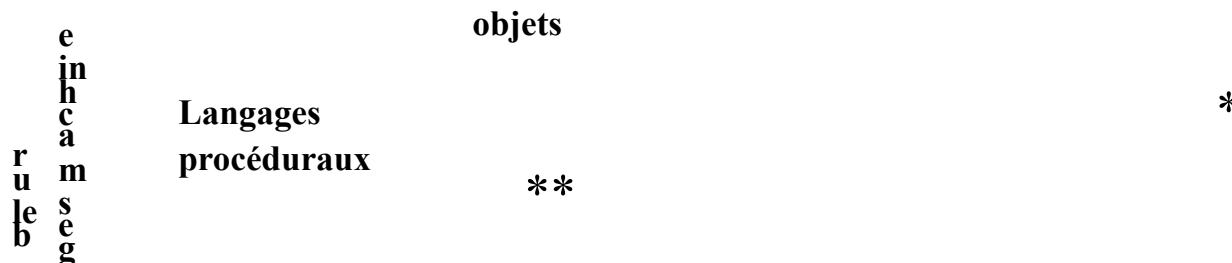
- independent of vendors
- products
- Technologies.

□ A **service** is a discrete **unit of functionality** that can be accessed **remotely** and acted upon and updated **independently**.

5

Page 6

Qu'est ce qu'une SOA?



Les SOA sont une évolution des paradigmes précédents...

6

Page 7

2. Principes de base des SOA

- SOA est une évolution des plates-formes
 - Elle préserve les caractéristiques réussies des architectures traditionnelles
 - Tout en y ajoutant quelques principes nouveaux

- SOA est un paradigme abstrait, base de l'architecture distribuée sans aucune référence à une implémentation technique
 - Elle est souvent implémentée sous forme de Web Services, mais pas obligatoirement

7

Page 8

2. Principes de base des SOA

- SOA est une approche de conception où plusieurs **services** collaborent pour fournir un certain nombre de capacités de fin.

- Qu'est ce qu'un service?
 - Un service doit être "abstrait" : il n'est pas lié à une implémentation
 - Ex. :
 - Service d'enregistrement d'un abonné
 - Service de réservation d'un billet d'avion
 - Service de diffusion d'information

8

Page 9

Qu'est ce qu'un service ?

- Partage des caractéristiques des objets
 - Modulaire (fonctionnalités cohérentes)
- Partage des caractéristiques des composants
 - Boite noire (séparation interface/implémentation)
 - Indépendant de la localisation

– Neutralité vis-à-vis des protocoles de transport

- Correspond à un périmètre fonctionnel que l'on souhaite exposer à des consommateurs
- Expose un petit nombre d'opérations offrant un traitement de bout en bout
- Est sans état
- Est faiblement couplé (indépendant d'autres services)

9

Page 10

Exemple de couplage fort

Entités

LoanAgent

LoanApproval

Account

Loan

SMSTGateway

calculateRisk

checkCredit

*createLoan**sendConfirmation*

- ☐ LoanAgent est lié à LoanApproval
- ☐ LoanApproval est lié à Account
- ☐ Loan est lié à SMSGateway

10

Page 11

Gestion de prêts en couplage faible

Services

LoanProcessCheckAccount
BalanceCalculate
LoanRisk

CreateLoan

Notify
ViaSMS

Qu'est ce que LoanProcess ?

☐ Un **processus métier** !

Il permet d'**orchestrer** les services => couplage faible

11

Page 12

Quatre propriétés des services

☐ Un Service est **autonome**
et **sans état**

☐ Un Service expose un **contrat**

☐ Les frontières entre services
sont **explicites**

☐ Les services **communiquent**
par messages

12

Page 13

Conséquences de ces propriétés

☐ Une SOA transmet des **messages**
– **Pas des objets**

- Le consommateur (client) est **découplé** de l'architecture technique du service qu'il invoque
- Le consommateur et le fournisseur n'ont pas forcément les mêmes technologies

13

Page 14

2. Principes de base du SOA

- Au cœur des SOA on a donc

— Des services

– Des processus

14

Page 15

2. Principes de base du SOA

□ Processus métier

– « Un ensemble d'activités qui s'enchaînent de manière chronologique pour atteindre un objectif,

généralement délivrer un produit ou un service,
dans le contexte d'une organisation de travail » –
Wikipédia

- Un flux d'informations au sein d'une organisation
- Une vue transversale sur les activités de l'entreprise

15

Page 16

Business Process Management

- But : donner à l'entreprise les moyens de gérer ses processus métiers de manière informatisée (modélisation, simulation,
- Un processus est composé de sous processus, de décisions (Business rules) et d'activités
 - Un sous processus a son propre but, entrées et sorties

exécution et audit)

- Optimisation, adaptation aux besoins en temps réel

- Un processus est le résultat d'une orchestration de service
- Le processus est lui-même accessible en tant que service

□ Les activités

- Correspondent aux parties du processus métier sans décision, associées à des rôles
- Sont réalisées par des systèmes ou des humains

- Des mesures (KPI: Key Performance Indicators) permettant de capturer les performances du processus

16

3. Standards de l'architecture

- Les standards sont un élément clé d'une SOA, ils assurent l'interopérabilité

SOAP W3C Simple Object Access Protocol	WSDL W3C Web Services Description Language	UDDI Microsoft, IBM, HP Universal Description Discovery and Integration	BPEL Oasis Business Process Execution Language
Transporte	Décrit le contrat	Spec pour Repository/Registry	Décrit les processus métier

Les trois piliers des Services Web

17

Page 18

SOA et web services

- Attention à ne pas confondre les deux !
 - SOA est un ensemble de concepts

- Une SOA peut être mise en œuvre sans Web Services
 - Les WS sont de l'ordre de la technologie
 - On peut utiliser les Web Services sans faire de SOA
- Les WS constituent la meilleure solution standardisée disponible
 - Un service métier = un Web Service

18

Page 19

Le langage BPEL

- Standard de l'OASIS
 - Permet de décrire des processus en XML

- Propose les fonctions basiques d'un langage de programmation
 - Sequence, flow, loop, switch...

- Identification des instances de processus
- Gestion des transactions longue durée (scope, compensation)
- Gestion des fautes

19

Page 20

BPEL, le chef d'orchestre

Phase d'identification

- Un des problèmes principaux pour mettre en œuvre SOA

- La granularité des services est fondamentale
 - Détermine en grande partie la réutilisabilité des services
 - Succès SOA = % de réutilisation des services
- Éviter une granularité trop fine qui entraîne
 - Beaucoup d'interactions
 - Des problèmes de performance
- On recommande des services à “gros grain”
 - Attention à une granularité trop “épaisse”
 - Un service qui fait trop de chose, risque de ne pas être réutilisable
- Trouver le juste milieu...

21

Page 22

Méthodes d'identification

- | | |
|---|---|
| <ul style="list-style-type: none">□ Une première phase d'identification doit être effectuée sur l'ensemble du | <ul style="list-style-type: none">□ Approche bottom-up<ul style="list-style-type: none">– On part des briques informatiques, on rassemble les |
|---|---|

système en s'appuyant sur les domaines métiers de l'entreprise et sur le code existant

bouts (abstraction)

– Plus adéquat pour réutiliser l'existant non “SOA”

- Approche incrémentale : une phase d'identification est nécessaire au démarrage de chaque nouveau projet SOA en s'appuyant sur les processus et services répertoriés précédemment

- Approche Top-down

- On part des interactions métier pour aboutir aux interactions techniques
- Plus adéquat pour démarrer un nouveau projet

22

Méthodes d'identification - Approche “Bottom Up”

Besoins

Legacy applications

Diagrammes
d'activités

Décomposition du
diagramme de classes

Orchestration

Specification des services

Nouveaux Services +
services réutilisables
(l'existant)

Nouvelle application

23

Page 24

Méthodes d'identification - Approche “Top Down”

Besoins

Analyse des domaines métiers

Décomposition du processus métier

Orchestration

Specification des services

Nouveaux Services +
services réutilisables
(l'existant)

Nouvelle application

24

Page 25

Approche “Outside in”

□ Dans la pratique, on utilise les deux approches

- Pour obtenir une granularité pertinente des services, il est nécessaire de concilier les deux
 - Faire l’analyse Top-down sans considérer l’existant
 - Faire l’analyse Bottom-up en ne considérant que l’existant
 - Comparer les services “remontés” avec ceux déduits des processus
 - Faire les compromis nécessaires pour réutiliser le maximum de code

25

Page 26

Huit principes de bases d’une SOA

- Standardized service contract : le contrat de service adhérent

à un accord de communication, collectivement défini avec un ou plusieurs documents de description

- Service loose coupling : faible couplage des services avec la maintenance d'une relation réduisant les dépendances
- Service abstraction : l'abstraction des services doit dissimuler la logique du service à l'extérieur
- Service reusability : réutilisation des services partageant la logique entre plusieurs services avec l'intention de promouvoir la réutilisation

26

Page 27

Huit principes de bases d'une SOA

- Service autonomy: services have control over the logic they

encapsulate

- Service statelessness: services minimize resource consumption by deferring the management of state information when necessary
- Service discoverability: services are supplemented with communicative meta data by which they can be effectively discovered and interpreted
- Service composability: services are effective composition participants, regardless of composition size and complexity

27

5. Avantages des SOA : bénéfices métier

- Améliorer l'agilité et la flexibilité du métier
- Faciliter la gestion des processus métier
- Offrir la capacité à casser les barrières organisationnelles (silos)
- Réduire en temps le cycle de développement des produits
- Améliorer le retour sur investissement
- Accroître les opportunités de revenu

28

5. Avantages des SOA : bénéfices techniques

- Réduire la complexité de la solution
- Construire les services une seule fois et les utiliser fréquemment
- Garantir une intégration standardisée et le support de clients hétérogènes
- Faciliter la maintenabilité

Modeleurs de processus

- Outils de modélisation des processus métier
 - IBM WebSphere Business Modeler
 - Bull Bonita
 - De Gamma BPM
 - MEGA
 - Aris
 - Corporate Modeler
 - WinDesign
 - Power AMC
 - Popkin System Architecture

30

Moteurs d'exécution de processus

□ Plate-forme d'intégration

- IBM Websphere Process Server
- BEA Weblogic Integrator/Aqualogic
- Microsoft Biztalk
- De Gamma Workflow
- Oracle BPEL PM
- Bull Orchestra
- SAP “Netweaver”

□ ESB

- IBM Websphere ESB
- Celtix hosted on ObjectWeb/IONA Technologies
- OpenESB (java.net)
- Mule (codehaus.org)
- Sonic ESB
- EBM Web Sourcing Distributed Petals Bus (on OW2)

31

jBPM – Business Process Management (BPM) Suite

<http://www.jbpm.org/>

32

Page 33

5. Inconvénients des SOA

- Difficile à tester
- Risque de prolifération des messages (entre services)
- Risque liés à la sécurité des messages provenant de sources diverses

<http://nealford.com>

34

Page 35

<http://nealford.com>