



République Algérienne Démocratique et Populaire  
Université Abou Bakr Belkaid– Tlemcen  
Faculté des Sciences  
Département d'Informatique

Mémoire de fin d'études

pour l'obtention du diplôme de Master en Informatique

*Option: Réseau et Système Distribué (R.S.D)*

*Thème*

**Conception et Réalisation d'une  
Application basée sur l'Architecture  
SOA : Traitement des Appels d'Offres**

Réalisé par :

- CHAIF Lamia

*Présenté le 03 Juillet 2013 devant le jury composé de MM.*

- BENMAMMAR Badr (Président)
- LEHSAINI Mohammed (Encadreur)
- BELHOCINE Amin (Examineur)
- BOUDEFLA Amine (Examineur)

Année universitaire : 2012-2013



# RESUME

## RESUME

De nos jours, l'environnement concurrentiel des SI est de plus en plus rude et évolue rapidement. Cet accroissement est causé par plusieurs facteurs : la technologie de l'information et de communication.

Les architectures orientées services (SOA) offrent aujourd'hui un nouveau modèle pour permettre de mettre le SI au service des métiers, en procédant à son alignement avec les processus d'entreprise. Dans le cadre de notre PFE, on a proposé d'inclure l'architecture SOA dans le traitement des appels d'offre.

**Mots clés** : SOA, SI, WSOA, Service, Service Web, Appel d'offre, entreprise, processus métier,

## ABSTRACT

Nowadays, the competitive environment of IT is becoming increasingly harsh and rapidly changing. This increase is caused by several factors: the information technology and computers.

Service-oriented architectures (SOA) now offer a new model for put IT to serve the business by performing alignment with business processes. Via the work presented in this thesis, we propose to include the SOA in the treatment of supply calls.

**Keywords:** SOA, TI, WSOA, Service, Service Web, supply calls, business, business processes.

## ملخص

في الوقت الحاضر تعد البيئة التنافسية لرجال اعمال المؤسسات في تزايد مستمر و هذا يعود الى عدة عوامل هي : تكنولوجيا المعلومات والاتصالات

البنية الخدمية تقدم نموذج جديد لتجعل نظم المعلومات في خدمة الهيكليات التجارية, العمل المقدم في هذه المذكرة يقترح ادماج البنية الخدمية في نظام المناقصات

**الكلمات الرئيسية** : البنية الخدمية, نظام المعلومات, خدمة الواب , الأعمال التجارية , المناقصات

# Remerciement

---

## Remerciement

Que seraient nos efforts sans l'aide du tout puissant que nul ne peut s'estimer suffisamment reconnaissant envers lui ; nous le remercions pour le courage qu'il nous a offert toute au long de notre parcours au sien de département d'informatique de l'université ABOU BAKR BELKAID de Tlemcen

Nous remercions avant tout le Bon Dieu de nous avoir donné la volonté de finir ce mémoire

Nous remercions jamais assez la personne qui nous a guidé, conseillé, orienté pas à pas dans notre travail ; notre sincère gratitude envers Mr LEHSEINI qui en tant qu'un encadreur qui nous a apporté écoute et aide avec une patience prodigieuse ; que le tout puissant lui soit d'appuis là où ses pas prendront part dans la vie.

Nous tenons également à exprimer nos sincères reconnaissances à Mme BOUMAHDI de l'université de Blida pour ses conseils, sa disponibilité et l'aide précieuse dont elle nous a fait part.

Nos gratitudes vont à l'ensemble des enseignants de département d'informatique qui nous ont transmis leurs savoirs et leurs connaissances tout au long de nos cinq années, sans oublier d'adresser des remerciements particuliers à Mr BENAMMAR, le chef de département, pour le dynamisme de ce département d'étude.

Nos remerciements vont aux membres du jury qui ont accepté de sacrifier une part de leurs temps pour examiner notre travail.

Enfin, à tous nos proches, familles et amis pour le soutien et leur présence.

## Dédicaces

---

### Dédicaces

*En signe d'amour et de respect éternel je dédie mon modeste travail  
aux êtres les plus chères : Mes parents pour leurs sacrifices, que Dieu les  
protège et puisse les rendre fiers de moi,*

*Ma sœur Hanane et sa petite famille Safi, Safaa et Hadil*

*Mes frères Zouheir, Hichem et Mahmoud*

*A mes très chers grands-parents maternels et paternels, à qui  
j'espère une très longue vie fleurissante au sien de notre famille,*

*Mes oncles et mes tantes un par un*

*A tous mes amis et amies pour le réconfort qu'ils m'ont apporté toute  
au long de ces années et dieux sait que j'on avait besoin en particulier*

*Ismahan, Sihem, Imane, Bouchra et Ahlem.*

*Le comble de ma gratitude, respect, affection vont vers mon grain de  
lumière mon fiancé Hamid et toute sa famille.*

# Table Des Matières

---

## Table Des Matières

<b>RESUME</b> .....	I
<b>ABSTRACT</b> .....	I
<b>ملخص</b> .....	I
<b>Remerciement</b> .....	II
<b>Dédicaces</b> .....	III
<b>Tables Des Matières</b> .....	1
<b>Introduction Générale</b> .....	4
<b>Chapitre I Synthèse sur les différentes architectures</b> .....	6
I.1 Introduction.....	6
I.2 Paradigme informatique.....	6
I.2.1 Paradigme impératif (Procédural) .....	7
I.2.2 Paradigme objet.....	7
I.2.3 Paradigme logique.....	7
I.2.4 Paradigme fonctionnel .....	8
I.2.5 Paradigme composant .....	8
I.3 Programmation orienté objet .....	10
I.3.1 Définition .....	10
I.3.2 Principes de base de POO .....	10
I.3.3 L'évolution du POO .....	11
I.3.4 Les limites du POO .....	12
I.4 Le composant.....	12
I.4.1 Définition .....	12
I.4.2 Propriétés.....	12
I.4.3 Que doit fournir un modèle de composants .....	12
I.4.4 Relations entre composants et objets .....	12
I.4.5 Les limites du composant .....	13
I.5 Conclusion .....	13

## Table Des Matières

---

<b>Chapitre II Architecture Orienté Service (SOA)</b> .....	14
II.1 Introduction.....	14
II.2 Architecture Orientée Services (SOA) .....	14
II.2.1 Définition.....	14
II.3 Pourquoi SOA ?.....	15
II.4 Les Services .....	16
III.4.1 Définition.....	16
III.4.2 Propriétés d'un service .....	17
III.4.3 Les composants techniques d'un service.....	20
III.4.4 Les types de service.....	21
II.5 Les services web .....	22
II.5.1 Définition.....	22
II.5.2 L'architecture orientée service web.....	23
II.5.3 Les standards de l'architecture orientée service web .....	23
II.6 Les grands principes du SOA .....	24
II.6.1 La définition du service .....	24
II.6.2 Autonomie des services .....	24
II.6.3 Partage des contrats .....	25
II.6.4 Compatibilité basée sur les règles .....	25
II.7 Les éléments de base du SOA.....	25
II.7.1 Eléments organisationnels d'une SOA.....	25
II.7.2 Eléments techniques d'une SOA .....	26
II.8 Mécanisme de SOA .....	27
II.9 Méthodologies de conception orientée services .....	28
II.9.1 Praxeme .....	28
II.9.2 SoaML (Services Oriented Architecture Modeling Language).....	29
II.9.3 MDA (Model Driven Architecture).....	31
Conclusion .....	33
<b>Chapitre III Les appels d'offre</b> .....	34
III.1 Introduction.....	34
III.2 Les Appels d'offres.....	34
III.3.1 Définition.....	34
III.3.2 Pourquoi les appels d'offres ? .....	35

## Table Des Matières

---

III.3.3	Le processus d'appel d'offres .....	35
III.3	Types d'Appels d'Offres .....	37
III.3.1	Appel d'offres ouvert (AOO) .....	38
III.3.2	Appel d'offres restreint (AOR) .....	38
III.4	Conclusion .....	40
<b>Chapitre IV Conception et réalisation de l'application .....</b>		<b>41</b>
IV.1	Introduction.....	41
IV.2	BPMN (Business Process Modeling Notation) .....	41
IV.3	Modélisation .....	42
IV.4	Implémentation .....	43
IV.5	Description de l'application.....	46
IV.5.1	Côté serveur.....	46
IV.5.2	Côté Client.....	47
IV.6	Quelques exécutions .....	47
IV.7	Conclusion .....	52
<b>Conclusion Générale.....</b>		<b>53</b>
<b>Références Bibliographiques .....</b>		<b>IV</b>
<b>Liste des Figures.....</b>		<b>VIII</b>

# Introduction Générale

---

## Introduction Générale

Un système d'Information (noté SI) représente l'ensemble des éléments participant à la gestion, au traitement, au transport et à la diffusion de l'information au sein de l'organisation. C'est un composant indispensable pour la mise en œuvre de la stratégie de l'entreprise. Une des techniques de concevoir le SI qui privilégie les processus métiers et garantit l'interopérabilité, la flexibilité et la réutilisabilité, est l'architecture orientée service ou SOA (Service Oriented Architecture).

SOA est un ensemble de services métiers qu'une organisation veut offrir à ses partenaires et clients. Ces services étant décrits d'une façon standardisée permettant l'assemblage de Services. Son objectif est donc de décomposer une fonctionnalité en un ensemble de fonctions basiques, fournies par des composants et de décrire finement le schéma d'interaction entre ces fonctions.

L'élément clé de l'architecture SOA est le service web. En effet, les services web ne sont qu'un moyen de mettre en place une architecture orientée services et d'atteindre ses objectifs. Ils sont considérés comme étant l'évolution naturelle du web. Les services web facilitent non seulement les échanges entre les applications de l'entreprise mais surtout permettent une ouverture vers les autres entreprises.

Dans le cadre de ce PFE on a proposé un système d'information de traitement des appels d'offre en incluant la notion de l'architecture SOA.

## Problématique

La problématique de communication entre les applications hétérogènes au sein des entreprises est un vieux défi de l'informatique. On a essayé de la résoudre en proposant diverses méthodes et technologies. Or l'empilement progressif des applications a conduit à une situation intenable de « silos ».

Les architectures orientées services (SOA) offrent aujourd'hui une formidable opportunité pour résoudre cette problématique, par conséquent elle est de plus en plus demandée et utilisée dans les entreprises, en contrepartie, ces dernières ont besoin des processus métiers bien définis.

# Introduction Générale

---

## Objectif

Objectif de ce mémoire est de pouvoir appliquer l'architecture SOA sur un système d'information de traitement des appels d'offre et mettre à profit les possibilités offertes par les services web en identifiant les processus métiers.

Ce mémoire est organisé comme suit :

- ✓ Le premier chapitre s'intéresse aux différents paradigmes de programmation.
- ✓ Le deuxième chapitre passe en revue le domaine SOA et la technologie des web services.
- ✓ Le troisième chapitre s'intéresse aux concepts généraux de traitement des appels d'offre.
- ✓ Le quatrième chapitre décrit le cas pratique de ce projet à savoir la proposition d'une architecture orienté service appliquée sur le système des appels d'offre, et reprend le volet réalisation de l'application proposée.
- ✓ Et enfin, une conclusion générale et des perspectives pour clôturer ce mémoire.

## Chapitre I

### Synthèse sur les différentes architectures

---

#### I.1 Introduction

Au cours des 35 dernières années, les concepteurs de matériel informatique sont passés de machines de la taille d'un hangar à des ordinateurs portables légers basés sur de minuscules microprocesseurs. Au cours des mêmes années, les développeurs de logiciels sont passés de l'écriture de programmes en assembleur et en COBOL à l'écriture de programmes encore plus grands en C et C++. On pourra parler de progrès (bien que cela soit discutable), mais il est clair que le monde du logiciel ne progresse pas aussi vite que celui du matériel [1].

A travers ce chapitre nous allons répondre aux questions suivantes:

- Quels sont les différents paradigmes informatiques ?
- On veut dire quoi par l'orienté objet ?
- Quelles sont les évolutions qu'a connues l'orienté objet par rapport à la procédure ?
- Quelles sont les limites de l'orienté objet ?
- Un composant : qu'est-ce que c'est ?

#### I.2 Paradigme informatique

Le paradigme informatique est un modèle de description d'un programme. Chaque paradigme de programmation privilégie un ensemble particulier de stratégies d'analyse et de descriptions. Chacun impose une approche, un point de vue particulier sur tout problème. Certains types de problèmes se traitent plus facilement selon un certain paradigme. Par exemple, les problèmes de simulation gagnent à être abordés d'un point de vue orienté objet [2].

Examinons brièvement quelques-uns des plus importants paradigmes de programmation dans ce qu'ils sont et selon les stratégies de résolution de problèmes qu'ils supportent.

## I.2.1 Paradigme impératif (Procédural)

C'est une séquence d'instructions qui modifie l'état de la machine par effet de bord et chaque instruction représente un cas particulier de l'affectation. Il est caractérisé par :

- affectation
- l'importance de l'ordre
- l'effet de bord
- la mise à jour directe de l'état de la machine

Les langages de programmation qui utilisent le paradigme impératif sont : C, Pascal, etc.....

## I.2.2 Paradigme objet

Un programme est un ensemble d'objets en interaction. Les programmes orientés objets sont éminemment réutilisables et modifiables. Ce mode de programmation est un levier intellectuel très puissant. Il offre, via l'objet, un excellent outil de modularité [2]. Il est caractérisé par :

- les classes
- l'héritage
- polymorphisme
- encapsulation
- l'envoi de message

Les langages de programmation qui utilisent ce paradigme sont : C++, Java, etc.....

## I.2.3 Paradigme logique

Un programme est un ensemble des clauses logiques (axiomes). Un tel programme est "exécuté" lorsqu'on pose une requête visant à prouver qu'un énoncé peut être déduit à partir des axiomes. L'objet de base est le prédicat (par exemple couleur : rose, rouge) qui décrit une relation entre un certain nombre d'individus [2].

Il est caractérisé par :

## Chapitre I : Synthèse sur les différentes architectures

---

- les clauses
- les littéraux
- l'unification
- les règles de résolution et de facturation
- la substitution

Les langages de programmation qui utilisent ce paradigme sont: prolog, etc.....

### I.2.4 Paradigme fonctionnel

Un programme est un emboîtement de fonctions pures. Une fonction, dans son incarnation informatique, accepte des données et produit données. Elle peut être soit "primitive" soit formée par une composition de fonctions. Il n'y a pas de séparation entre données et programmes [2]. Il est caractérisé par:

- pas d'affectation
- pas d'effet de bord
- uniquement des fonctions pures
- les variables sont affectées une fois au plus (constantes) pendant la déclaration
- pas de boucles, elles sont remplacées par la récursivité

Les langages de programmation qui utilisent ce paradigme sont : LISP, LOGO et APL, etc.....

### I.2.5 Paradigme composant

Un programme est un ensemble de composants en interaction. Un composant est un code logiciel identifié par une interface. L'interface définit les fonctions offertes et requises. Il est caractérisé par:

- facilité de mise à jour
- choix des langages de développement
- productivité

Les langages de programmation qui utilisent ce paradigme sont : EJB, CCM, etc.....

Il y en a d'autres paradigmes, on les représente par la figure I.1 et la figure I.2 suivante représente l'historique des différents paradigmes.

# Chapitre I : Synthèse sur les différents architectures

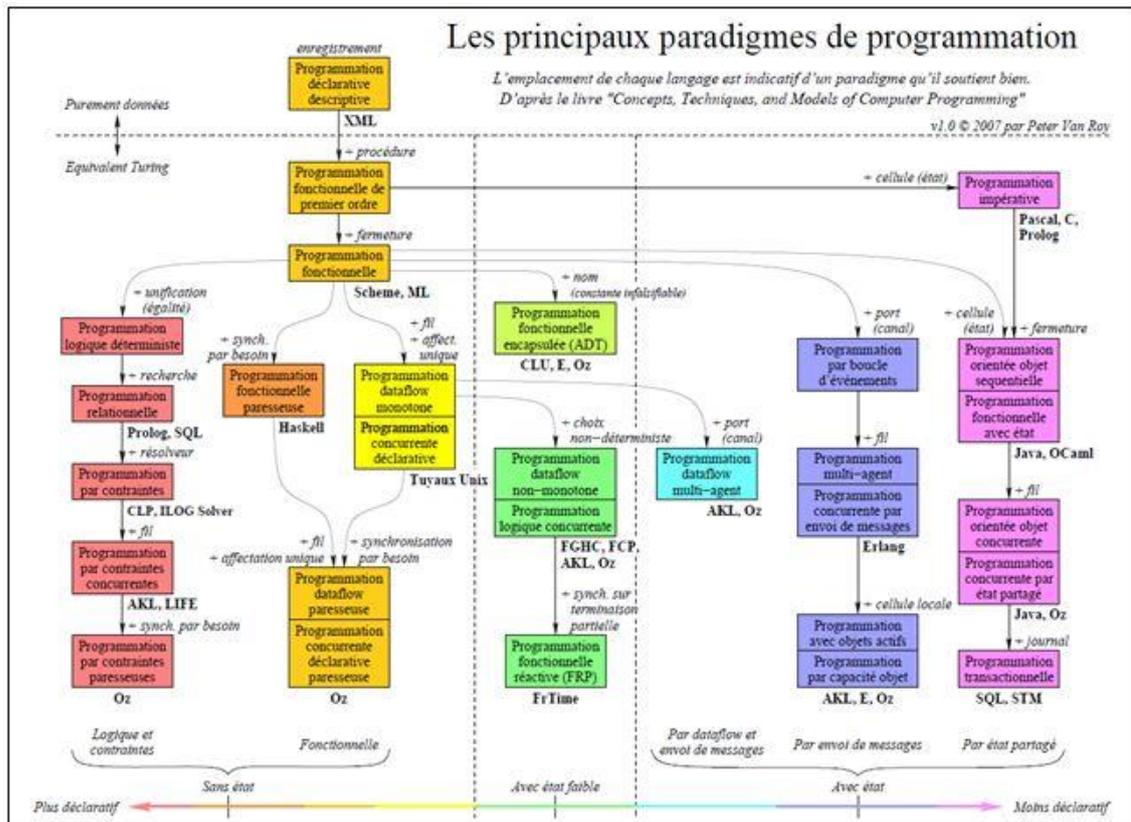


Figure I.1 Principaux paradigmes de programmation [6]

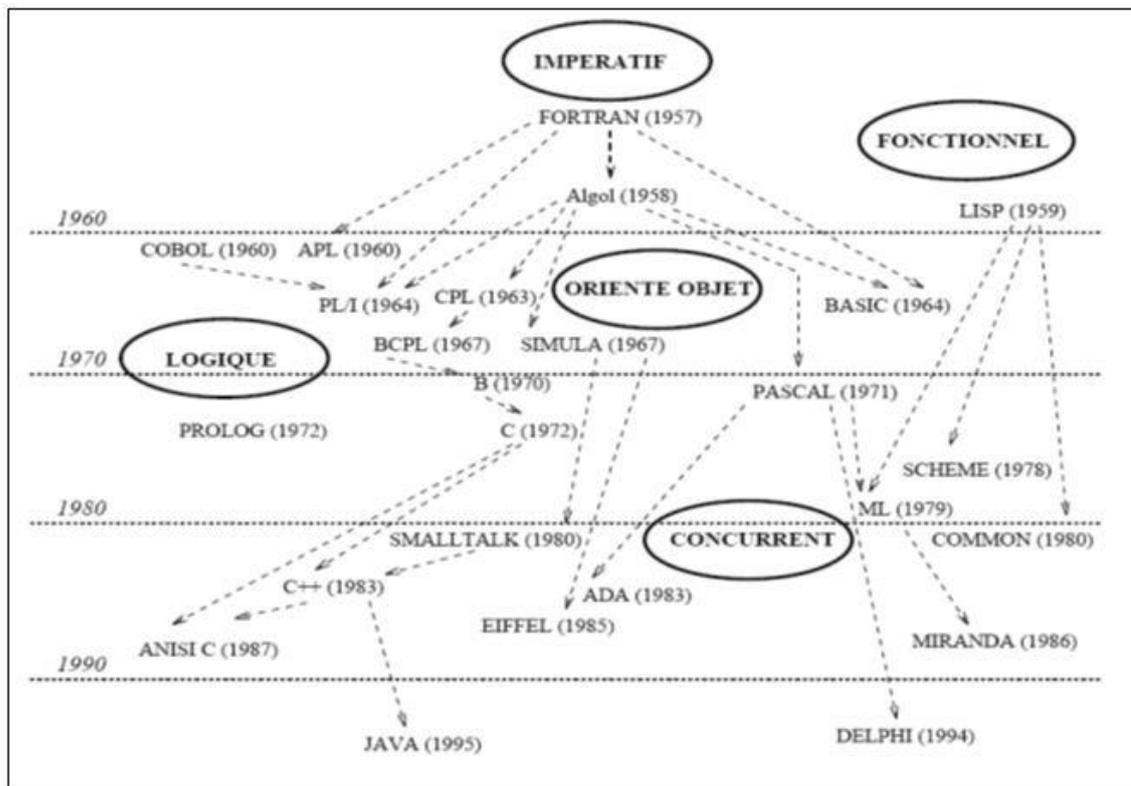


Figure I.2 Historique des différents paradigmes [6]

## I.3 Programmation orienté objet

### I.3.1 Définition

La programmation orientée objet (POO) est un type de programmation qui a pour avantage de posséder une meilleure organisation, surtout dans les gros programmes. Ces derniers seront agencés de façon plus logique et seront donc plus facilement modifiables. La POO est cependant plus difficile à maîtriser.

Un programme orienté objet est un ensemble d'objets autonomes et responsables qui s'entraident pour résoudre un problème final en s'envoyant des messages [2]. La figure I.3 illustre l'interaction entre objets.

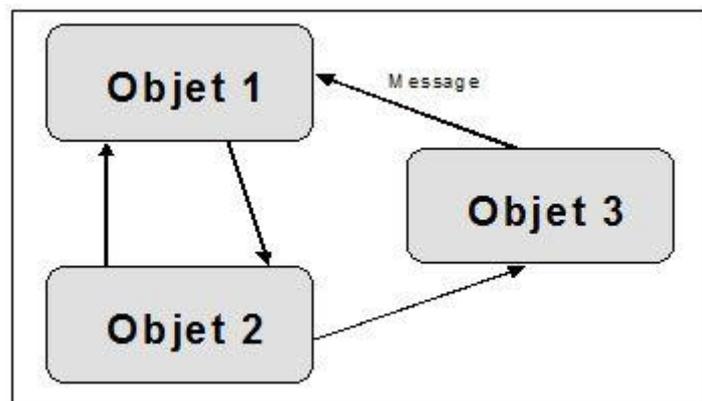


Figure I.3 Interaction d'objets [3]

### I.3.2 Principes de base de POO

#### a. Encapsulation des données [3]

- **L'accès aux données des objets est réglementé**
  - ✓ Données privées → accès uniquement par les fonctions membres.
  - ✓ Données publiques → accès direct par l'instance de l'objet.
- **Conséquences**
  - ✓ Un objet n'est vu que par ses spécifications.
  - ✓ Une modification interne est sans effet pour le fonctionnement général du programme.
  - ✓ Meilleure réutilisation de l'objet.

# Chapitre I : Synthèse sur les différentes architectures

## b. L'Héritage [3]

- Permet de définir les bases d'un nouvel objet à partir d'un objet existant
- Le nouvel objet hérite des propriétés de l'ancêtre et peut recevoir de nouvelles.

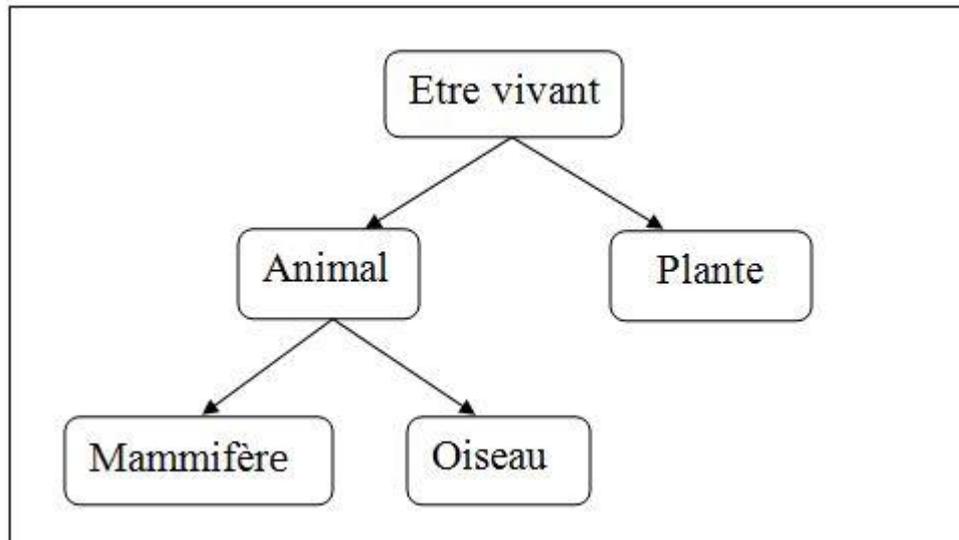


Figure I.4 Exemple d'héritage

## c. Polymorphisme

Un nom (de fonction, d'opérateur) peut être associé à plusieurs utilisations mais à condition qu'elles soient différentes [3].

### I.3.3 L'évolution du POO [4]

- Tout est objet! : chaque objet encapsule des données et des méthodes agissant sur ces données.
- L'encapsulation réalise une abstraction des données : vu de l'extérieur de l'objet, les détails d'implémentation sont cachés.
- Le concept de classe généralise la notion de type : tous les objets d'une certaine classe sont des instances de cette classe.
- Des classes peuvent hériter d'autres classes (classe mère - classes filles). La notion d'héritage permet d'établir une hiérarchie entre les classes.
- Avec l'héritage, il devient possible de redéfinir des méthodes au sein des classes filles. On parle de polymorphisme.

## I.3.4 Les limites du POO [5]

- Pas de vue globale de l'architecture d'une application
- Interactions entre objets enfouies dans le code
- Évolution / modification difficile
- Peu d'outils pour le déploiement et l'administration
- Gestion de la consistance d'un changement délicate.

## I.4 Le composant

### I.4.1 Définition

Un composant est une unité binaire indépendante de production, d'acquisition et de déploiement qui interagit via un Framework. Les composants sont une vision qui met en avant l'opération de composition (assembler des composants pour réaliser un logiciel plus complexe) [7].

### I.4.2 Propriétés

Un composant est défini par ses propriétés [7] :

- c'est une unité de déploiement indépendant.
- c'est une unité de composition par une tierce entité.
- un composant n'a pas d'état persistant.

### I.4.3 Que doit fournir un modèle de composants [8]?

#### a. Encapsulation

- Interfaces = seule voie d'accès, séparation interface-réalisation.
- Possibilité de définir des interfaces multiples.

#### b. Composabilité

- Dépendances explicites (expression des ressources fournies et requises).
- Composition hiérarchique (un assemblage de composants est un composant).

### I.4.4 Relations entre composants et objets [7]

- Un composant peut être constitué d'un ensemble d'objets.
- Un composant peut ne contenir aucun objet.
- Un objet ne peut pas contenir de composant.

# Chapitre I : Synthèse sur les différentes architectures

---

## I.4.5 Les limites du composant [7]

- Les composants abordent des problèmes non pris en compte jusqu'à présent.
- Ils présentent une vision particulière de comment développer des logiciels.
- Ils ne disent pas comment programmer : l'approche orientée objet est complémentaire.
- Analogie avec l'approche orientée objet et la programmation structurée.

## I.5 Conclusion

Chacun de ces paradigmes cités a bien marqué sa place dans l'histoire de l'informatique, bien évident, chacun a ses avantages et ses inconvénients. Par ailleurs, la généralisation de l'outil informatique dans les directions métiers fait que les utilisateurs attendent toujours plus du SI (Système d'Information), soit au niveau d'agilité, soit au niveau de l'intégration des acteurs externes de leur SI. Il devient donc vital de trouver des moyens de mieux gérer et homogénéiser le système d'information.

Dans le chapitre qui suit, on présente une approche qui a essayé d'engendrer les limites des différents paradigmes présentés dans ce chapitre.

# Chapitre II

## Architecture Orientée Services

---

### II.1 Introduction

Au tout début, il n'existait que les langages purement procéduraux, la seule façon d'écrire du code réutilisable était d'écrire des fonctions et des procédures dans un fichier séparé du corps du programme, et de faire appel à ce fichier chaque fois que nécessaire. Ensuite, est apparue la Programmation Orientée Objet (POO). Elle était innovante dans le sens où le concept même d' « objet » permet l'encapsulation et donc de masquer la complexité des opérations. Mais on s'est souvent heurté à des problèmes de compatibilité entre plateformes, d'où le besoin d'une standardisation et la mise en commun des protocoles (SOAP, XML, ...). De là est née la notion d'architecture orientée services (SOA).

SOA s'est imposée aujourd'hui comme un thème majeur pour les systèmes d'information d'entreprise. Plus qu'une nouvelle technologie ou méthode, c'est la convergence de plusieurs approches existantes. En effet, SOA vise à construire le SI d'une entreprise en se basant sur un ensemble de services, utilisés et offerts par des applications. Cette architecture permet également de créer toute une nouvelle génération d'applications dynamiques sous forme de services réutilisables.

Dans ce chapitre on veut présenter la définition de l'architecture orienté service et les différents concepts liées à elle, ainsi que les éléments de base qui la constituent, et finalement on détaille un peu le concept de service.

### II.2 Architecture Orientée Services (SOA)

#### II.2.1 Définition

Il n'y a pas une définition exacte de l'architecture orienté service. En effet plusieurs définitions ont été proposées, mais toutes les définitions sont d'accord que SOA est un paradigme destiné à résoudre les problèmes d'hétérogénéité et interopérabilité des logiciels qui constituent le système d'information.

## Chapitre II Architecture Orientée Services

---

SOA est un paradigme permettant d'organiser et d'utiliser des savoir-faire distribués pouvant être de domaines variés. Cela fournit un moyen uniforme d'offrir, de découvrir, d'interagir et d'utiliser des savoir-faire pour produire le résultat désiré avec des pré-conditions et des buts mesurables [9].

« L'architecture orientée service constitue un style d'architecture basée sur le principe de séparation de l'activité métier en une série de services. »

« Ces services peuvent être assemblés et liés entre eux selon le principe de couplage lâche pour exécuter l'application désirée. »

« Ces services sont définis à un niveau supérieur de la traditionnelle approche composants » [11]

« Architecture orientée service (SOA):est un paradigme d'organisation des ressources distribuées, potentiellement contrôlées par des domaines différents. »[12]

L'architecture SOA peut être utilisée dans un cadre interne à l'entreprise, ainsi les services peuvent être classés par domaine fonctionnel (achat, finance, marketing, etc.), mais surtout SOA présente un moyen efficace pour faciliter la communication avec d'autres entreprises [13].

### II.3 Pourquoi SOA ?

Au début les départements informatiques ont tenu un point de vue orienté application du monde. Ils ont alloué la plupart de leurs budgets à l'achat, le déploiement et la maintenance des applications. A l'époque, un grand pourcentage de budget a été alloué aux projets d'intégration pour fournir un soutien plus large et plus transparent pour les processus d'affaires. En effet, ces projets n'ont souvent pas atteint leurs promesses en termes de la rigidité de l'intégration câblés. Tout changement à une seule application rend le retour en arrière de l'infrastructure très coûteux.

## Chapitre II Architecture Orientée Services

Après que les techniques conventionnelles de programmation ont été remplacées par le développement de modèle à objet, la programmation par composant est apparue et a permis un développement des applications basé sur cette technique de programmation, un nouveau concept architectural est apparu : l'architecture orienté service ou SOA (Service Oriented Architecture) [14].

En quelques années, SOA est devenue un thème majeur pour les systèmes d'informations d'entreprise. Plus qu'une nouvelle technologie ou méthode, c'est la convergence de plusieurs approches existantes, et l'émergence d'une forte adhésion des directions informatique et métier à un même objectif : une meilleure agilité des systèmes face aux transformations nécessaires. La figure II.1 représente une architecture basique du SOA.

Cette architecture est apparue pour :

- Gérer l'intégration entre des applications dans le SI.
- Construire un SI plus flexible.
- Faciliter l'innovation.
- Réduire les coûts.
- Réagir rapidement à tous les changements.
- Réduire les doublons entre applications.

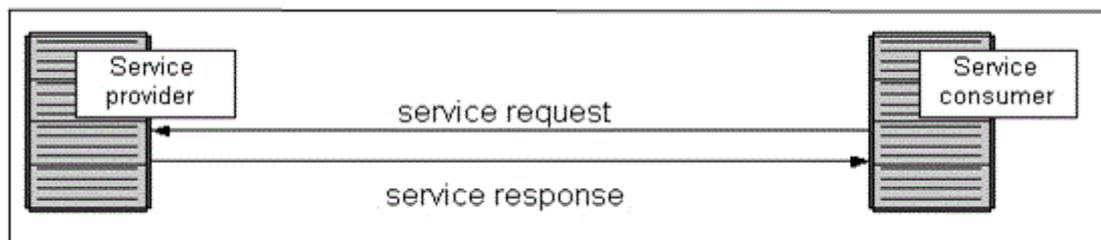


Figure II.1 Architecture basique de SOA [14]

### II.4 Les Services

#### III.4.1 Définition

SOA présente un modèle d'architecture informatique basée sur l'émergence d'une couche de **services**. Ces services offrent une vue logique des traitements et données existant déjà ou à développer, et chaque service encapsule ces traitements et données et

## Chapitre II Architecture Orientée Services

masque ainsi l'hétérogénéité du système d'information. Plusieurs définitions ont été proposées, parmi elles:

Un service est un comportement défini par contrat, qui peut être réalisé et fourni par tout composant pour être utilisé par tout composant, sur la base unique du contrat [15].

### III.4.2 Propriétés d'un service

Un service doit posséder les caractéristiques suivantes:

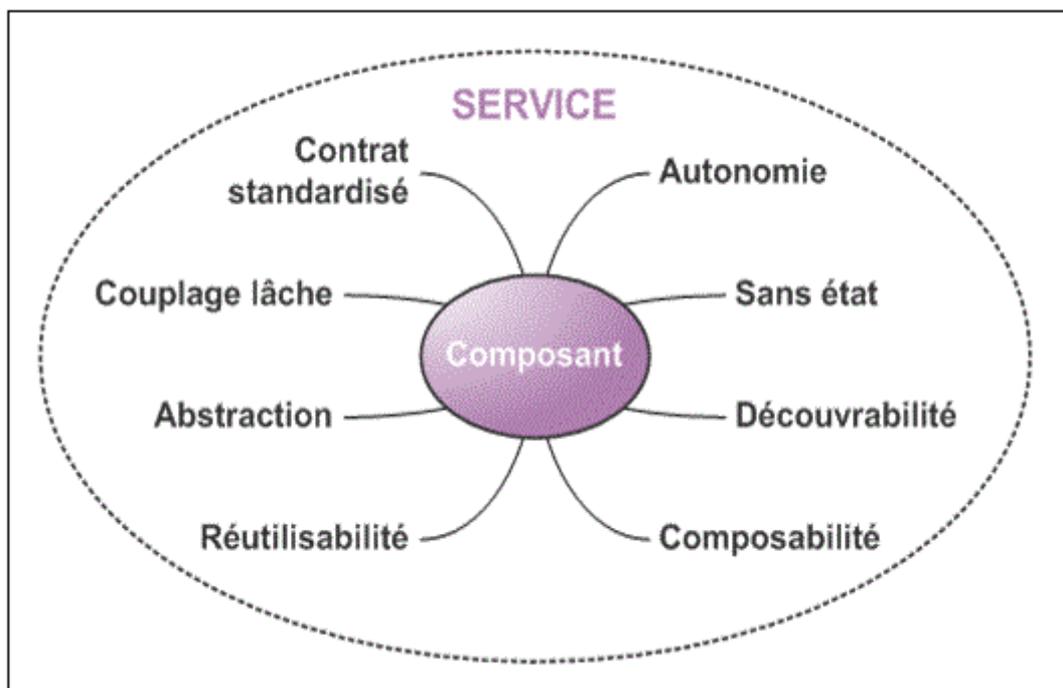


Figure II.2 Les propriétés d'un service

#### a. L'autonomie

Un service doit être autonome et indépendant dans le contexte de l'architecture SOA.

#### b. Granularité

Un enjeu de la SOA est de trouver la bonne granularité des services proposés par une application. En effet, un service à granularité trop fine n'offre que peu d'intérêt au niveau métier (des services renvoyant uniquement le nom d'un client, ou juste son adresse n'ont guère de valeur ajoutée.)

## Chapitre II Architecture Orientée Services

Des services à granularité plus forte, créés à partir de plusieurs composants structurés, offrent plus d'intérêt dans la réalisation d'un processus global [17].

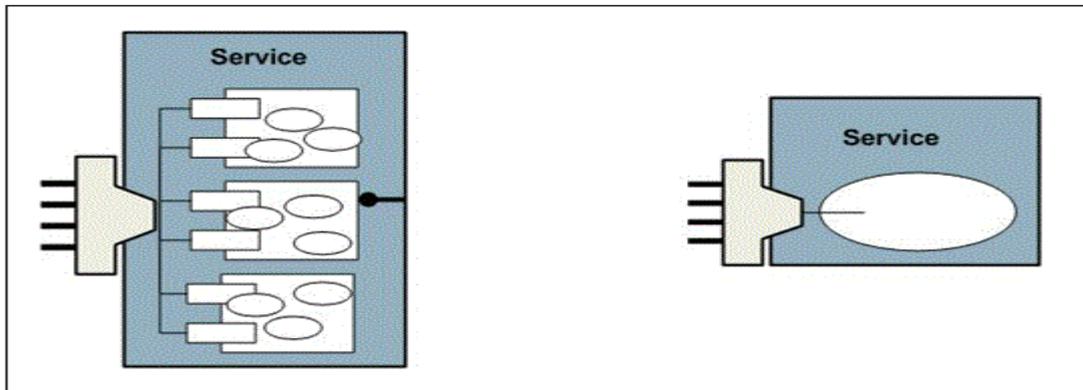


Figure II.3 Service à forte granularité versus service à faible granularité [17].

### c. Couplage lâche

En théorie, un service ne peut pas appeler un autre service. Il délègue cette fonction à un traitement spécialisé dans l'enchaînement (fonction d'orchestration) [18].

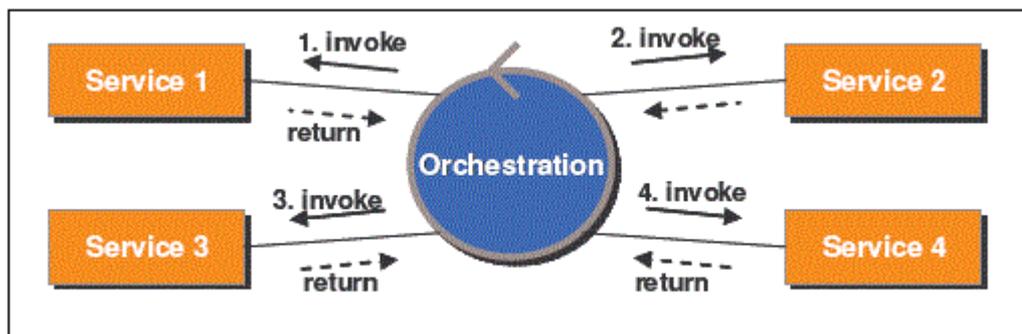


Figure II.4 Le principe d'orchestration des services [18].

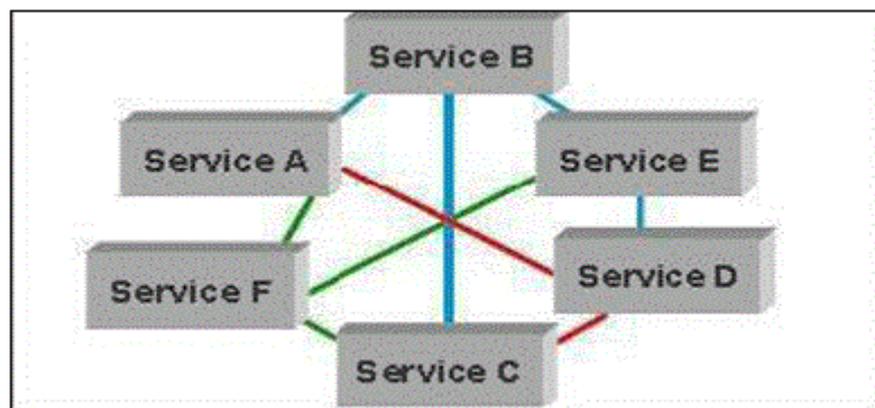


Figure II.5 organisation des services n'est pas en couplage faible [19].

## Chapitre II Architecture Orientée Services



Figure II.6 organisation des services est en couplage faible [19]

### d. Interopérabilité

Elle est considérée comme propriété fondamentale, dans le sens où l'appel au service fonctionne quel que soit le langage de programmation et le système d'exploitation.

### e. Réutilisabilité

Les services sont conçus de façon à ce qu'ils puissent être réutilisés ultérieurement. Le but est de minimiser la redondance et rendre le service réutilisable par les différentes applications du système d'information.

### f. Composabilité

Les services peuvent invoquer ou utiliser d'autres services, le principe est la décomposition des processus métiers en un certain nombre d'unités d'activités sous forme de service. La composition de ces derniers mène à l'exécution des différentes tâches du processus métiers.

### g. La qualité de service

L'interaction entre un fournisseur et un consommateur de services se fait à travers une interface selon un contrat d'utilisation, évidemment on échange des données de nature métier.

### h. Découvrabilité

Pour invoquer un service on doit connaître que le service existe, donc il y a un domaine publique là où on peut trouver tous les services ainsi que la description de chaque service.

### III.4.3 Les composants techniques d'un service

Les éléments techniques qui composent un service sont:

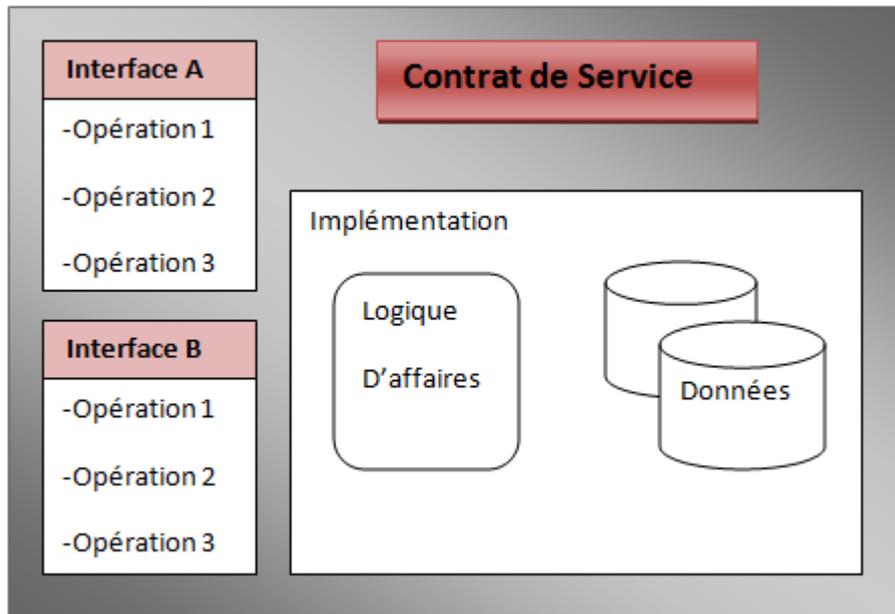


Figure II.7 Les composants techniques d'un service. [20]

#### a. L'interface

La fonctionnalité du service est exposée par l'interface aux clients qui sont reliés au service en utilisant un réseau. Bien que la description de l'interface fasse partie du contrat de service, elle encapsule l'implémentation et l'exécution physique de service.

#### b. Le contrat

Le contrat de service joue un rôle majeur: il détaille les conditions d'utilisation du service sous forme de pré et post conditions, protocoles, et contraintes (QoS, SLA5). Un contrat de service est un document organisé en plusieurs parties qui sont [21] :

- L'identification des parties
- La description des fonctions du service.
- La description de l'interface du service.
- La description de la qualité du service.
- La description du cycle de vie du service et du contrat.
- La description des termes de l'échange (s'il y'en a).

## Chapitre II Architecture Orientée Services

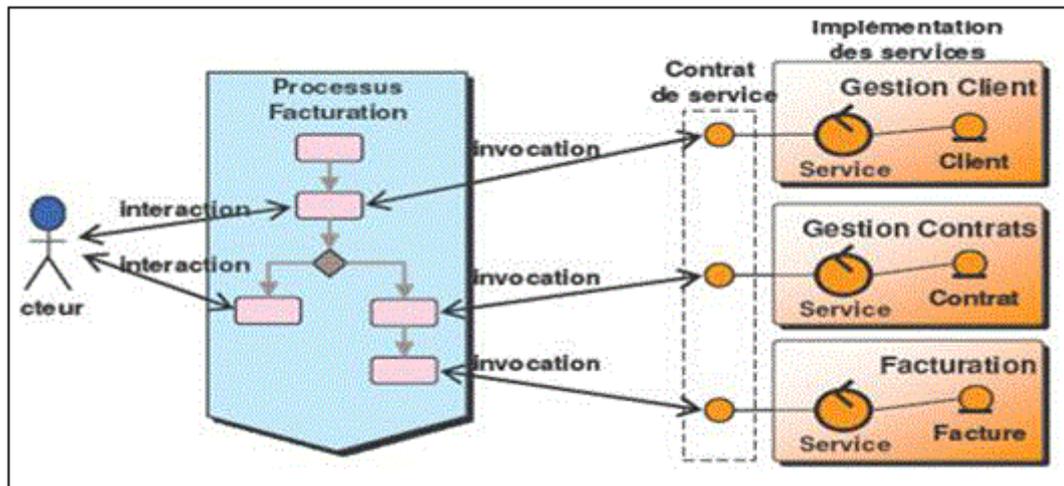


Figure II.8 le contrat de service [18].

### c. L'implémentation

L'implémentation de service fournit physiquement la logique d'affaire exigée et les données appropriées. C'est la réalisation technique qui accomplit le contrat de service.

### d. Données

Un service peut également inclure des données.

### e. La logique métier

Il se compose des règles de service métier, les exécutions et les résultats d'un service dans la réalité.

## III.4.4 Les types de service

Il y a deux grands groupes de services [22] :

### a. Les services métiers

Il offre un service de type particulier et il correspond à un périmètre fonctionnel que l'on souhaite exposer à des consommateurs indépendamment des choix d'architecture applicative.

Les services métiers sont divisés en trois principaux groupes :

- **service de persistance** : qui a le rôle de créer, rechercher, mettre à jour et supprimer de l'information des référentiels du SI,

## Chapitre II Architecture Orientée Services

- **service fonctionnel** : sert à encapsuler tout ou partie des règles de gestion et des traitements métier. Il s'appuie en général sur des services de persistance pour accéder aux informations métier dont il a besoin. Le service fonctionnel est caractérisé par une granularité moyenne à forte et une réutilisabilité moyenne à forte.
- **service applicatif** : Sert à coordonner les appels aux services fonctionnels. Sa granularité est forte et sa réutilisabilité est nulle.

### b. Les services techniques

Ils donnent l'accès à une ressource technique donnée (Messagerie, imprimante, Base de données relationnelle... etc.). Les services techniques sont génériques, c'est-à-dire lié à une catégorie de ressources. Ils sont caractérisés par une granularité moyenne à forte et une réutilisabilité forte.

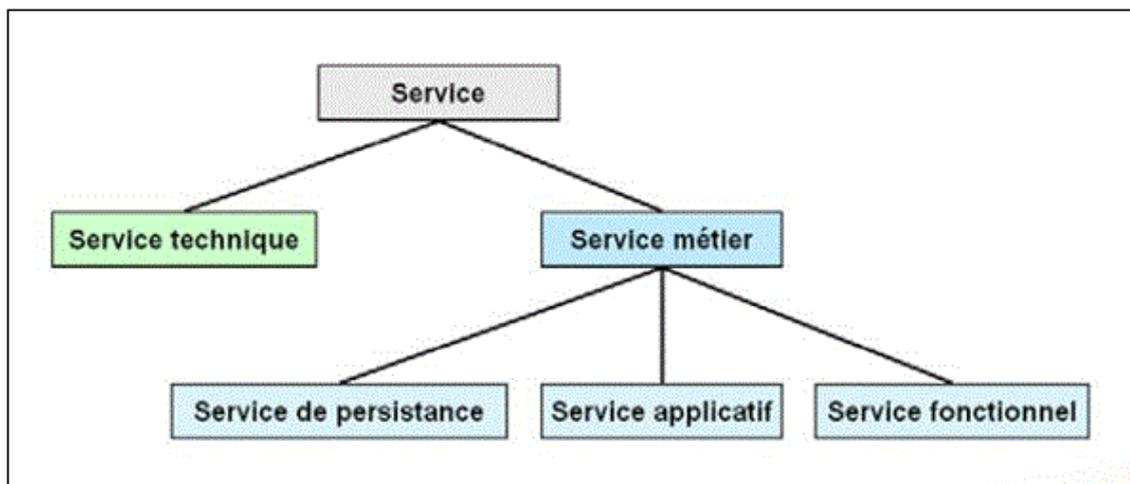


Figure II.9 Les types de services [22].

## II.5 Les services web

### II.5.1 Définition

Un service web est un mécanisme qui tend à donner plus d'interactions pour permettre à deux entités hétérogènes (entreprises, clients, applications, etc. ...) de dialoguer au travers du réseau Internet. Les logiciels écrits dans divers langages de programmation (C#, Visual Basic, Java, etc....), sur diverses plateformes (Linux, Windows, etc. ...) et avec diverses architectures peuvent employer des services Web

## Chapitre II Architecture Orientée Services

pour échanger des données à travers des réseaux informatiques. Chaque service web doit pouvoir être découvert et invoqué dynamiquement par les applications [23].

### II.5.2 L'architecture orientée service web

Lorsque l'architecture SOA s'appuie sur des services web, on parle alors de WSOA. WSOA s'est concrétisée autour de trois spécifications considérées comme des standards, à savoir SOAP, UDDI et WSDL.

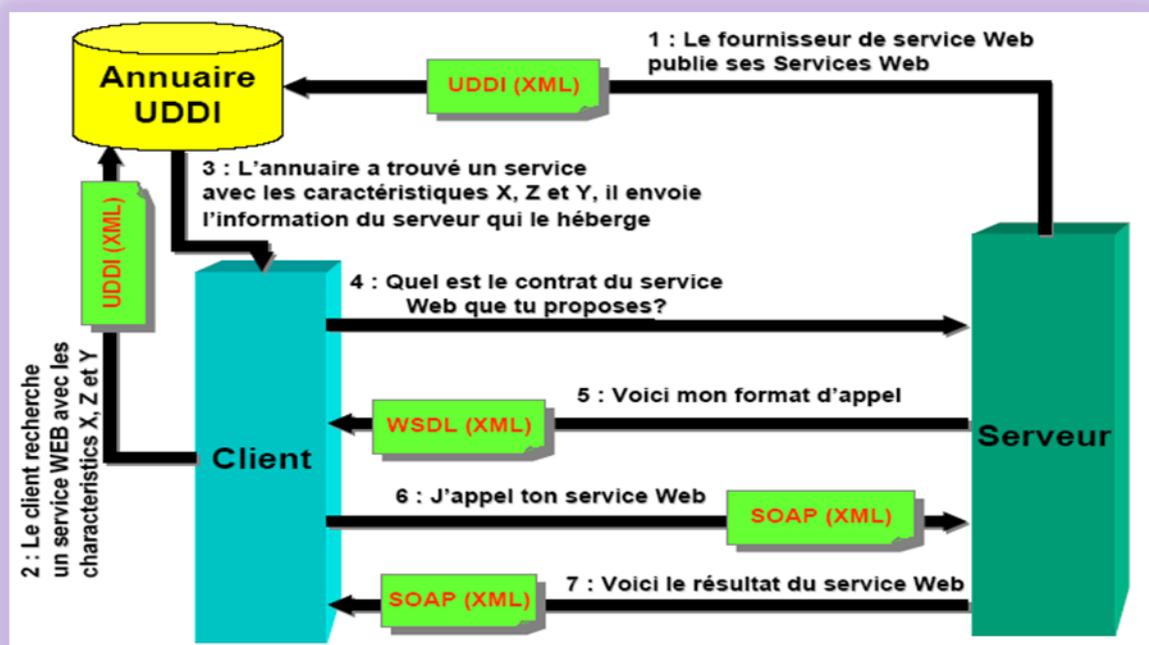


Figure II.10 L'architecture orientée service web [24].

### II.5.3 Les standards de l'architecture orientée service web

#### a. Transport : SOAP (Simple Object Protocol communication)

SOAP est un protocole de transmission de messages fondé sur XML pour l'échange d'informations entre systèmes hétérogènes en environnement décentralisé.

Il définit un ensemble de règles pour structurer des messages principalement pour exécuter des dialogues requête /réponse [25].

## Chapitre II Architecture Orientée Services

---

### b. Découverte : UDDI (Universal Description Discovery and Integration)

C'est un annuaire de publication de services, il se repose sur le protocole SOAP et de ce fait, les requêtes et les réponses sur l'annuaire sont des messages SOAP.

UDDI comporte plusieurs catégories de données :

- Pages blanches : elles recensent les entreprises et contiennent des informations telles que le nom de l'entreprise, ses coordonnées, des descriptions accessibles aux clients, et des identifiants permettant de la retrouver par recherche.
- Pages jaunes : elles contiennent, au format WSDL, la description des services déployés par l'entreprise. Les services sont répertoriés par catégories.
- Pages bleues : elles fournissent des informations techniques détaillées sur les services [25].

### c. Description : WSDL (Web Services Description Language)

C'est une interface qui cache le détail de l'implémentation du service, permettant une utilisation indépendante de la plate-forme et de langage utilisés. Il regroupe au format XML toutes les informations nécessaires pour interagir avec le web service. Il y a deux types de documents WSDL [25] :

- Le document WSDL décrivant l'interface du service.
- Le document WSDL décrivant l'implémentation du service.

## II.6 Les grands principes du SOA

SOA repose sur les quatre principes suivants [24]:

### II.6.1 La définition du service

Un service est une classe exposée à travers les réseaux. Le but de ce service est de fournir une prestation bien définie (exemple: fournir la date et l'heure dans un pays donné). Les détails d'un service sont contenus dans un document standardisé, qui fait office de contrat entre le client et le serveur.

### II.6.2 Autonomie des services

Un service se doit être totalement autonome. On doit pouvoir le remplacer ou le déplacer sans que cela affecte d'autres services. Un service implémente ses propres

## Chapitre II Architecture Orientée Services

---

composants et ses propres méthodes d'accès aux données, il ne doit dépendre d'aucun élément externe.

### II.6.3 Partage des contrats

On a vu dans le premier principe du SOA, que les services exposent des contrats pour exposer aux clients leurs fonctionnalités et comment les utiliser. Cette interface est la seule chose que le serveur partage avec le client. Comme en programmation orientée objet, le client n'a pas à connaître comment procède le service pour arriver à ses fins. En aucun cas, le service et le client ne doivent partager du code.

### II.6.4 Compatibilité basée sur les règles

Chaque service annonce ses possibilités et exigences sous la forme d'assertions intégrables par un moteur de stratégies. La stratégie indique quelles exigences et garanties (appelées assertions) doivent être vérifiées pour permettre le fonctionnement normal du service.

## II.7 Les éléments de base du SOA

Ils doivent être compris de deux niveaux, technique et métier (organisationnel).

### II.7.1 Éléments organisationnels d'une SOA

Selon F.Vernadat, [13], les éléments métiers du SOA sont définis comme suit :

#### a. Service métier

C'est une fonctionnalité de l'entreprise qui apparaît atomique du point de vue de l'appelleur de service. Un service doit être identifié d'une manière unique dans l'entreprise.

#### b. Évènement métier

C'est un fait qui se produit en relation avec les opérations de l'entreprise. C'est un changement dans l'état de l'entreprise, qui doit avoir une réponse. Il existe plusieurs types d'évènements : un évènement sollicité (réception d'une nouvelle requête, etc.), un évènement d'exception (panne de machine, etc.), évènement programmé (délai de temps passé, etc.) ou un évènement de synchronisation (début ou fin d'une activité)

## Chapitre II Architecture Orientée Services

---

### c. Processus métier

Un processus métier est un séquençement partiellement ordonné d'un ensemble d'activités et/ou de services dans le but de réaliser un objectif de l'entreprise.

### II.7.2 Eléments techniques d'une SOA

SOA est une architecture de logiciel qui est composée par des facteurs principaux : des applications qui utilisent les services, les services, les annuaires, et l'infrastructure de communication pour relier tous [20].

#### a. Les services

Vue dans la section II.4.

#### b. Les annuaires des services

Un annuaire de service fournit des informations concernant l'utilisation des services. Ces derniers doivent être découverts en dehors de la portée fonctionnelle et temporelle du projet qui les a créés. Bien qu'une grande partie de l'information soit déjà une partie du contrat de service, l'annuaire de service peut fournir des informations additionnelles telles que l'endroit physique, les informations sur le fournisseur, les personnes de contact, les honoraires d'utilisation, les contraintes techniques, les titres, et les niveaux de service disponibles.

#### c. Infrastructure de SOA (Service Bus)

Une infrastructure de service relie tous les participants d'un SOA. Si deux participants ont besoin d'une communication, par exemple, si une application doit appeler une certaine fonction d'un service. À cet égard, le service bus est semblable au concept d'un bus de logiciel. Cependant, d'une manière primordiale, le service bus ne se compose pas nécessairement d'une technologie simple, mais comporte plutôt une variété des produits et des concepts [20].

Les caractéristiques de Service Bus sont [20]:

##### ➤ **Connectivité**

Le but primaire du service bus est de relier l'ensemble des participants d'une SOA en leurs fournissant une communication standard.

##### ➤ **Hétérogénéité de technologie**

## Chapitre II Architecture Orientée Services

La réalité des entreprises est caractérisée par des technologies hétérogènes. En conséquence, le service bus doit pouvoir relier les participants qui sont basés sur différents langages de programmation, logiciels d'exploitation, ou environnements.

➤ **Hétérogénéité des concepts de communication**

En raison des conditions divergentes de différentes applications, le service bus doit permettre différents modes de communication.

➤ **Les caractéristiques techniques**

Bien que le but du service bus soit principalement communication, il doit également fournir des capacités techniques telles que la sécurité, la transformation de message, ou les transactions...

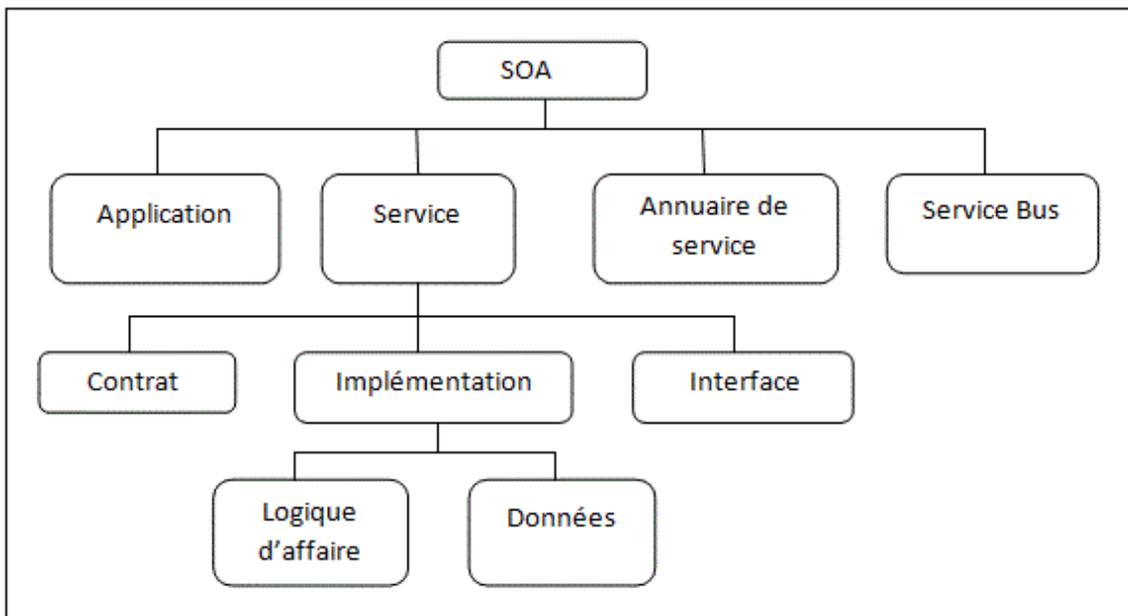


Figure II.11 les éléments techniques du SOA [20].

### II.8 Mécanisme de SOA

Les principes de base qui soutiennent SOA sont illustrés dans le schéma suivant : le fournisseur d'un service décrit sa sémantique, c.-à-d., les informations des fonctions qu'il fournit, leurs caractéristiques opérationnelles ainsi que les conditions d'accès et d'utilisation. Ces informations descriptives (ou métadonnées) sont publiées dans l'annuaire des services. Quand un consommateur veut utiliser un service, il doit, d'abord, se connecter à l'annuaire pour trouver les services qu'ils l'intéressent. Basé sur

## Chapitre II Architecture Orientée Services

les informations que l'annuaire fournit, le demandeur accède finalement aux services choisis et les utilise.

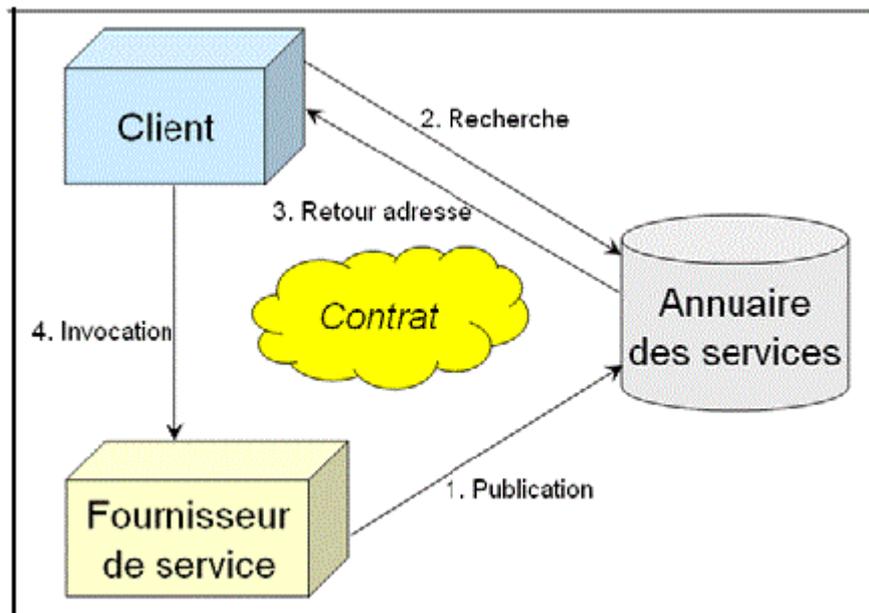


Figure II.12 Le mécanisme de SOA [22].

### II.9 Méthodologies de conception orientée services

Les nouveaux standards (UML, MDA) ne répondent pas complètement aux questions qui se posent sur le terrain pour la modélisation des systèmes complexes : Que faut-il représenter ? Qu'est-ce qu'un bon modèle ? Comment trouver les services ? etc. La réponse à ces questions est de recenser les différentes perspectives de la réalité sur laquelle nous voulons réagir. Dans cette optique, deux méthodologies ont été proposées dont l'objectif est de proposer des procédés de modélisation pour toutes les facettes de l'entreprise. Les méthodologies en question sont Praxeme et Soaml.

#### II.9.1 Praxeme

C'est une méthode produite par le « Praxeme Institute » (association française soumise à la loi de 1901). Elle base sa méthode sur une partition d'aspects selon les questions auxquelles on répond et les compétences spécifiques [26].

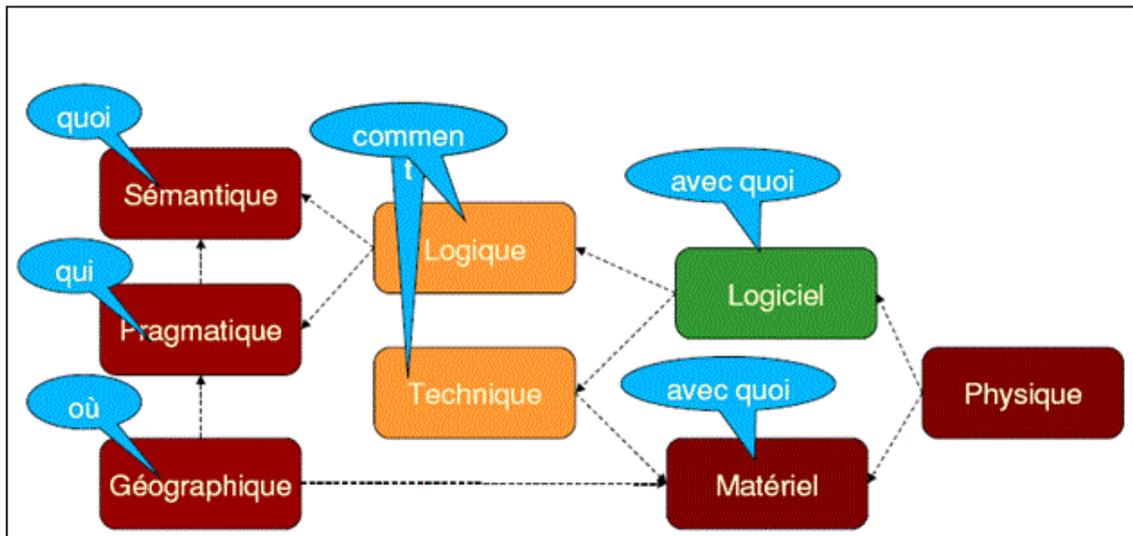


Figure II.13 Les aspects de la méthodologie Praxeme [26].

### II.9.2 SoaML (Services Oriented Architecture Modeling Language)

C'est une spécification de l'Object Management Group (OMG), qui fournit un Métamodèle pour la conception de services au sein d'une architecture orientée services [27].

Soaml se base sur un ensemble de composants qui s'adaptent au niveau métier de l'entreprise:

- **Les participants:** sont utilisés pour définir les fournisseurs et les consommateurs de services dans un système, ils peuvent jouer le rôle d'un fournisseur de service, d'un consommateur de service ou les deux.
- **L'architecture de services:** est utilisée pour définir la manière dont les participants s'échangent les services.
- **Les contrats de services :** définis par des interfaces les contrats sont utilisés pour décrire les modèles d'interaction entre les participants. Un contrat est utilisé pour modéliser un accord entre deux ou plusieurs parties.

Pour modéliser l'architecture orientée service, la méthodologie SOMAL s'appuie sur quatre principaux modèles illustrés dans le schéma ci-dessus, le premier et le dernier modèle CIM et PSM font partie des modèles de l'approche MDA ils seront plus détaillés dans la section suivante :

## Chapitre II Architecture Orientée Services

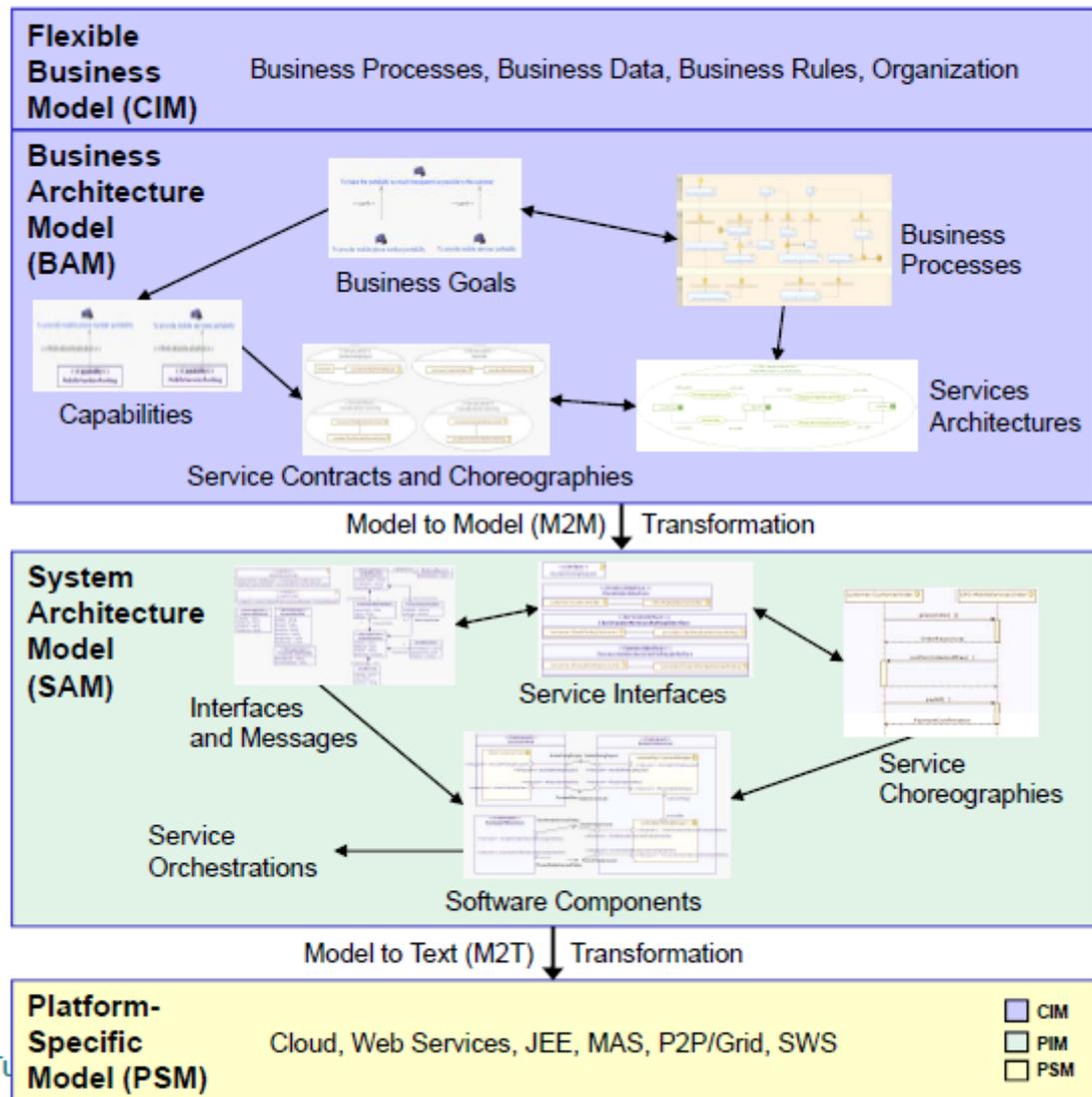


Figure II.14 Les modèles de base de la méthodologie Soaml [28].

### a. BAM (Business Architecture Model)

C'est le modèle de l'architecture métier. Il représente le perspectif métier de l'architecture SOA, définit les spécifications métier et identifie les services (les objectifs métiers, les processus métier, Les aptitudes) et spécifie la communauté métier et ses services.

### b. SAM (System Architecture Model)

C'est le modèle de l'architecture du système. Il représente la perspective technique de SOA. C'est une modélisation structurelle pour spécifier les composants,

## Chapitre II Architecture Orientée Services

---

leurs interfaces et leurs dépendances et comportementale pour spécifier l'interaction entre composants et les protocoles.

### II.9.3 MDA (Model Driven Architecture)

Le principe clé de MDA consiste en l'utilisation de modèles aux différentes phases de cycle de développement d'une application et offre des réponses aux comment, pourquoi, quand et quoi modéliser. Plus précisément, MDA préconise l'élaboration de modèles d'exigences (CIM), d'analyse et de conception (PIM) et de code (PSM) [29].

L'approche MDA définit quatre types de modèles, les CIMs, les PIMs, les PDMs (ou PMs) et les PSMs.

#### a. CIM (Computation Independent Model)

Appelé aussi modèle de domaine ou modèle métier. Son objectif est double : aider à la compréhension du problème et montrer le système dans son environnement organisationnel.

#### b. PIM (Platform Independent Model)

Ce modèle représente la logique métier de l'application. Comme exemple de PIM on peut noter un modèle d'architecture logique [30].

#### c. PDM (Platform Description Model) ou PM (Platform Model)

Décrit la plateforme sur laquelle le système va être exécuté ( : C#, EJB, EDOC, etc.). Dans une démarche MDA, on se base sur les PDMs pour générer les PSMs à partir des PIMs [31].

#### d. PSM (Platform Specific Model)

Le PSM sert essentiellement de base à la génération de code exécutable vers les plateformes techniques décrites. Un modèle de service Web est un modèle d'un PSM [20].

## Chapitre II Architecture Orientée Services

### Le principe de MDA

MDA propose un processus qui gère la transformation entre les modèles déjà décrits. Ce processus permet de transformer un modèle en un autre modèle du même système, mais à un niveau différent.

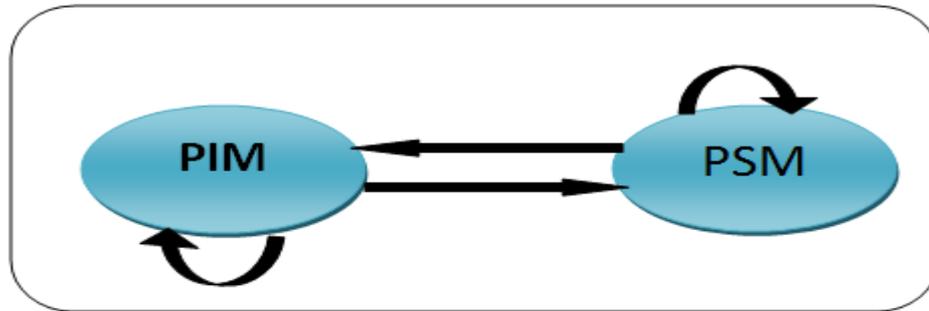


Figure II.15 Les transformations MDA

#### ✓ PIM vers PIM

Ces transformations enrichissent les modèles par un ensemble d'informations. La transformation du PIM vers PIM vise à le rendre plus complet et plus raffiné.

#### ✓ PIM vers PSM

Ces transformations s'effectuent lorsque les PIMs sont suffisamment enrichis et raffinés pour pouvoir être projetés sur une plateforme technologique. Cette projection est basée sur les caractéristiques de cette plateforme. Un exemple de ce genre de transformation est la transformation d'un modèle métier en UML vers un modèle utilisant une plateforme comme les services Web [31].

#### ✓ PSM vers PIM

Ces transformations doivent permettre d'obtenir un modèle indépendant à partir d'une implantation existante sur une plateforme spécifique. Ce sont généralement les transformations les plus difficiles à automatiser.

#### ✓ PSM vers PSM

Ces transformations s'appliquent sur un modèle spécifique et proposent un autre modèle spécifique à la même plateforme. Il s'agit de raffinement de plates-formes technologiques, de déploiement et d'optimisation [31].

## Chapitre II Architecture Orientée Services

---

### Conclusion

Aujourd'hui, les SI sont devenus l'échine de l'entreprise. L'innovation des entreprises moderne a posé les problèmes à cette échine. SOA, tel que l'idée de l'encapsulation des services a été considérée comme la meilleure solution. Le succès de SOA, pourtant, ne part pas de nouvelles technologies qui lui supportent, mais part plutôt de son approche. Peut-être on doit attendre dans quelques années pour avoir des infrastructures assez fortes qui ont la capacité de réaliser toutes les idées de SOA. Cependant, on a pu partir aujourd'hui pour commencer bénéficier ce que SOA fournit.

En résumé, SOA est une évolution des plates-formes passées, tout en préservant les caractéristiques réussies des architectures traditionnelles.

### Chapitre III

### Les Appels d'offres

---

#### **III.1 Introduction**

Afin que l'entreprise puisse s'adapter à un environnement en mouvance continue et perpétuelle générée par une croissance et évolution technique, et scientifique rapide, elle se trouve dans l'obligation d'établir une stratégie basée sur des moyens techniques et humains dont elle dispose. C'est dans ce cadre que les entreprises envisagent automatiser leurs activités. Parmi ces activités, la gestion des achats et plus spécifiquement celle par appel d'offres [32].

La gestion informatisée des appels d'offres intéresse de plus en plus les entreprises soucieuses de réduire leurs coûts. Le recours aux appels d'offres permet en effet d'effectuer des achats plus adaptés à leurs besoins, de mieux faire jouer la concurrence, et de faire baisser les prix. Seulement cette gestion demande du temps, des ressources et des compétences afin d'en récolter les avantages [33].

Les appels d'offres sont devenus en quelques années une pratique courante de tous les processus d'achat dans le monde des ventes. Ils traduisent le poids des directions des achats et l'intensité de la concurrence [34].

Dans ce chapitre, nous exposons le contexte général de l'appel d'offre, ainsi les ses différents types et les étapes du processus de l'appel d'offre.

#### **III.2 Les Appels d'offres**

##### **III.3.1 Définition**

Un appel d'offres est une pratique permettant à un maître d'ouvrage ou un commanditaire de choisir un fournisseur ou un soumissionnaire pour réaliser une prestation de services, de travaux ou de fournitures. Cette procédure est passée dans le respect des règles de la concurrence, dans le but de choisir l'offre économiquement la plus avantageuse afin de faire la meilleure utilisation des deniers publics.

## Chapitre III Les Appels d'Offres

---

L'Appel d'offres est une procédure organisée par la Maîtrise d'Ouvrage (MOA) d'un projet, afin de sélectionner le prestataire proposant les services ou expertises les plus adaptées à son besoin et au meilleur prix [35].

### III.3.2 Pourquoi les appels d'offres ?

Le but de l'appel d'offres est de réaliser une prestation de travaux, de fournitures ou de service et de mettre pour cela plusieurs entreprises en concurrence, afin d'obtenir un produit ou un service. Le choix du soumissionnaire qui sera un fournisseur vient suite à plusieurs commissions qui discuteront les offres selon les différentes soumissions [32].

L'appel d'offres est une mise en compétition et les soumissionnaires devront donc s'attacher à proposer des offres présentant le meilleur rapport qualité/prix, au regard du besoin du commanditaire [35].

### III.3.3 Le processus d'appel d'offres

Un appel d'offres se gagne au moyen d'un travail en équipe pour décider de la stratégie de réponse et formaliser de façon concurrentielle la réponse des entreprises [36].

Dans le processus d'appel d'offres, on distingue les étapes suivantes :

#### a. Approbation du dossier d'appel d'offres (DAO)

En effet, l'organisation d'un appel d'offres ne commence qu'à partir de l'approbation du dossier d'appel d'offres (DAO) par le bailleur de fonds ou le maître d'ouvrage. A cet étape, les fonds pour réaliser les travaux doivent être disponibles et tous les contours du projet analysés. Dans le cas contraire, les offres risqueraient d'être annulées ou les travaux traineront sur un délai trop long [37].

Le DAO est alors constitué de deux types de documents qui sont [37]:

#### ✓ Les pièces constitutives du DAO

- L'avis d'appel d'offres dûment signé et daté par le maître d'ouvrage

## Chapitre III Les Appels d'Offres

---

- Le Règlement Particulier d'Appel d'Offres (RPAO) ou les Instructions aux Soumissionnaires (IS)
  - Le Cahier des Clauses Administratives Particulières (CCAP)
  - Le Cahier des Clauses Techniques Particulières (CCTP)
  - Les spécifications techniques et les plans
  - Le Cadre du devis quantitatif et estimatif
  - Le Cadre du bordereau des prix unitaires
  - Le modèle de soumission
  - Etc.
- ✓ **Les pièces d'informations**
- Le modèle du Contrat
  - Les modèles de cautionnement provisoire et définitif
  - Etc.

### **b. La publication**

La publicité est faite soit par les radios, soit par les journaux et revues spécialisées, sur internet, et même dans les consulats des pays éligibles. Les délais de publications peuvent varier d'une dizaine de jours (en cas d'urgence) à 90 jours ou plus suivant la complexité des travaux à exécuter [37].

### **c. Ouverture des plis**

Les plis sont ouverts au jour, à l'heure et au lieu indiqués dans l'Avis d'Appel d'Offres (AAO). Une lettre invitant les soumissionnaires à assister à l'ouverture des plis aux dits jour, heure et lieu. Les représentants dûment mandatés des soumissionnaires peuvent assister à l'ouverture des plis et ceux qui le font, signent un registre attestant leur présence [38].

## Chapitre III Les Appels d'Offres

---

### **d. Evaluation des offres techniques**

La commission instituée à cet effet examine les informations fournies par le soumissionnaire. Elle doit vérifier si les offres techniques sont complètes, si les documents ont été correctement renseignés et signés ou paraphés selon le cas, et si les offres sont d'une façon générale en bonne et due forme.

Toute offre qui est jugée non conforme ou qui ne correspond pas aux niveaux minima des critères de qualification spécifiés dans les DPAO, sera rejetée par la structure contractante, et ne sera pas évaluée.

Pour les soumissionnaires dont les offres techniques auront été jugées conformes aux dispositions du Dossier d'Appel d'Offres, la commission instituée à cet effet procédera à l'examen et l'évaluation de leurs offres financières [38].

### **e. Evaluation des offres financières**

A l'issue de l'évaluation des offres techniques, la commission instituée à cet effet, évalue les offres financières des seuls soumissionnaires ayant présenté des offres techniques jugées complètes et conformes aux exigences du DAO. L'examen et l'évaluation des offres financières doit permettre de déterminer l'offre la plus avantageuse sur la base du critère unique du prix [38].

### **f. Attribution des marchés**

Le maître de l'ouvrage attribuera le marché au soumissionnaire dont l'offre a été reconnue conforme pour l'essentiel au dossier d'appel d'offres et qui a soumis l'offre estimée la plus avantageuse, et conclut le marché. Le soumissionnaire attributaire sera invité par écrit à se présenter pour signer le marché.

## **III.3 Types d'Appels d'Offres**

Un Appel d'Offres peut être un appel d'offre ouvert ou bien restreint. Les détails de chaque type avec le processus correspond sont les suivants :

## Chapitre III Les Appels d'Offres

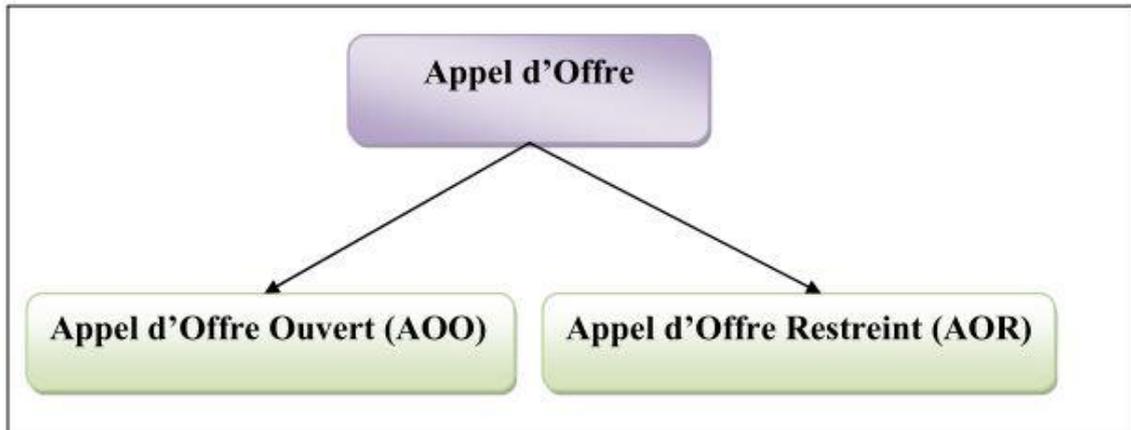


Figure III.1 Types d'appel d'offre

### III.3.1 Appel d'offres ouvert (AOO)

Procédure « classique » de passation de marchés publics par laquelle les entreprises peuvent se porter candidates [39].

Lors de l'appel d'offres ouvert, le maître d'ouvrage ne limite pas l'ouverture à la concurrence à un certain nombre de candidats. Les entreprises doivent transmettre des plis de candidature et d'offre bien distincts, mais cela avant une même date limite de réception. L'appel d'offres est dit ouvert lorsque tout candidat peut remettre une offre [40].

La figure III.2 illustre le processus de déroulement d'un appel d'offres ouvert.



Figure III.2 processus d'appel d'offre ouvert

## Chapitre III Les Appels d'Offres

### Appel d'offres restreint (AOR)

C'est la procédure du marché public selon laquelle l'accès à l'AO est limité à une liste restreinte de candidats (short-list), retenus grâce à une phase de présélection via un Avis à Manifestation d'Intérêt [41].

L'appel d'offres restreint où l'on ne s'adresse qu'à des candidats ciblés pour la concurrence. Ces candidats ciblés peuvent provenir d'un listing établi à l'issue d'une présélection (dossier de présélection organisé comme un dossier d'appel d'offres) ou données satisfaction plusieurs fois, Entreprises spécialisées, etc.) [37]. La figure III.3 présente comment se déroule un appel d'offres restreint.

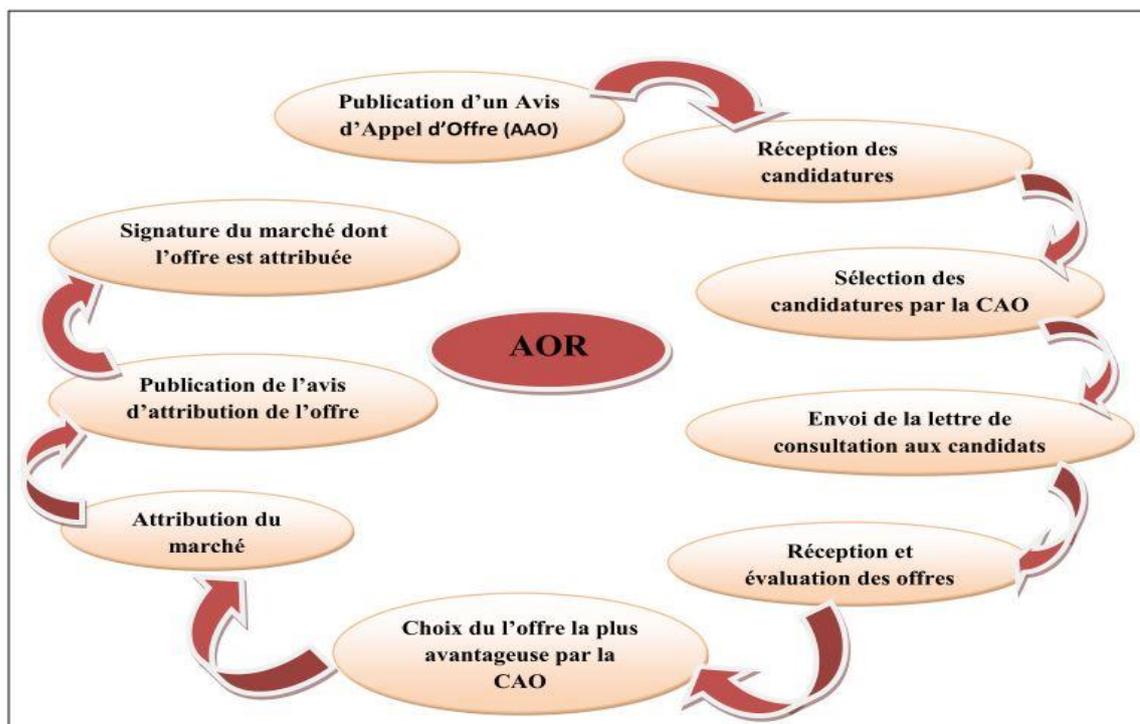


Figure III.3 Processus d'appel d'offre restreint.

### III.4 Conclusion

Le traitement d'un appel d'offres est une opération assez délicate qui nécessite l'art de convaincre son client pour qu'il porte son choix final sur la proposition que le candidat lui a faite. La première disposition de l'entreprise candidat est qu'elle est prête à être à l'écoute du maître d'ouvrage et/ou bailleur de fonds pour sa satisfaction et pour l'atteinte de ses objectifs. Il est alors intéressant d'étudier l'impact des appels d'offres sur les relations commerciales.

En conclusion, l'appel d'offres est une pratique qui a des conséquences importantes sur les relations commerciales.

## Chapitre IV

### Conception et réalisation de l'application

#### IV.1 Introduction

Dans ce chapitre, on présente la démarche de développement de notre projet, en parcourant les différentes étapes de modélisation et de l'implémentation de l'application. Pour accomplir cette tâche avec succès il faut savoir utiliser les outils adéquats et nécessaires.

#### IV.2 BPMN (Business Process Modeling Notation)

BPMN est un standard de modélisation des processus métier, élaboré en 2001 et il a été publié une première fois en 2004 [42]. BPMN a pour objectif de fournir un moyen simple et visuel de communication entre les différents intervenants chargés de mettre en œuvre une approche de gestion des processus métiers dans l'organisation. La figure IV.1 présente les éléments graphiques de BPMN.

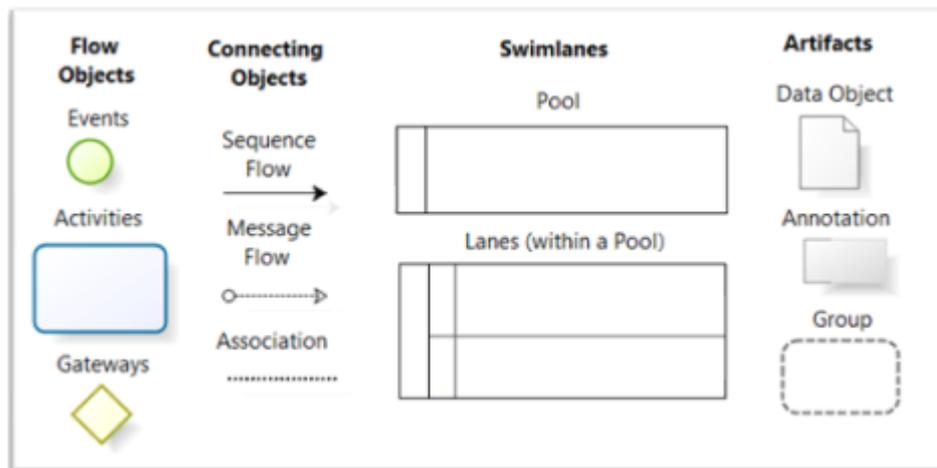


Figure IV.1 Les éléments graphiques de BPMN.

La notation BPMN couvre uniquement la description des éléments de la notation sans préciser de méthodologie particulière de sa mise en œuvre. La modélisation des processus métiers s'effectue en rassemblant ces éléments dans un diagramme de processus métier appelé BPD « Business Process Diagram » [42].

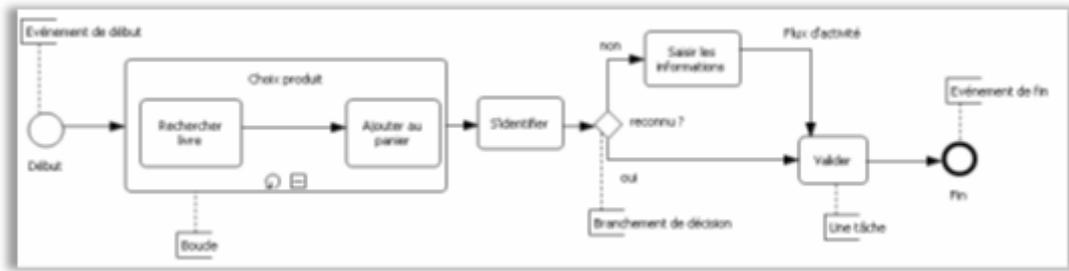


Figure IV.2 Exemple d'un BPD.

### IV.3 Modélisation

#### IV.3.1 Processus général de l'appel d'offres

Le processus se déclenche par la demande du maître d'ouvrage au service concerné pour créer un cahier de charges d'un appel d'offres selon les besoins (acheter un matériel, réaliser un projet .....etc.), l'expert rédige le cahier de charges et l'envoie à la commission des marchés pour l'approuver. Puis, ils font la publication de l'appel d'offres soit dans le site officiel de l'entreprise soit dans le journal officiel selon un délai bien spécifié.

Après la publication de cet appel d'offres, plusieurs fournisseurs vont la consulter et demandent d'acheter le cahier de charges. Chaque fournisseur étudie le cahier de charges pour participer à cette offre (disponibilité du matériel si c'est un appel d'offres pour acheter un matériel ou bien le délai pour réaliser un projet .....etc.). S'il est capable de participer à cet appel d'offres il déposera son offre dans une enveloppe qui contient deux enveloppes : un pour l'offre technique et l'autre pour l'offre financier.

L'entreprise reçoit plusieurs offres des différents fournisseurs, la commission d'ouverture des plis juste après le délai de terminaison de l'appel d'offres ouvre les plis un par un, après ça la commission d'évaluation des plis évalue les plis techniquement selon un barème. Si ce n'était pas bon elle l'élimine sinon elle l'évalue financièrement selon un autre barème et aussi si ce n'était pas bon elle l'élimine sinon la commission d'évaluation fait un classement des fournisseurs selon le résultat de leurs évaluations. La commission des marchés choisit l'un de ces fournisseurs qui a déposé l'offre la plus avantageuse techniquement et financièrement, et à la fin ils font la publication du résultat de l'appel d'offres dans le site officiel de l'entreprise ou bien dans le journal

# Chapitre IV Conception et Réalisation de l'Application

officiel. Chaque fournisseur consulte le résultat pour voir si l'offre lui a été attribuée. Dans ce cas, il contacte l'entreprise pour signer le marché.

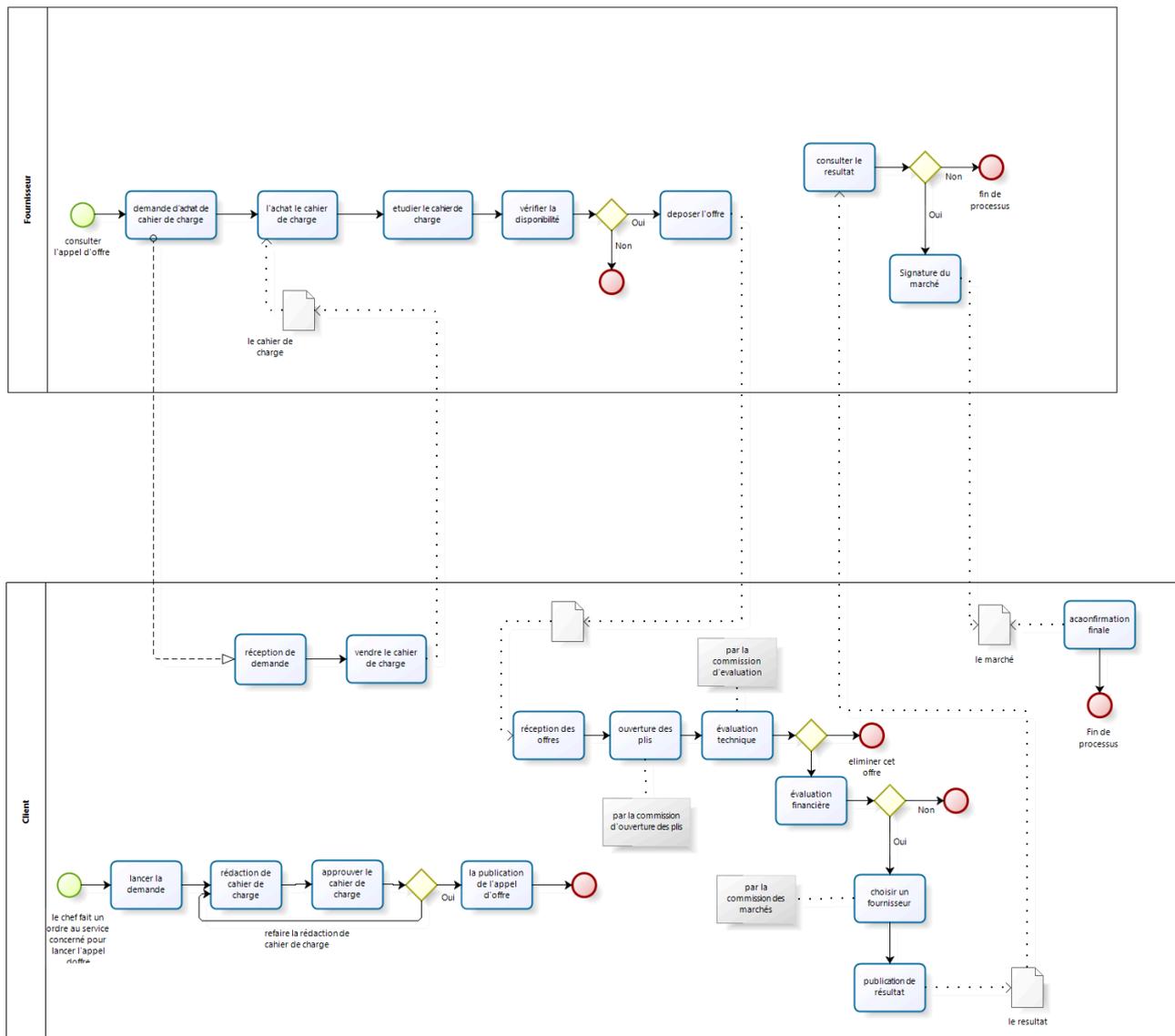


Figure IV.3 Le processus de gestion des stocks.

## IV.4 Implémentation

### IV.4.1 Plateforme de développement

La plupart des plateformes de développement fournissent le support de services web.

## Chapitre IV Conception et Réalisation de l'Application

---

(Outils et APIs).

- ✓ Plateforme .NET
- ✓ plateforme Java,
- ✓ plateforme PHP, C++, Python...etc

Ces outils permettent de manipuler des messages SOAP et des données au format XML, et accéder à la couche HTTP. Pour la réalisation de cette application, on a choisi la plateforme Java vue qu'elle est outillée, gratuite, accessible, légère et elle respecte des standards.

Ce choix de plateforme est accompagné d'autres outils d'implémentation:

- ✓ Le choix du serveur d'applications GlassFish Server 3 qui est basé essentiellement sur sa disponibilité.
- ✓ Pour le développement des services web, on a utilisé le Framework open-source JAX-WS 2.x vu qu'il a un package java donc multi-plateformes, conforme aux dernières évolutions de SOAP.

### a. Serveur d'applications



Pour le serveur d'applications on a utilisé le serveur **GlassFish Server 3** puisqu'il est open-source et il s'installe automatiquement avec netbeans. GlassFish Server 3 est un serveur d'applications parmi les meilleurs serveurs d'applications open-source du marché à l'heure actuelle. En plus, il s'installe automatiquement avec l'installation de Netbeans IDE.

### IV.4.2 Environnement de développement



Pour le développement des composants graphiques de notre système on a proposé de le faire avec le langage de programmation Orienté Objet Java, plus particulièrement **Netbeans IDE 6.9.1**. Le choix de ce dernier se justifie par ses

## Chapitre IV Conception et Réalisation de l'Application

---

caractéristiques, sa nature gratuite. Vu qu'il supporte de différents langages de programmation. Il permet l'accès et la gestion de bases de données et propose différents outils pour l'exploitation de web services.

### IV.4.3 Orchestration des services



**BizAgi** a pris sa place au marché du monde open-source. Il se base sur le standard BPMN pour la modélisation des processus métiers. Il est conçu pour les utilisateurs ayant peu d'expérience technique. Avec **BizAgi Process Modeler** on peut faire des diagrammes et documenter les traitements d'affaires d'une manière facile, intuitive et agile, en se basant sur le standard d'annotation BPMN.

### IV.4.4 Système de gestion de base de données



Le système de gestion de base de données qu'on a choisi est **MySQL** qui est un SGBD relationnel fonctionnant pratiquement sur plusieurs types de systèmes d'exploitation. Notre choix de **MySQL** se justifie par la disponibilité gratuite et aussi il s'agit d'un des SGBDR les plus connus et les plus utilisés. c'est certainement le SGBDR le plus utilisé à ce jour pour réaliser des sites web dynamiques.

### IV.4.5 Navigateur web



Pour tester les services web on a besoin d'un navigateur Web. Notre choix s'est fait sur le navigateur **Mozilla Firefox**. Firefox est un navigateur Web libre et gratuit et il a connu un succès croissant depuis sa création. **Mozilla Firefox** a introduit progressivement des fonctionnalités qui l'ont rendu flexible et facile à utiliser.

## Chapitre IV Conception et Réalisation de l'Application

---

### IV.4.6 Système d'exploitation



Notre application est réalisée dans le système d'exploitation le plus répandu, Windows7.

### IV.5 Description de l'application

#### IV.5.1 Côté serveur

Dans cette section, on veut montrer comment on a implémenté les services web qu'on a utilisé dans le traitement d'appel d'offres. En effet, on a créé plusieurs services web parmi eux le service CréerCC. Ce service contient une seule méthode créer qui sert à enregistrer les informations nécessaires d'un appel d'offres. Son code source est présenté comme suit:

```
public void creer(@WebParam(name = "a1")String a1, @WebParam(name = "a2")
String a2, @WebParam(name = "a3")String a3, @WebParam(name = "a4")String a4,
@WebParam(name = "a5") String a5, @WebParam(name = "a6")String a6) {
```

```
Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");//Charger le Driver et faire
connexion à la BDD
```

```
Con = DriverManager.getConnection("jdbc:odbc:appel");
```

```
Statement stm = Con.createStatement();// code SQL
```

```
Stmnt.executeUpdate(
```

```
stm.executeUpdate("insert into offre(Lot, Nom_AP, Nom_E, Adresse, date_limit,
Type) values ("'+a1+'","'+a2+'","'+a3+'","'+a4+'","'+a5+'","'+a6+'");}
```

Cette méthode est divisée en deux parties, la première pour établir une connexion à la base de donnée « appel » et la deuxième pour enregistrer les informations de l'appel d'offres.

✓ **Le fichier WSDL**

## Chapitre IV Conception et Réalisation de l'Application

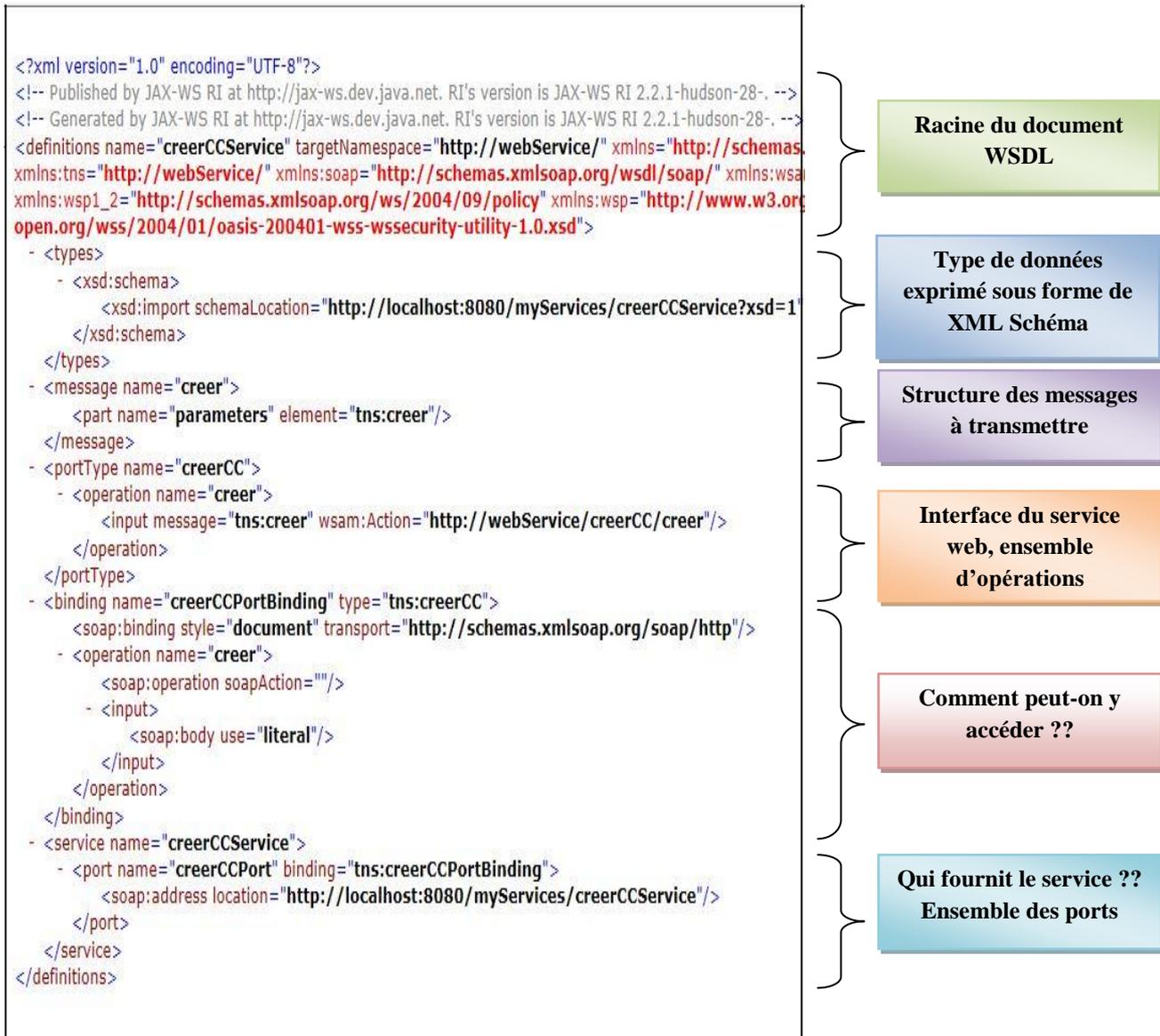


Figure IV.4 Fichier WSDL

### IV.5.2 Côté Client

Dans cette section, on a fait la consommation des services web qu'on a créé. En effet, on a créé des interfaces à l'aide des bibliothèques Swing du Netbeans dans laquelle on a fait un appel aux services créés.

### IV.6 Quelques exécutions

Lors de l'exécution de notre application on obtient une première interface comme s'est présenté par la figure IV.5. Dans cette interface il y a un accès Administrateur qui nécessite un login et mot de passe et un accès Client pour consulter les différents appels d'offres disponibles.



Figure IV.5 Page d'accueil



Figure IV.6

Accès Administrateur

## Chapitre IV Conception et Réalisation de l'Application

Quand on clique sur le bouton *Connexion* on obtient la fenêtre suivante :



**Figure IV.7 Page Administrateur**

Dans le menu de l'administrateur, il peut créer un cahier de charges, supprimer un appel d'offres, évaluer des soumissionnaires et attribuer l'offre à un soumissionnaire.

Parmi ces services, voilà le service de créer le cahier de charges de l'appel d'offres :

## Chapitre IV Conception et Réalisation de l'Application

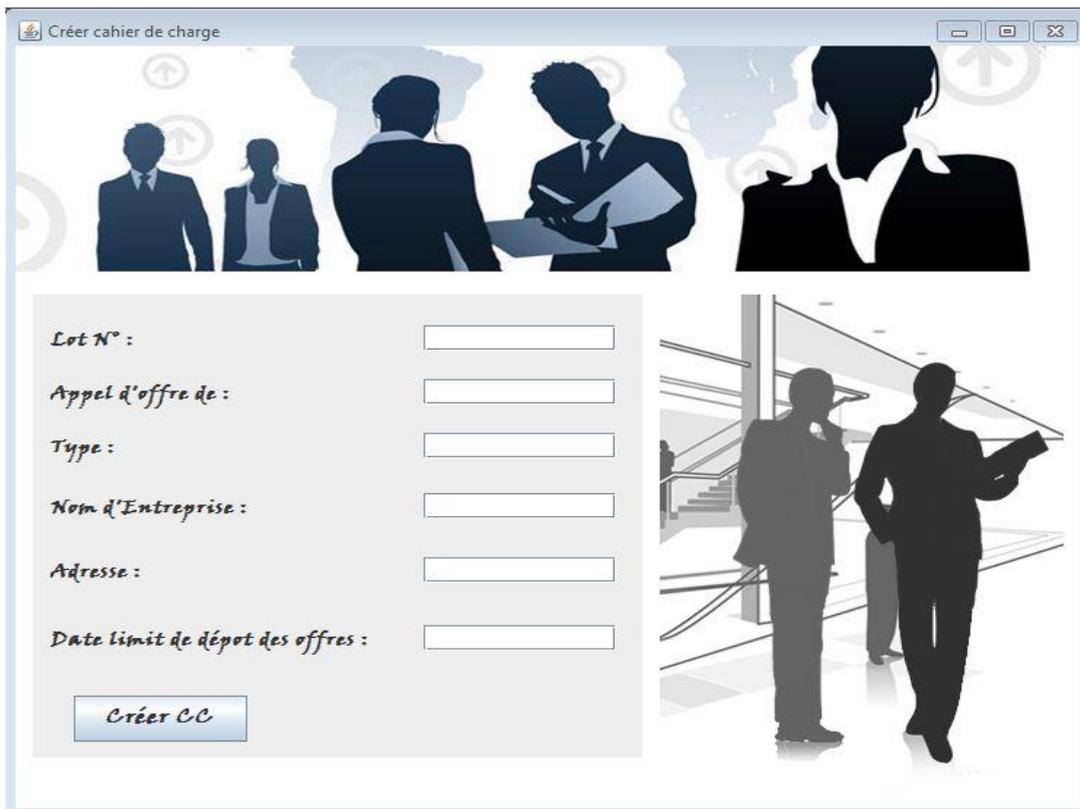
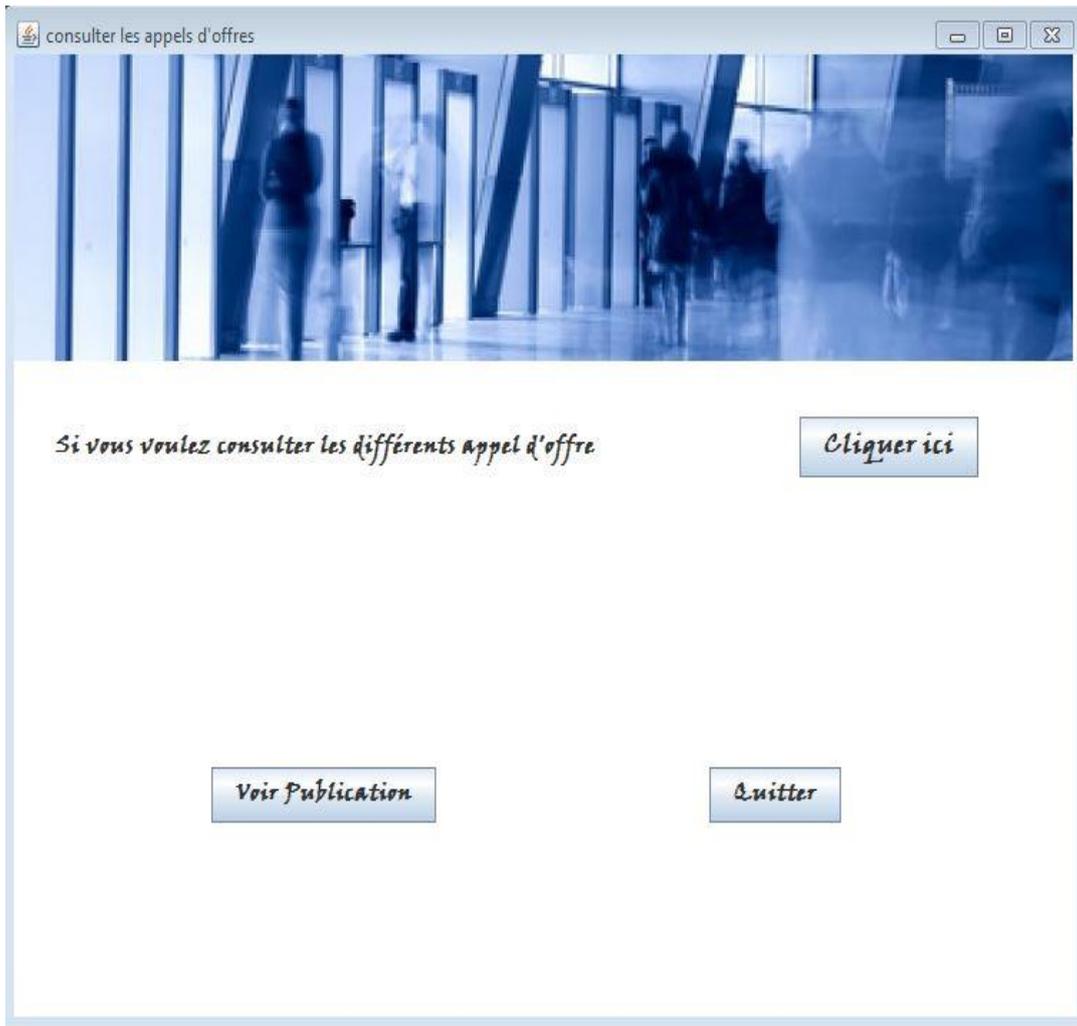


Figure IV.8 Créer le cahier de charge de l'appel d'offre

Le code suivant indique le gestionnaire d'évènement du bouton CréerCC

```
*****Code*****  
  
private void jButton1ActionPerformed(ActionEvent evt) {  
serv.CreerCCService service = new serv.CreerCCService()  
  
serv.CreerCC port = service.getCreerCCPort();  
  
port.creer(lt.getText(),ao.getText(),ne.getText(),as.getText(),date.getText(),type.getText  
());  
  
PublierAP c=new PublierAP(lt.getText(), ao.getText(), ne.getText(), as.getText(),  
date.getText());  
  
c.show();  
  
dispose();  
  
}  
  
*****fin code*****
```



**Figure IV.9 Accès Client**

Il y a deux boutons dans cette interface le premier pour voir les appels d'offres disponibles et le deuxième pour voir le cahier de charges d'un appel d'offres à partir de son numéro de lot. Lorsqu'on veut voir la publication d'un appel d'offre, la fenêtre de la figure IV.10 sera visualisée.

## Chapitre IV Conception et Réalisation de l'Application



**Figure IV.10 Publication de l'appel d'offres.**

Le client peut aussi participer à un appel d'offres et déposer son offre.

### IV.7 Conclusion

Dans ce chapitre, on a présenté la partie implémentation de notre travail, de ce fait on a intégrés l'approche SOA dans un système d'information des traitements des appels d'offre en se basant sur le processus général des appels d'offre. On a présenté aussi les outils nécessaires qu'on a utilisés pour finir ce travail.

En effet, on peut dire qu'il n'existe pas une seule solution pour finir ce travail, on a plein de solutions possibles pour ce traitement, on a choisis celle qui inclus l'approche SOA en raison de ses avantages qu'on a cité dans le chapitre II.

## Conclusion Générale

---

### Conclusion Générale

Le travail de PFE qui nous a été confié consistait à contribuer à une étude qui vise à intégrer une architecture technique de type SOA dans un système d'information.

Pour cette étude nous avons retenu un système de traitement des appels d'offre.

Après l'étude du domaine des appels d'offre, nous avons retenu les étapes de processus d'appels d'offre selon son type.

Une fois que nous avons acquises des connaissances sur l'architecture SOA basée sur les services réutilisables, nous avons conçue, pour les besoins de l'étude, un système de de traitement des appels d'offre qui intègre ce processus. Par la suite, on l'a implémenté sous une architecture SOA sous forme de web services.

En ce qui concerne la partie réalisation nous avons utilisé le SGBD MySQL, l'environnement de développement Netbeans et le serveur d'application GlassFish Server 3, Nous avons également utilisé l'outil «BizAgi» pour la modélisation et l'exécution de processus métier lié à le traitement des appels d'offre.

Néanmoins, comme tout projet le nôtre n'est pas exhaustif, il sera intéressant d'enrichir ce travail par :

- ✓ L'application de notre architecture sur différentes applications d'aide à la décision et pourquoi pas sur un SIAD.
- ✓ La mise en œuvre d'un ESB (donner accès aux services web pour différents utilisateurs).

## Références Bibliographiques

---

### Références Bibliographiques

- [1] : David Chappel, Au coeur de ActiveX et OLE, livre, 1996
- [2] : Jean-François, C.APPORT DE DIFFERENTS PARADIGMES DE PROGRAMMATION COMME AUTANT D'OUTILS DE PENSEE, p 199, P 201, p202
- [3] : Olivier HABERT, Programmation Orientée Objet (POO), Laboratoire d'Automatique des Systèmes Coopératifs (LASC)
- [4] : Emmanuel Hyon, Conception et Programmation Orientées Objet, Université Paris Ouest Nanterre, 7 février 2011.
- [5] : Occello Audrey, SAR O2/SAR O3 SOA, 2007.
- [6] : Hiver, paradigme de programmation, livre, 2007.
- [7] : Christian Pérez, Introduction aux Composants Logiciels, LIP/INRIA, 2010-11
- [8] : Sacha Krakowiak, Introduction aux composants, Université Joseph Fourier, 2005-2006.
- [9] : Pourraz, Frederic. 2007. "Diapason: une Approche Formelle Et Centrée Architecture Pour La Composition Évolutive Des Services Web". SAVOIE.
- [10] : Jay DiMare, IBM Global Services, 2006.
- [11] : Gartner Group, Note de recherche, 1996.
- [12] : Jean DALAT, Introduction à WCF, Association Dotnet France, Janvier 2009.
- [13] : Touzi, Jihed. 2007. "Aide à La Conception Des Systèmes D'information Collaboratif Support De L'interopérabilité Des Entreprises".
- [14] : Laurent GAYET, Principes, mise en œuvre et limites actuelles de la technologie OSGi, décembre 2006.
- [15] : Bieber, G. and Carpenter, J, Introduction to Service-Oriented Programming, (2002).

## Références Bibliographiques

---

- [16] : Hashimi, Sayed, “Service-Oriented Architecture Explained.” Windows Devcenter.com, 2003.
- [17] : Adrien, Louis,. “Bus De Service -ESB-. Nouvelle Technologie Pour L’intégration”. Livre blanc, 2008.
- [18] : LE Tanou, Yann, “Urbanisation & Intégration De Systèmes « Think Service» ”. Valtech Consulting Group, 2007.
- [19] : Pierre Bonnet , Cadre de Référence Architecture SOA, Orchestra Networks Paris, 2005.
- [20] : Thai Tri, Hung, “Architecture Des Systèmes D’information D’entreprise Et Architecture Orientée Service”. institut de la Francophonie pour l’informatique , 2005.
- [21] : Maesano, L, Bernard C, and X Le Galle, “Services Web En JEE Et .NET : Conception Et Implémentations”. Edition Eyrolles, 2003.
- [22] : Daehene, Peter, “Architecture SOA”. Haute école de gestion de Genève Informatique de gestion, 2009.
- [23] : Cyrielle.L, Florens.S, Sébastien.G, Rapport de TE, Université de Nice-Sophia Antipolis, 2005.
- [24] : Ahmed LAFTIMI , Architectures Orientées Services, CNAM 2008-2009
- [25] : [www-inf.int-evry.fr/cours/WebServices/Docs/Bob\\_WS-1.pdf](http://www-inf.int-evry.fr/cours/WebServices/Docs/Bob_WS-1.pdf) consulté le 25 Avril 2013.
- [26] : Wauquier, Dominique, “Modus : La Méthodologie Praxeme”, Praxeme Institute edition , 2006.
- [27] : Mongolo Assempra, Joël, “Refonte De La Plate Forme Architecturale Du Patrimoine Applicatif”, 2009.
- [28] : ] Elvesæter, Brian, “SoaML Tutorial Service Modelling with SoaML”. SHAPE, 2010.

## Références Bibliographiques

---

[29] : Xavier.Blanc, « MDA en action : ingénierie logicielle guidée par les modèles », Paris : Eyrolles, 2005.

[30] : NGUYEN Viet Hoa, Capitalisation des architectures métiers pour une implémentation sur différentes plates-formes techniques en utilisant la démarche MDA, INSTITUT DE LA FRANCOPHONIE POUR L'INFORMATIQUE, Juin 2008

[31] : Khoulood Boukadi, « Coopération interentreprises à la demande : Une approche flexible à base de services adaptables », Ecole National Supérieur des Mines, Saint-Étienne , Novembre 2009

[32] : Helmi GNICHI, « Conception et réalisation d'une application de gestion des marchés par appel d'offres au sein de l'Entreprise Tunisienne d'Activités Pétrolières », Institut supérieur d'informatique Tunisie, 2012.

[33] : CHACHOUA Embarka, « SYSTEME DE SUIVI DES APPELS D'OFFRES POUR L'ACTIVITE SONATRACH », Journal ENTP, 2004.

[34] : [https://fr.wikipedia.org/wiki/Appel\\_d%27offres](https://fr.wikipedia.org/wiki/Appel_d%27offres) consulté le 04/05/2013.

[35] : [www.afd.fr/webdav/shared/.../2.3\\_Conduire\\_des\\_Appels\\_d\\_Offres.doc](http://www.afd.fr/webdav/shared/.../2.3_Conduire_des_Appels_d_Offres.doc) consulté le 14/05/2013.

[36] : <http://www.buvetteetudiants.com/cours/167/les-appels-d-offres> consulté le 20/05/2013.

[37] : Habibou OURO-DJOBO, LE TRAITEMENT DES APPELS D'OFFRES DANS LE « BATIMENT ET TRAVAUX PUBLICS », Institut international d'ingénierie de l'eau et de l'Environnements, Octobre 2010.

[38] : [www.enafor.dz/pdf/INSTRUCTIONSOUUM.pdf](http://www.enafor.dz/pdf/INSTRUCTIONSOUUM.pdf) consulté le 30/04/2013.

## Références Bibliographiques

---

[39] : [http://www.marchespublicspme.com/marches\\_publics/41/](http://www.marchespublicspme.com/marches_publics/41/) consulté le 30/04/2013.

[40] : [http://www.franche-comte.fr/no\\_cache/services-en-ligne/marches-publics/en-savoir-plus/procedures/appel-doffres-ouvert.html?print=1](http://www.franche-comte.fr/no_cache/services-en-ligne/marches-publics/en-savoir-plus/procedures/appel-doffres-ouvert.html?print=1) consulté le 30/04/2013.

[41] : <http://www.docstoc.com/docs/122527622/A-quoi-sert-un-Appel-d%EF%BF%BDOffres> consulté le 01/05/2013.

[42] : Patrice Briol , « Ingenierie Des Processus Metiers: De L'elaboration a L'exploitation », 2008.

# Liste des Figures

---

## Liste des figures

Figure I.1 Principaux paradigmes de programmation .....	9
Figure I.2 Historique des différents paradigmes.....	9
Figure I.3 Interaction d'objets .....	10
Figure I.4 Exemple d'héritage.....	11
Figure II.1 Architecture basique de SOA .....	16
Figure II.2 Les propriétés d'un service.....	17
Figure II.3 Service à forte granularité versus service à faible granularité. ....	18
Figure II.4 Le principe d'orchestration des services.....	18
Figure II.5 organisation des services n'est pas en couplage faible.....	18
Figure II.6 organisation des services est en couplage faible.....	19
Figure II.7 Les composants techniques d'un service. ....	20
Figure II.8 le contrat de service. ....	21
Figure II.9 Les types de services. ....	22
Figure II.10 L'architecture orientée service web.....	23
Figure II.11 les éléments techniques du SOA.....	27
Figure II.12 Le mécanisme de SOA. ....	28
Figure II.13 Les aspects de la méthodologie Praxeme .....	29
Figure II.14 Les modèles de base de la méthodologie Soaml. ....	30
Figure II.15 Les transformations MDA .....	32
Figure III.1 Types d'appel d'offre .....	38
Figure III.2 processus d'appel d'offre ouvert .....	38
Figure III.3 Processus d'appel d'offre restreint. ....	39
Figure IV.1 Les éléments graphiques de BPMN. ....	41
Figure IV.2 Exemple d'un BPD. ....	42
Figure IV.3 Le processus de gestion des stocks .....	43
Figure IV.4 Fichier WSDL .....	47
Figure IV.5 Page d'accueil .....	48
Figure IV.6 Accès Administrateur.....	48
Figure IV.7 Page Administrateur.....	49
Figure IV.8 Créer le cahier de charge de l'appel d'offre.....	50
Figure IV.9 Accès Client .....	51
Figure IV.10 Publication de l'appel d'offres. ....	52