

Architecture Orientée Service, Urbanisation BPM(BPMN, BPEL) Concepts et Architectures

Yves LALOUM
Professeur Associé CNAM Paris

Page 1

Plan du cours

- 1- Introduction à l'urbanisation
- 2- Démarche globale
- 3- Architectures Orientées Services (SOA)
- 3- Les EAI
- 4- Le BPM

Page 2

Introduction

➤ Situation des entreprises :

- ✓ Rapidité de l'évolution de leur contexte / Concurrence / Technologie
- ✓ Rythme accéléré de renouvellement de l'offre au client
- ✓ Evolution de leurs produits , de leur organisation : besoins changeants, anticipation
- ✓ L'entreprise Agile : Réactivité, Flexibilité, Compétitivité, Efficacité, Souplesse

➤ Enjeux :

- ✓ Stratégiques : Fusions/Acquisitions, recentrage cœur de métier, cession d'activité
- ✓ Opérationnels : Transversalité des processus / Approche Client, Externalisation partielle, ...
- ✓ Financiers : Analyse et gestion de plus en plus approfondis et de plus en plus pointus, rythmes de plus en plus rapides

Situation des SI

Complexification du patrimoine SI à chaque évolution (spaghetti, strates géologiques, sédimentation, résultant d'une longue évolution et maintenance des SI)

- ✓ Incohérences, hétérogénéité, mauvaise communication, maîtrise difficile d'une multiplicité d'interfaces
- ✓ Coût d'exploitation et administration croissants des SI dans les entreprises
- ✓ Coût / MCO de l'existant , parfois 60 à 70 % du Budget Annuel
- ✓ Besoins d'échange des SI : B2B Ouverture vers/aux partenaires, B2C aux clients

➤ Enjeux des SI

- ✓ Gouvernance : audit, contrôle, gestion coût qualité délais, ROI justifiés et rapides
- ✓ Adéquation avec la stratégie et soutien opérationnel primordiaux
- ✓ Accroître la capacité à évoluer en maîtrisant les risques et les impacts
- ✓ Augmenter la cohérence, réduire les redondances entre applications (fonctions et données)
- ✓ Préparer le patrimoine SI à la flexibilité exigée par la nécessaire évolution de l'entreprise

Concepts et Définition

- L'urbanisation est une démarche globale visant :
 - Un système réorganisé autour de ses fonctions et de ses référentiels partagés
 - Une intégration des meilleures pratiques de l'état de l'art
 - Une Architecture Agile c.a.d : Flexibilité et possibilité d'évolution rapide
 - Une Architecture efficiente adaptée à la stratégie de l'entreprise et à ces évolutions

Quelques définitions

- Qu'est ce que la stratégie d'entreprise ?
 - « Élaborer la stratégie de l'entreprise, c'est choisir les domaines d'activité dans lesquels l'entreprise entend être présente et allouer des ressources de façon à ce qu'elle s'y maintienne et s'y développe. » [Source : Strategor]
 - La stratégie se décline à deux niveaux :
 - Stratégie de groupe : détermine les domaines d'activité de l'entreprise
 - Stratégie concurrentielle : mise en oeuvre dans chacun des domaines d'activités
- Qu'est ce qu'une stratégie système d'information ?
 - « Une stratégie système d'information doit définir un système d'information cible, les priorités, les étapes et les moyens nécessaires pour l'atteindre. »
- « Par ailleurs cette définition insiste sur le fait que ce sont les choix d'allocation de ressources (investissements et désinvestissements notamment), qui, davantage que les discours des dirigeants, font la stratégie. » [Source : Strategor]

Les citations proviennent de sources CIGREF

Concepts et définitions

- Architecture : Elaborer un « édifice » dans le cadre fixé par le POS

Concevoir et construire un ouvrage

- Terme très employé depuis les années 80/90 : structure d'un ensemble de composants et leur mode de relation/interactions pour former un tout
- Consiste à se préoccuper de mettre en œuvre un nouvel applicatif selon une démarche moderne et optimale, nouvelles méthodologies
- Réponse positive mais partielle aux impératifs de l'entreprise en matière de SI

Urbanisation : Concevoir et faire évoluer un SI global

Faire évoluer dans le temps un espace composé d'ouvrages

- Remonter d'un niveau d'abstraction : au niveau de l'ensemble du SI
- Objectif majeur : Acquérir une appréhension globale du SI, Coordonner les actions et décisions sur l'évolution du SI pour le faire évoluer au même rythme que l'entreprise et pouvoir prédire les conséquences d'une évolution incontournable.
- Moyens : Collecte des objectifs stratégiques, besoins métier, cartographie,
- Communiquer, convaincre, fédérer des acteurs de tous profils et de tous horizons

- Urbanisme / Architecture : Deux Termes plus voisins qu'opposés « on peut urbaniser une partie et architecturer le tout »

Urbanisme : Idées maîtresses

- Rôle du SI :
 - **Contributeur de plus en plus essentiel à l'efficacité des processus de l'entreprise**
- Démarche :
 - La référence : Vision stratégique et mission de l'entreprise
 - La dynamique : Vision des processus « métier »
 - L'objectif : définition de la cible idéale à atteindre
 - Le point de départ : audit de l'état actuel - cartographie
 - Contraintes et Enjeux : Critères de choix, degrés d'urgence, par quoi commencer et où finir ?
 - Aligner le SI sur l'entreprise d'aujourd'hui et de demain : déterminer une trajectoire, un plan de convergence.
 - Visibilité des résultats : Indicateurs Clés de Performance (KPI)

Page 9

Urbanisme : Idées maîtresses

POS Plan d'Occupation des Sols

- Echelle : au niveau de l'entreprise dans sa globalité
- Objet : définir les rôles et responsabilité de chaque sous-ensemble, la mission de chaque applicatif . Le regroupement en sous-ensembles cohérents
- ==> Forte cohérence et faible couplage

Zonage

- Zones Quartiers Ilots = Blocs
- Règles d'urbanisme
- Tout bloc doit comporter une « prise ». Interdiction d'accéder à un bloc autrement que par sa prise
- Une donnée est sous la responsabilité d'un et un seul bloc

Elaboration du POS et contrôle du respect de celui-ci

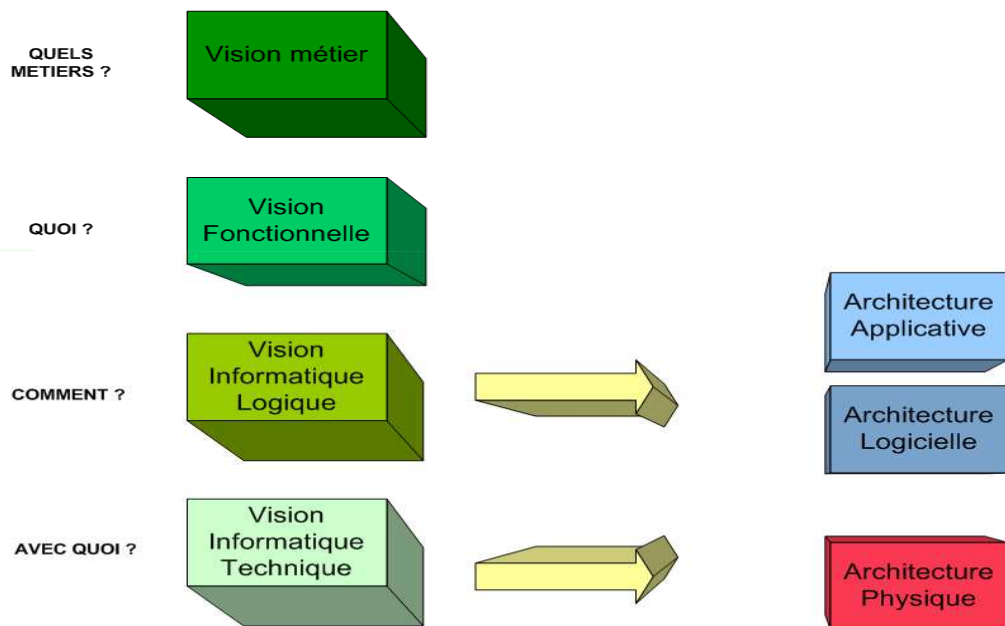
- Etude d'urbanisation , comité de pilotage, procédures de délivrance de permis de construire

Infrastructure :

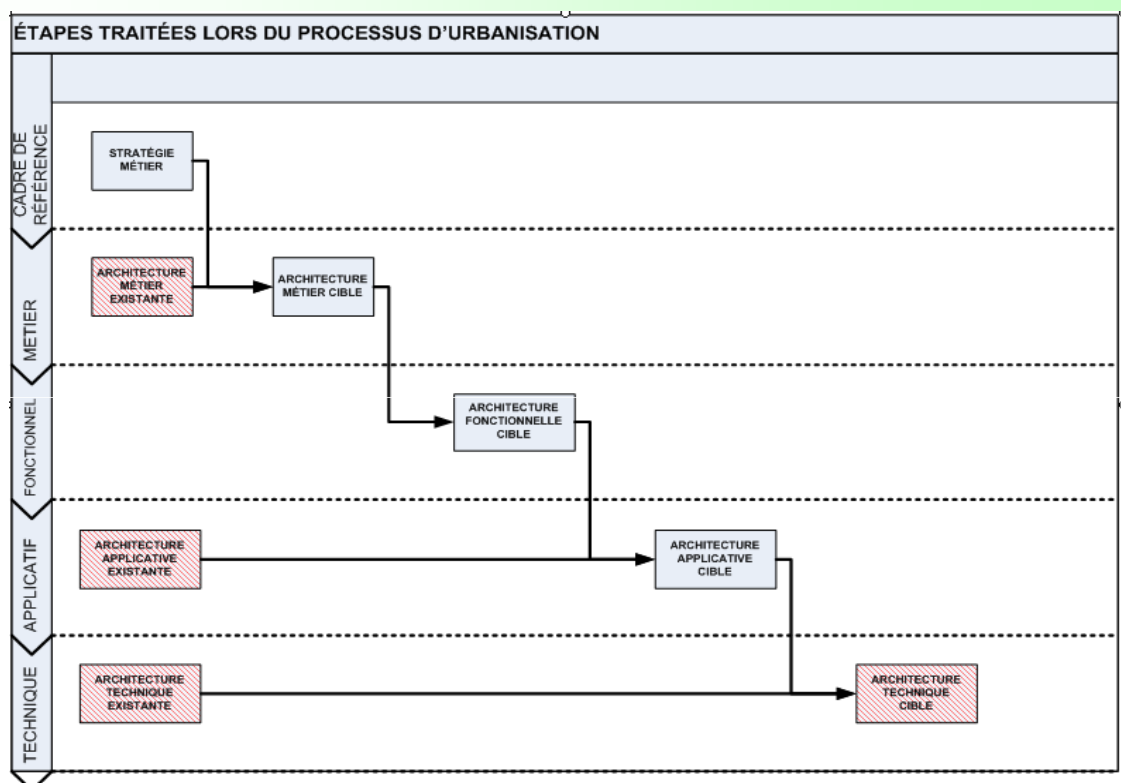
- Réseaux, plates-formes matérielles, logiciels de base (OS, SGBD, middleware, archi technique,...)
- ❑ Il n'est plus question d'approche « big bang » : Refonte = 100% de risque et seulement 20% du périmètre réellement amélioré

Page 10

Positionnement en 4 niveaux



Positionnement



Positionnement

- A chaque Niveau :
 - Démarche, décomposée en étapes, itérative et incrémentale :
 - bouclage sur les étapes
 - progression dans le degré de validation et de qualité du travail réalisé pour ce niveau
 - Règles d'urbanisme
 - Bonnes pratiques
 - Plus on passe à un niveau inférieur, plus on passe de règles « universelles » à des règles sujettes à adaptation à l'entreprise
 - En cible, les règles des niveaux applicatifs et techniques sont applicables à l'ensemble du SI

Modélisation Stratégie et Processus

- 1- **Modéliser la Stratégie** :

Diagramme d'Ishikawa pour les Objectifs stratégiques
Métiers

Diagrammes d'entreprise, actuel et cible.

Matrice Objectifs Métier/Objectifs S.I.

- 2- **Architecture Métier** :

Cartographie des processus. Situations actuelle et cible.

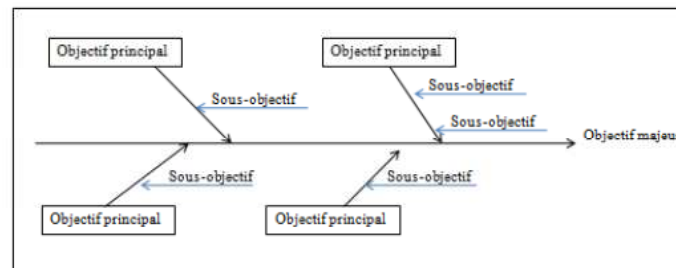
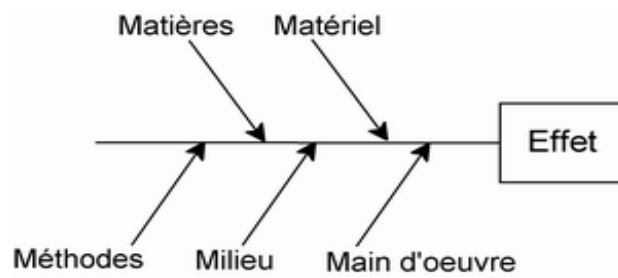
Matrice Objectifs Stratégiques Métiers/ Processus.

contributions (fortes ou faibles).

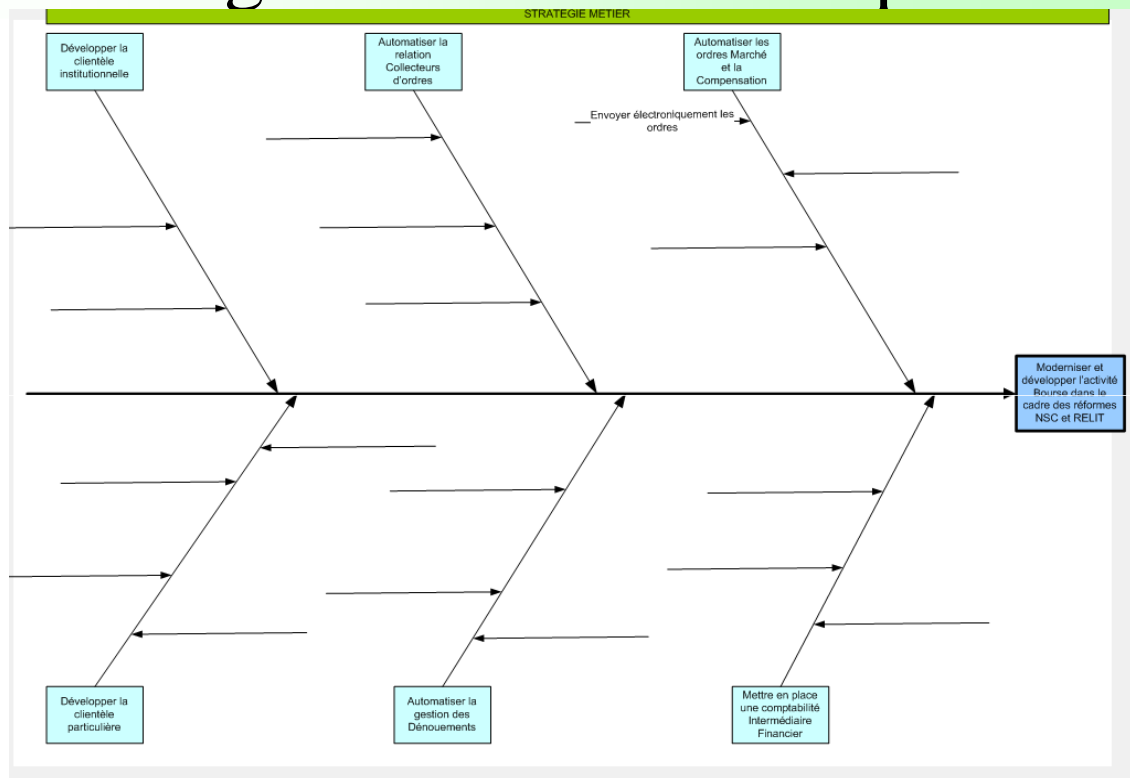
met en lumière les processus sensibles

Faire les modèles de processus pour chacun des processus.

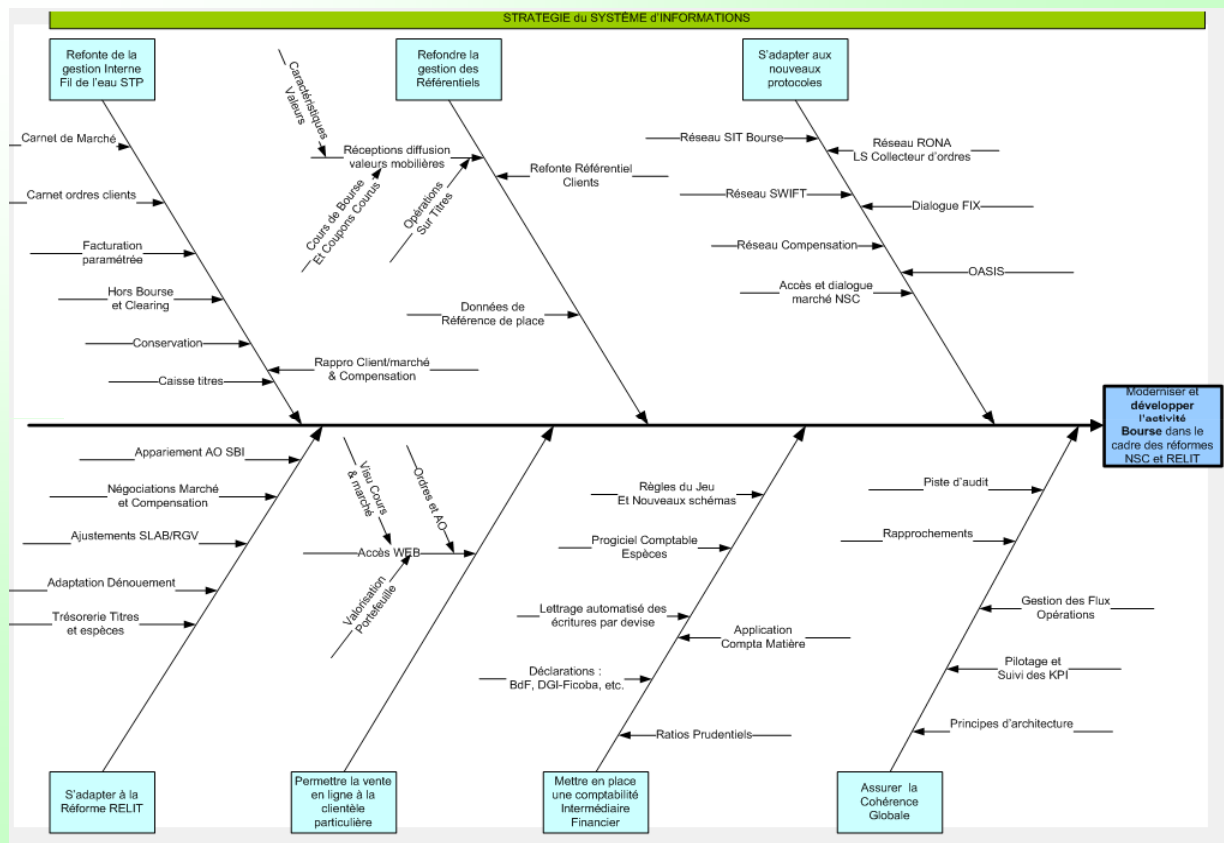
Diagramme D'Ishikawa



Stratégie : Ishikawa Entreprise



Stratégie : Ishikawa SI



Page 17

Modélisation Stratégie et Processus

• Règles d'urbanisme pour la modélisation de la stratégie

Diagramme d'Ishikawa

- n° 1 : Un même objectif ne figure qu'une seule fois dans le diagramme.
- n° 2 : Lorsqu'un objectif est décliné en sous-objectifs, la liste des sous-objectifs doit être exhaustive

• Bonnes pratiques pour la modélisation de la stratégie

Diagramme d'Ishikawa

- exemple : Un objectif commence par un verbe et ne comprend pas de « et »

Diagramme d'entreprise

- exemple : Privilégier la clarté et la lisibilité à l'exhaustivité.

Page 18

Modélisation Stratégie et Processus

- **Règles d'urbanisme pour l'architecture métier**

exemple : Une activité d'un processus appartient à un et un seul SI. Une activité d'un processus ne peut donc faire appel aux services que d'un seul SI.

- **Bonnes pratiques pour l'architecture métier**

Cartographie des processus

exemple : Les processus opérationnels, les processus de pilotage et les processus de support sont distingués.

Processus

exemple : La décomposition des processus est limitée à trois niveaux. Par définition un sous-processus est un processus et doit donc satisfaire la définition d'un processus.

exemple : Chaque déclenchement est associé à une décision qui peut commander une activité ou une autre encore.

exemple : Une activité peut nécessiter un ou plusieurs déclenchements si des activités doivent être synchronisées.

Processus et Processus Métier

• **Définitions**

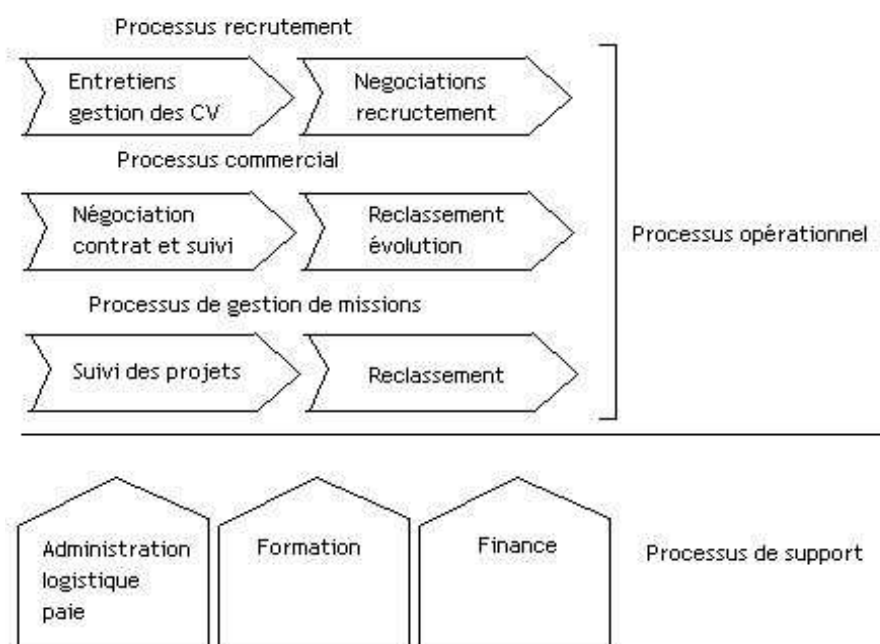
Un Processus :

est un ensemble d'activités ayant pour objectif le traitement d'un événement de gestion initiateur de ces activités. Il doit produire des flux de résultats définis dans les conditions de délais et de qualité fixés pour répondre aux besoins de tiers internes ou externes.

Processus métier :

processus mis en œuvre dans le cadre d'une organisation structurée. Il désigne une séquence d'opérations effectuées par des acteurs agissant sur des données et régies par des règles de gestion. Une processus métier est déclenché par une demande et finalisée par l'acceptation d'un résultat.

Exemple de cartographie des processus



Page 21

Matrice Processus/Objectifs Stratégiques

La matrice « Processus/Objectifs Stratégiques » permet de contrôler l'alignement des processus métier sur la stratégie de l'entreprise ou de l'organisation

- Les objectifs stratégiques sont en colonne (autant que de flèches sur le diagramme d'Ishikawa moins une qui correspond à l'objectif central de l'entreprise)
- Les processus sont en ligne
- Les cellules intersection des lignes et colonnes permettent d'indiquer la contribution du processus à l'atteinte de l'objectif :
 - Pas de contribution
 - Contribution faible : un dysfonctionnement sur le processus ne met pas en cause à lui seul l'atteinte de l'objectif
 - Contribution forte : un dysfonctionnement sur le processus peut mettre en cause à lui seul l'atteinte de l'objectif
- Sur la matrice, on peut identifier :
 - Les processus ne contribuant à aucun objectif stratégique
 - Les objectifs stratégiques non adressés par au moins un processus

Page 22

Passage au niveau fonctionnel

- Le lien entre architecture métier et fonctionnelle est assuré par l'association entre les classes 'activité' et 'bloc'. Faire l'architecture fonctionnelle de l'existant pour prévoir la migration vers la cible.

Le Quoi : indépendant des acteurs

- *- Appliquer les Bonnes pratiques qui permettent de définir *a priori* les zones pour l'architecture fonctionnelle cible.

ZONES : Echange | Référentiel de données et de règles | pilotage | 1 Gisement de données + 1 zone Opération par métier | Ressources.

- *- Exploiter les processus métier : Identifier les classes concepts (c'est-à-dire les invariants métiers) donc les classes de substance et les classes secondaires.
- *- Identifier les services des différents blocs composant l'architecture fonctionnelle.
- *- La cinquième étape poursuit l'exploitation des processus métier. Vérifier pour chaque activité du processus à quels blocs fonctionnels elle peut faire appel pour être menée à bien et si les blocs fonctionnels en question contiennent les services nécessaires.
- *- Bouclage par rapport aux objectifs stratégiques.
- *- Bouclage sur les règles d'urbanisme pour l'architecture fonctionnelle.

Modélisation Fonctionnelle

• Règles d'urbanisme pour l'architecture Fonctionnelle

- * : Règle d'unicité des blocs. Un îlot appartient à un et un seul quartier, un quartier appartient à une et une seule zone, donc un îlot appartient à une et une seule zone.
Au niveau de l'architecture fonctionnelle, un bloc ne doit donc pas être dupliqué.
- * : Un bloc comporte obligatoirement une prise (interface externe). Elle est capable d'activer les services du bloc et de gérer les communications entrantes et sortantes du bloc.
- * : Toute communication entrante ou sortante d'un bloc passe par sa prise.
 - centraliser les appels de services et limiter le nombre d'interfaces
 - Un bloc est à considérer comme une **boîte noire** par l'extérieur : **encapsulation**
 - mutualiser : un service public et un seul pour répondre à des besoins identiques formulés par des demandeurs différents appartenant le cas échéant à des blocs, quartiers ou zones distincts : **principe de réutilisation et cohérence**
 - accroître la modularité et réduire au strict minimum les impacts suite à l'évolution d'un îlot, détermination aisée de la chaîne d'impacts : **évolutivité, « maintenabilité »**
- * : Une donnée est sous la responsabilité d'un îlot et d'un seul
 - quel que soit le type d'accès : création, modification, suppression, visualisation. Un des objectifs de l'urbanisme : portabilité/indépendance des îlots grâce au respect des règles d'autonomie et d'asynchronisme.

Modélisation Fonctionnelle

- **Bonnes pratiques pour l'architecture fonctionnelle**

- * : Toute architecture fonctionnelle comporte une zone échange (acquisition/restitution) qui est en quelque sorte la **prise du SI**.

Formats pivot

- * : Toute architecture fonctionnelle comporte une zone « référentiel de données et de règles ».

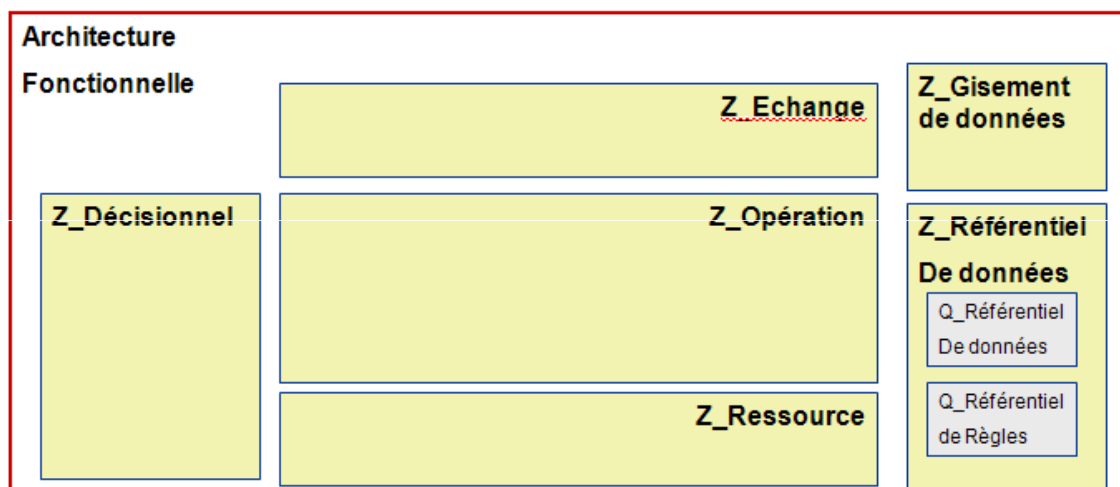
regroupe toutes les informations communes de référence dont le cycle de vie est relativement stable.

contient les règles de gestion administrative et comptable de la compagnie, ses métiers, son organisation

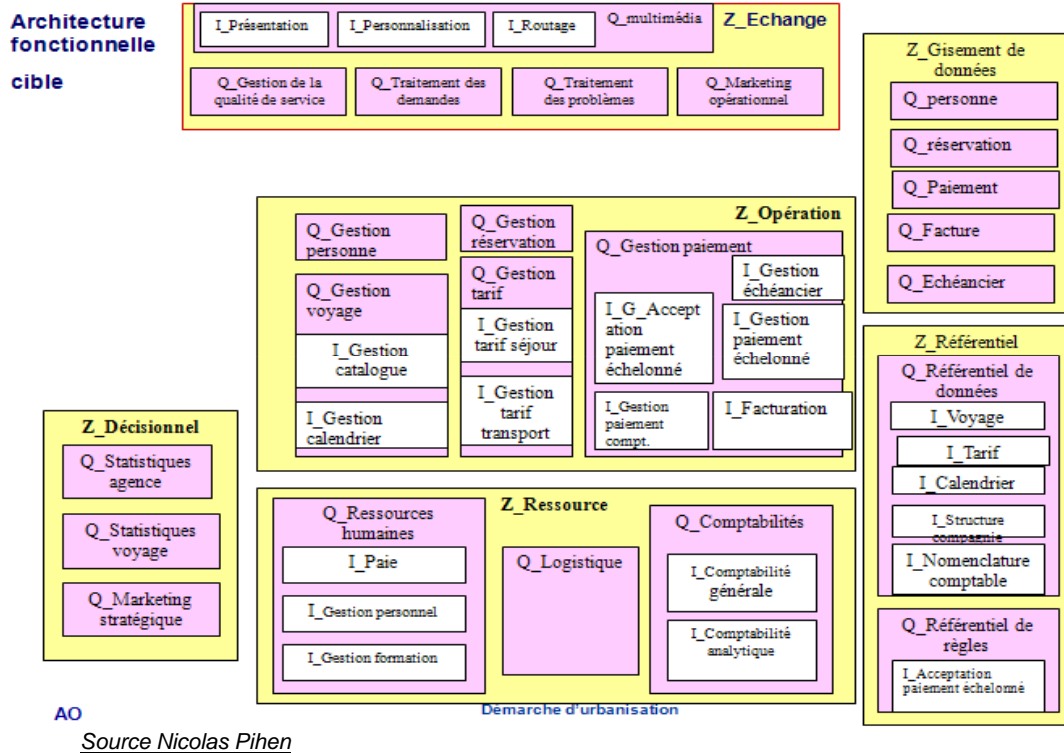
- * : Toute architecture fonctionnelle comporte une zone pilotage unique. blocs dédiés aux processus de gouvernance et d'analyse

- * : Toute architecture fonctionnelle comporte une zone opération par métier principal

Cartographie fonctionnelle générique



Exemple cartographie Fonctionnelle détaillée



Page 27

Passage Fonctionnel / Applicatif

- Le lien entre architecture fonctionnelle et applicative est assuré par l'association entre les classes 'bloc fonctionnel' et 'bloc applicatif'.

Le Comment : Les acteurs entrent en jeu

Etude de l'existant : cartographie des architectures Fonctionnelle et Applicative actuelles.

- * Questionnaire type par application (plus les questions sont fermées et mieux c'est)
- * Présenter aux équipes d'étude et de développement les résultats recherchés.
Les équipes étude et développement remplissent les fiches en bénéficiant du *coaching* de l'équipe urbanisation
- * Exploiter les retours. Les informations sont saisies dans l'outil retenu pour la cartographie.
- * Demander la validation de la cartographie applicative de l'existant ainsi réalisée.

Vision globale déjà mieux partagée.

Page 28

Passage Fonctionnel / Applicatif

- **Passage de l'architecture fonctionnelle à l'architecture applicative :**

- * Préciser les fonctions attendues du gestionnaire de flux.
- * Réaliser le mapping entre les architectures fonctionnelle et applicative (vue **statique**).
- * Décrire la prise et les fonctions de chacun des blocs applicatifs.
- * Projeter cette architecture applicative en cours d'élaboration dans l'organisation. Acteurs, différents types de sites, blocs applicatifs devant être instanciés de manière multiple.
- * Donner une vue **dynamique** de cette architecture applicative et à identifier les grandes **artères de communication**

Page 29

Architectures Applicative et logicielle

Démarche pour l'architecture applicative logique :

- Décrire de façon détaillée (fonctionnelle, applicative et technique) chacun des blocs applicatifs (interne, externe ou partenaire)
- Construire une cartographie applicative détaillée présentant tous les flux (synchrones/asynchrones, TP/batch) et messages échangés entre les blocs applicatifs (interne, externe ou partenaire) : **vision statique**
- Construire la matrice des flux à partir de la cartographie applicative des flux.
- A partir des cas d'utilisation identifier un nombre limité de cinématiques représentatives : **vision dynamique**

Démarche pour l'architecture applicative logicielle :

- Définir le modèle d'architecture en couches et en tiers à mettre en œuvre pour chacun des blocs applicatifs
- Préconiser motifs de conception, librairies, composants, frameworks (cadre de travail) . couches logicielles (présentation / coordination / services / domaine / persistance, gestion de logs,...)
- Guider les phases de conception et de développement . Coaching des concepteurs et développeurs

Page 30

Architectures Applicative et logicielle

- **Règles d'urbanisme pour l'architecture applicative**

- Les données des gisements de données doivent être accompagnées d'une date de publication de mise à jour.
- Les données des référentiels de données doivent être accompagnées d'une date de publication de mise à jour mais aussi d'une date d'effet.
- Duplication des données. Le niveau gisement de données doit rester maître. La synchronisation au sein d'un bloc se fait par publication du contexte en respectant la règle d'intégrité des gisements de données
- Le bloc offrant un service est le responsable de la qualité du service.

- **Bonnes pratiques pour l'architecture applicative**

- Toute architecture applicative comporte une zone ordonnancement
traduction, ordonnancement et pilotage des demandes. Demande traduite en un ensemble de services appelés dans un certain ordre
 - le pilotage des processus internes et gestion des priorités.

Architecture Technique et Physique

- **Démarche :**

Décrire l'architecture physique pour chaque bloc applicatif, en précisant les moyens d'infrastructure technique à mettre en œuvre

Définir la qualité de service attendue pour le système à partir des contraintes exprimées et décrire les solutions à mettre en œuvre : performance (volumétries statiques et dynamiques) disponibilité, sécurité

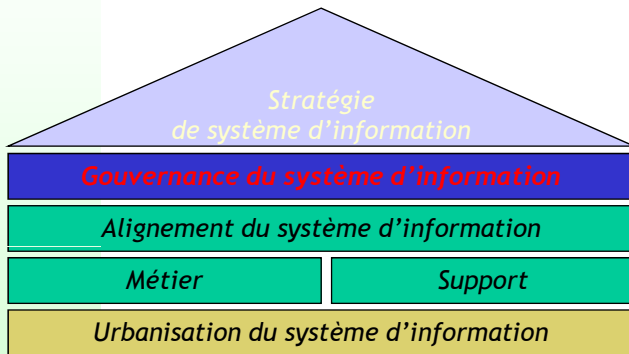
Définir un modèle de dimensionnement théorique à partir des hypothèses et des contraintes existantes et exprimées

Piloter/participer à l'étude de prototypage et au benchmark permettant ainsi de valider/affiner le modèle de dimensionnement de l'infrastructure technique qui sera testé sur des cas d'utilisation représentatifs

Définir l'architecture de l'exploitation (administration / supervision du SI) Déploiement, configuration, procédures d'exploitation, de supervision

Déterminer le coût du système

Les trois composantes d'une stratégie ?



Source P. VERGER

Page 33

La gouvernance désigne « l'art ou la manière de gouverner » un système

Elle se distingue du gouvernement vu en tant qu'institution

Elle décrit comment un système est dirigé et contrôlé :

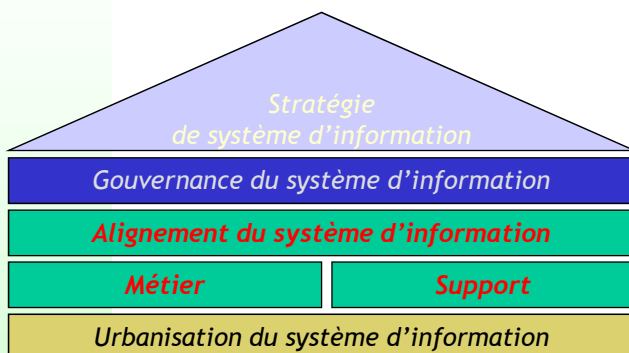
Pilotage

S'assurer, sachant d'où on part, que l'on va bien où l'on veut arriver

Contrôle

Savoir mesurer à tout instant où on est et l'écart qu'il y a par rapport à où on devrait être

Les trois composantes d'une stratégie ?



Source P. VERGER

Page 34

L'alignement a pour objectif que le système d'information serve de manière optimale la stratégie métier de l'entreprise

L'alignement vise donc à « **renforcer la valeur d'usage** » du système d'information et faire de celui-ci un atout pour l'entreprise

La distinction métier / support :

Métier d'un hôpital : soigner les patients

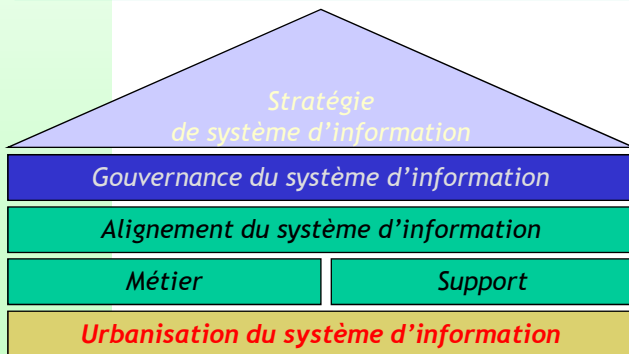
Support d'un hôpital : gestion des ressources humaines, gestion financière, ...

Les trois composantes d'une stratégie ?

Urbanisation

faire que le système d'information serve la stratégie de l'entreprise et soit à même de supporter les changements de son environnement

NB :
l'urbanisation inclut l'alignement



Source P. VERGER

Page 35

Urbanisation : mise en oeuvre du plan d'urbanisme

Plan d'urbanisme du système d'information

- Définition d'un système d'information cible
 - Adapté aux besoins
 - Adaptable aux changements
- Des règles d'urbanisme
- D'une représentation de l'existant permettant de définir une trajectoire vers la cible

Urbanisation : La Métaphore

- Le complexe urbain :
 - Cohabitation d'anciens et de nouveaux quartiers
 - Partage de services communs
 - Infrastructure routière
 - Electricité, gaz, eau,
 - Téléphone
- Le système d'information
 - Anciennes et nouvelles applications
 - Anciennes et nouvelles machines
 - Communications
 - Middleware
 - Services communs : annuaire, messagerie, ressources humaines

Page 36

Objectifs: Pourquoi urbaniser ?

- Intégrer les nouvelles applications harmonieusement dans l'existant de l'entreprise
- Maintenir la cohérence globale du SI
- Préservation des investissements logiciels
- Rendre les applications inter opérables,
- optimiser le nombre d'applications et les flux échangés.
- Orchestrer les applications dans les processus métiers de l'entreprise
- Définir des référentiels et des règles ainsi qu'un cadre cohérent, stable et modulaire auquel les différentes parties prenantes se réfèrent pour toute évolution du Système d'Information

Page 37

Votre SI est-il urbanisé ?

•Quelques questions clés :

•Pouvez vous facilement développer et distribuer de nouveaux produits ?

–Pouvez vous facilement fournir de nouveaux services aux directions opérationnelles, notamment dans une logique transversale ?

–Constatez vous une augmentation des tâches à « non-valeur ajoutée » (ressaisies, dédoublemnages, doubles vérifications) ?

–Arrivez vous à maintenir un niveau de service en cohérence avec l'évolution technologique ?

–Constatez vous un surcoût d'exploitation et maintenance (métier et informatique) ?

–Constatez vous une persistance voire un accroissement de la non-qualité ?

–Constatez vous une diminution voire une absence de maîtrise sur le SI ?

–Constatez vous un accroissement des vulnérabilités et risques de dysfonctionnement générés par l'augmentation de la complexité du SI ?

–Constatez vous une difficultés à intégrer les technologies nouvelles ?

–Constatez vous une non maîtrise des évolutions logicielles (rustines, versions...) par manque de cible identifiée ?

–Constatez vous une dégradation de l'image de la DSI vis-à-vis des utilisateurs et des directions opérationnelles ?

•Si vous répondez oui ou non selon le cas ...

Page 38

Mise en place d'un plan d'urbanisme

- Le plan d'urbanisme est une vision partagée et transversale du système d'information
 - Il a un rôle de décloisonnement
 - Il permet de gérer l'obsolescence normale du système d'information
 - gérer un patrimoine
 - remplacer l'obsolète
 - préserver la capacité d'évolution
 - C'est un investissement économique
 - Il faut donc évaluer les risques
 - risques si l'on fait
 - risques si l'on ne fait pas
 - risques si l'on reporte
- d'abord fait pour améliorer la capacité du système d'information à servir la stratégie de l'entreprise
 - Il ne s'agit pas d'un projet technique
- Le plan d'urbanisme est un projet d'entreprise avec un mode de financement adapté
 - La réflexion sur le mode de financement est une contribution à l'établissement d'une gouvernance
 - Il n'y a pas de règles universelles valables partout
- Le plan d'urbanisme inclut un plan de conduite du changement

Page 39

L'alignement du système d'information

- La valeur d'un système d'information est la valeur d'usage de l'information qu'il permet
 - So far - to my best knowledge - nobody has been able to demonstrate that there is a positive correlation between money spent on IT and sustainable profits (Paul Strassman)
- L'alignement stratégique du système d'information suppose deux conditions
 - compréhension et intégration de la stratégie de l'entreprise par la fonction système d'information dans son ensemble
 - prise en compte des contraintes et des opportunités de l'informatique dans la stratégie de l'entreprise
- Causes possibles de non alignement
 - absence de stratégie claire de l'entreprise
 - perception inexacte de la stratégie
 - absence de stratégie SI
 - dans la formulation de la stratégie non-prise en compte
 - du système d'information
 - du potentiel des technologies de l'information
 - organisation en silos de spécialisation au détriment de la transversalité
 - absence de dialogue entre les métiers, la direction générale et la direction des SI
 - pas de centre de décision concernant le SI
 - pas de « proactivité » de la fonction SI

Page 40

Dans ces causes, l'élément majeur est l'absence de gouvernance

Acteurs et conditions de l'alignement

- L'alignement stratégique concerne d'abord
 - la direction générale
 - les directions métiers
 - la direction des systèmes d'information
- Le dialogue entre ces acteurs est le plus important
 - Un dialogue entre utilisateurs et direction des systèmes d'information est essentiel
- Une stratégie système d'information se construit à partir d'éléments entrants
 - la stratégie de l'entreprise et celle des différents métiers
 - une analyse critique de l'existant
- interne : le système d'information de l'entreprise, les ressources, les partenariats ...
- externe : le potentiel technologique, l'état de l'art, la concurrence sur le marché des fournisseurs...
- La stratégie système d'information définit
 - un système d'information cible
 - la description des priorités, des étapes et des moyens nécessaires pour l'atteinte de cette cible
- Une stratégie système d'information est par essence dynamique
 - Un plan stratégique est toujours « en glissement »

Stratégie

- Stratégie de système d'information
 - Gouvernance : Pilotage et contrôle du système d'information
 - Alignement : Adaptation au métier (valeur d'usage du SI)
 - Urbanisation : SI cible adapté au métier et supportant les changements

Système d'information et informatique

- réservé aux informaticiens De la vision traditionnelle...
- L'informatique est un centre de coût
- L'informatique est un moyen
- L'informatique est une fonction transversale de support et non stratégique
- L'informatique est un bien privatif, cloisonné à chaque service ou direction
- L'informatique est un domaine
- ... vers une vision intégrée
- Le système d'information est un élément clé de la chaîne de valeur
- Le système d'information est un actif de l'entreprise
- Le système d'information est une fonction de transformation stratégique
- Le système d'information est un bien collectif pour l'entreprise, partagé par tous, assurant la coopération transversale des acteurs
- Le système d'information est un domaine transversal à l'entreprise, au service de tous

Page 43

La gouvernance du Système d'Information.

- La gouvernance décrit comment un système est dirigé et contrôlé.
- Tous ces éléments descriptifs d'une démarche d'urbanisation sont aussi autant de choix fondamentaux de la gouvernance du Système d'Information de l'entreprise.
- Il est donc possible, utile, nécessaire, ... d'étudier le positionnement de l'urbanisation dans le cadre de la gouvernance du système d'information.
- Avantages :
 - Meilleure compréhension de toutes les parties prenantes
 - Meilleure intégration dans le dispositif global

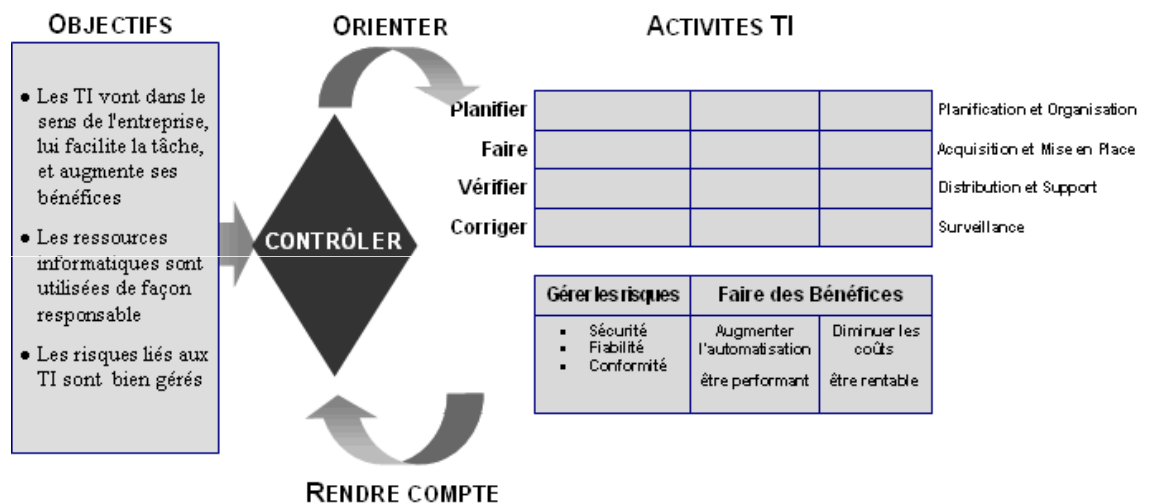
Page 44

COBIT

- COBIT standard de la gouvernance du Système d'Information
 - Maintenant adopté comme étant le cadre standard de la gouvernance du SI soutenu en France par l'AFAI et le CIGREF
 - Issu du monde des auditeurs du SI
 - Internationalement reconnu
 - Utilisable et utilisé par l'ensemble des parties prenantes
 - Son utilisation devient incontournable
 - Englobe d'autres standards (ITIL, ISO2700x, CMMI...).
- COBIT est un cadre pour la gouvernance des SI.
- www.isaca.org
- www.afai.fr



Principes de COBIT

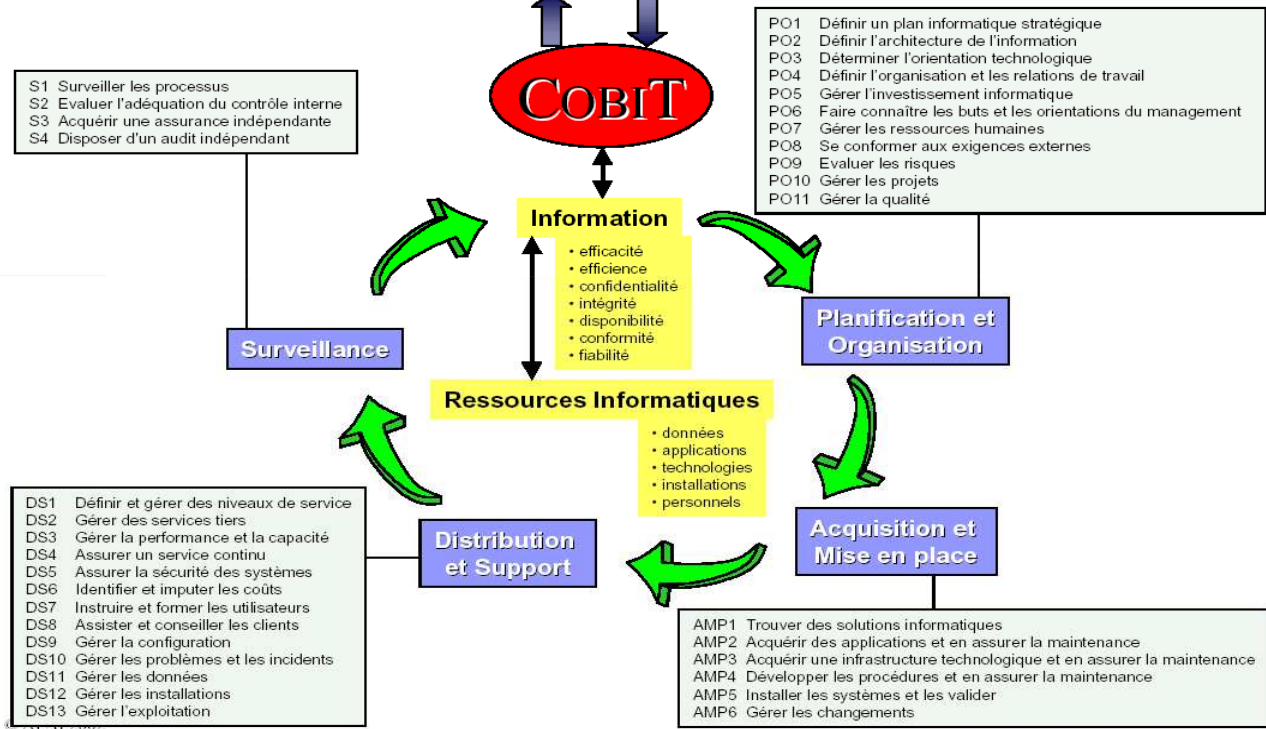


La démarche COBIT

Objectifs de l'Entreprise

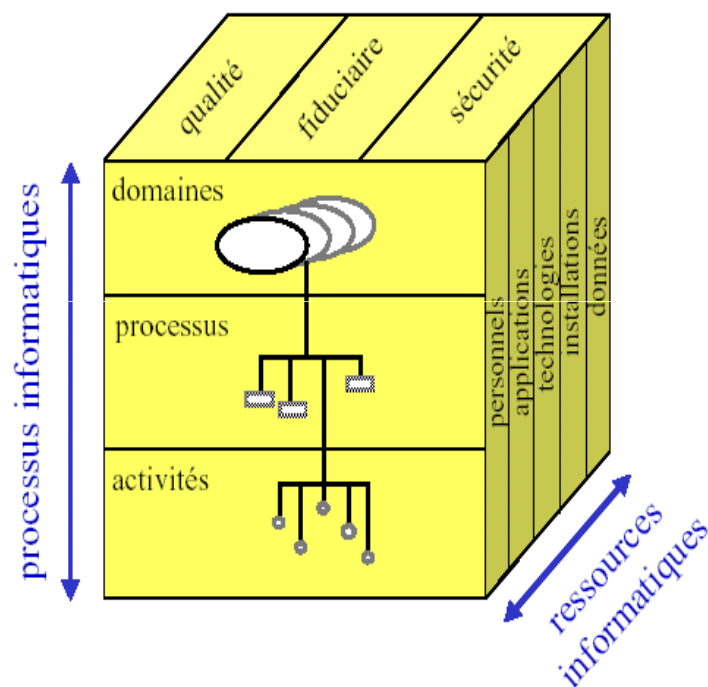
A F A I
ANALYSE FINANCIÈRE ET DE GESTION
ET DE CONSEILS INFORMATIQUES SA

Gouvernance des Technologies de l'Information



La vision COBIT du SI

critères liés à l'information



La démarche d'urbanisation

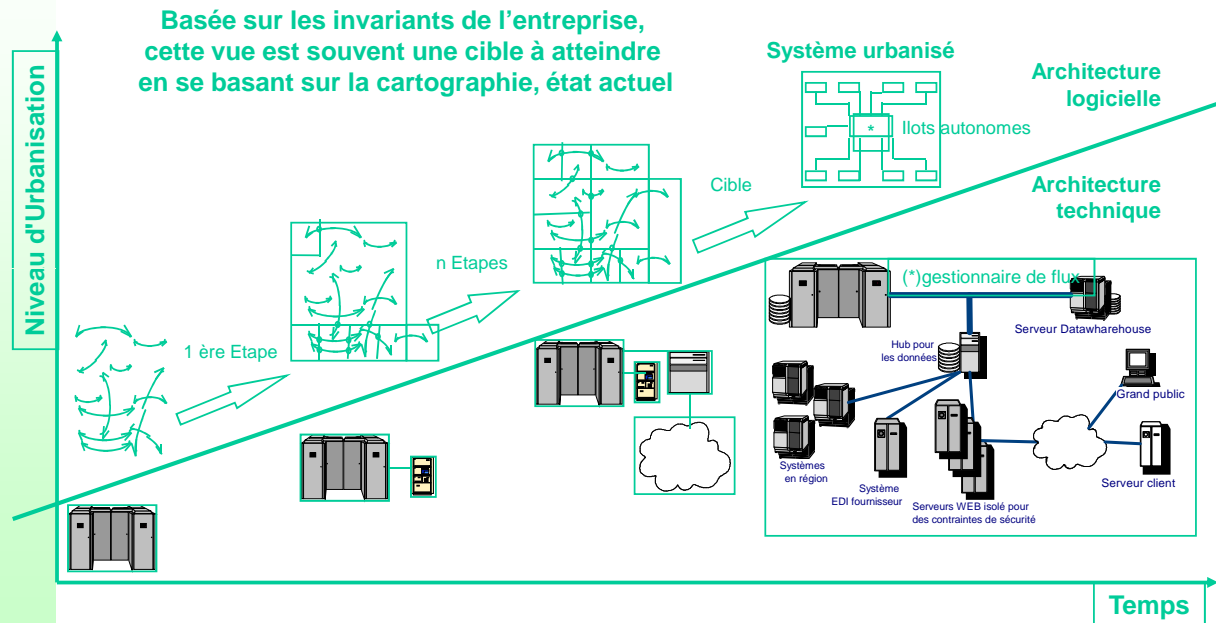
Page 49

La démarche : Comment Urbaniser ?

- Les étapes :
 - Constitution de la cartographie du SI
 - Etablissement des règles d'urbanisation,
 - Mise en conformité du SI existant,
 - Gestion des besoins d'évolution

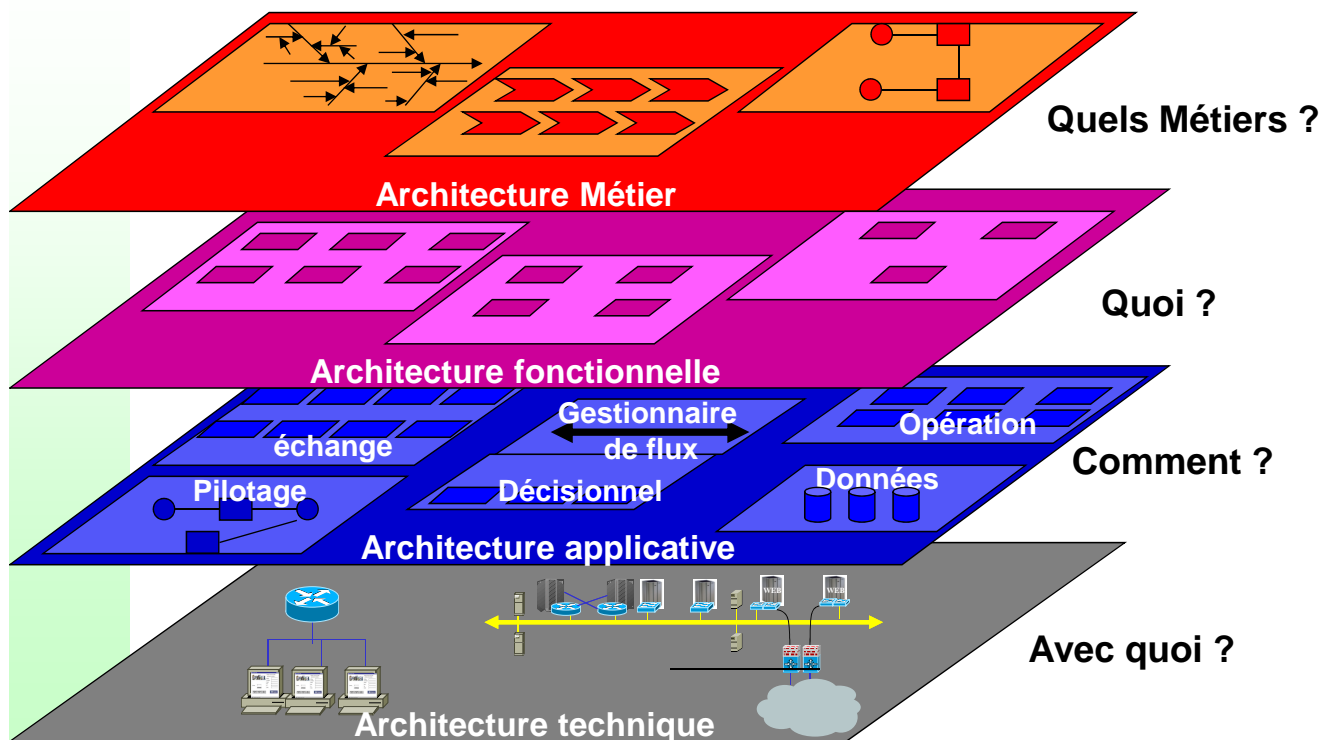
Page 50

Démarche itérative et incrémentale



Page 51

Les différentes Architectures



Page 52

Etablissement de la cartographie

Objectifs :

- Répertorier l'ensemble des informations du système d'informations.
- Représenter les cibles d'urbanisme construits au fur et à mesure, grâce aux plans de migrations définis

Cartographie : Niveaux de représentation

- La cartographie est composée de quatre niveaux de représentations du SI :
 - Les processus métier : modéliser les processus de fonctionnement
 - Les applications : vue globale de toutes les liaisons inter-applications
 - Les données : relations entre les données (MCD)
 - Les moyens techniques : toutes les applications circulant sur un serveur
- Apports :
 - Une meilleure connaissance de l'existant
 - Une facilité dans la suppression des applications redondantes
 - Une réalisation de la représentation des cibles d'urbanisme

Établissement des règles

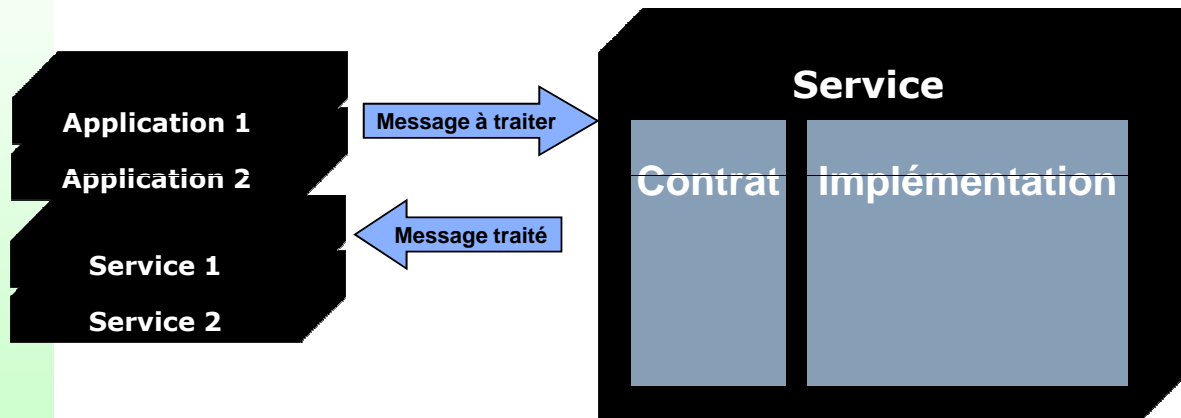
- Objectifs :
 - Définir un cadre structurant aux applications de l'entreprise
 - Mise en place d'une SOA : Service Oriented Architecture
 - Les applications sont vues comme des services
 - Utilisation d'un bus logiciel commun : EAI (Enterprise Application Integration)
 - Implémenter les processus métiers : BPM (Business Process Management)

Architecture Orientée Service

- « Une vision d'un système destinée à traiter toute application comme un fournisseur de services »
- Tout traitement applicatif est vu comme un service potentiellement utilisable par un client
- Le client est découplé de l'architecture technique du service qu'il invoque
- SOA véhicule des Messages et non des objets
- Pour adapter les technologies hétérogènes, on fait appel à des Connecteurs/Adaptateurs

Architecture Orientée Service

- Un service est une fonction qui reçoit des messages et qui les restitue après traitement.
- Une Architecture Orientée Service diffuse des services



- Un service respecte un contrat
 - Un service est sans état
- Un service est réutilisable

Architecture SOA

- Il n'existe pas "une seule" architecture orientée services ou même des techniques caractérisant une SOA.
- La mise en place d'une architecture orientée services répond à un besoin : la réutilisation des traitements.
- Le contexte de réutilisation de ces traitements va dicter des besoins fonctionnels : interopérabilité, fiabilité, sécurité, hétérogénéité, pérennité, adaptabilité, ACIDité...
- De ces besoins fonctionnels vont découler des choix technologiques: couplage lâche pour garantir interopérabilité, asynchronisme pour garantir la fiabilité,...
- Les technologies sont au service de l'architecture et pas l'inverse.

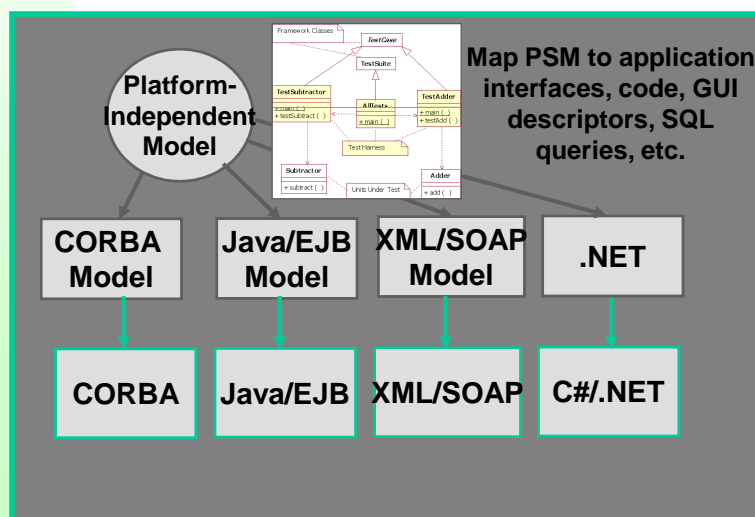
SOA Fonctionnalités prise en charge

- Indépendance vis à vis de la plate-forme
- Indépendance vis à vis du langage de programmation
- Découverte dynamique des services
- Fiabilité
- Sécurité
- Gestion des transactions
- Orchestration des services
- Possibilité d'enregistrer les séquences de messages
- Possibilité de produire des rapports d'audit
- Routage intelligent des messages
- Tolérance aux pannes, montée en charge
- Capacité à agir en fonction d'indicateur
- Système de versionning

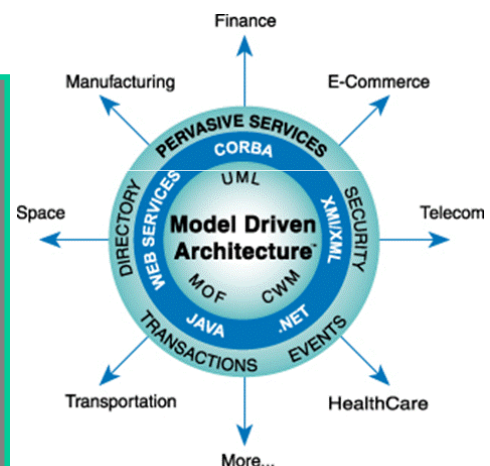
Page 59

MDA: Model Driven Architecture

- Démarche pilotée par les modèles
- PIM (Platform Independent Model)
- PSM (Platform Specific Model)



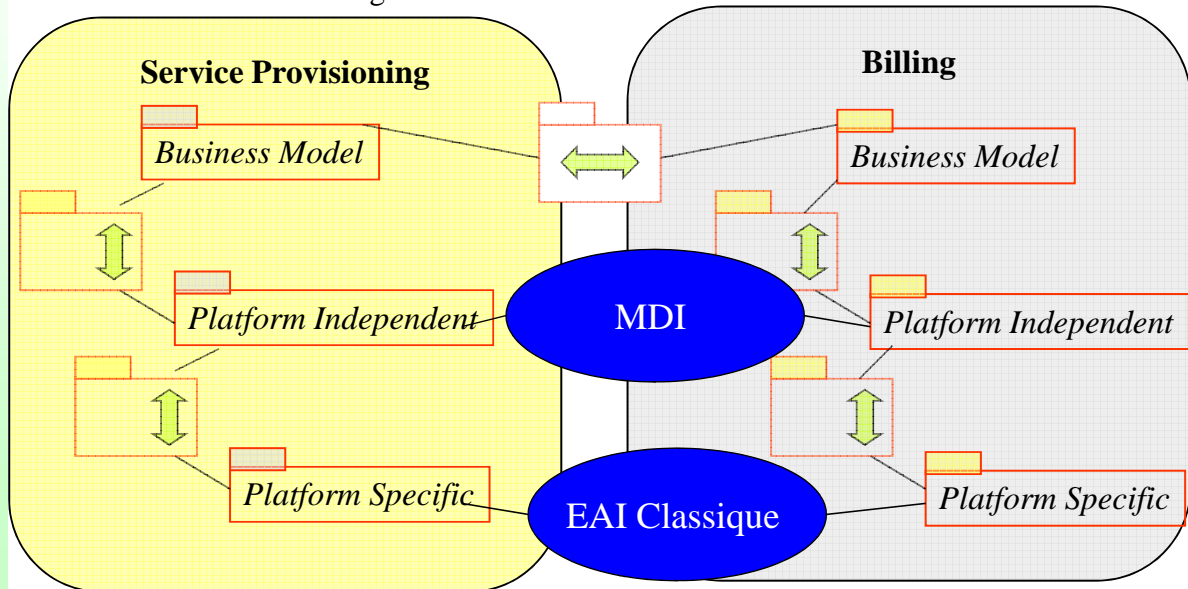
Page 60



Model Driven Integration

– Business Process Models permet la composition d'applications

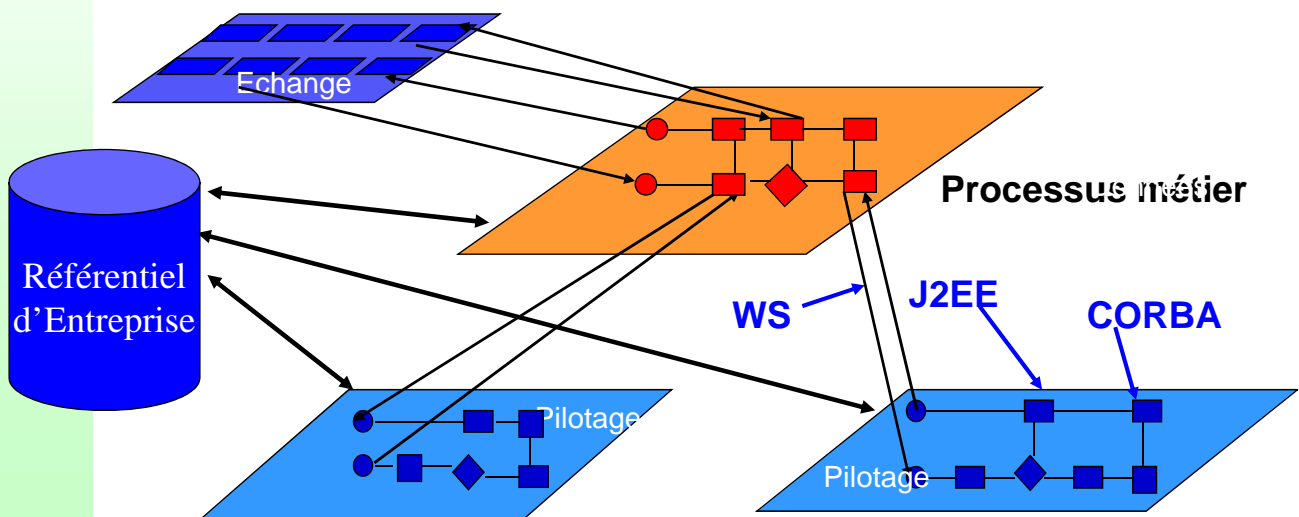
- Les messages et documents échangés dans la plate-forme d'intégration sont définis par les outils de modélisation
- Le mapping de la Plate-forme Indépendant Model au Plate-forme Specific Model est effectué en bordure de l'outil d'intégration



Page 61

Principe du MDA

- Composition d'applications sans tenir compte des technologies
- Les données échangées sont définies par l'outil de modélisation
- BPM gère les processus de l'entreprise



Page 62

SOA et WEB SERVICES

Page 63

WEB SERVICES

Rappel sur les WEB SERVICES

Page 64

Objectifs

- Accès rapide, intégré et généralisé à l'information pertinente
 - en interne (Intranet) et en externe (Extranet, Internet)
- Système ouvert réduisant les coûts
 - capable d'inter-opérer avec les applications existantes
 - capable d'inter-opérer avec le monde extérieur (Extranet, Internet)
- Développement rapide d'applications (RAD)
 - utilisation de composants distribués
 - réduction des coûts de développement
- Administration simplifiée du système
 - depuis le Web avec des outils d'assistance

Page 65

Besoins des Applications Web

- Accès programmable à des services
 - Cours de bourse
 - Entrées d'annuaires
 - Autorisation de cartes de crédits
 - Authentification d'un client
 - Enregistrement de clients
 - ...
- Interface d'accès par fonctions
 - URL longues contraignantes
 - Pouvoir découvrir dynamiquement les services
 - Paramètres structurés et diversifiés

Page 66

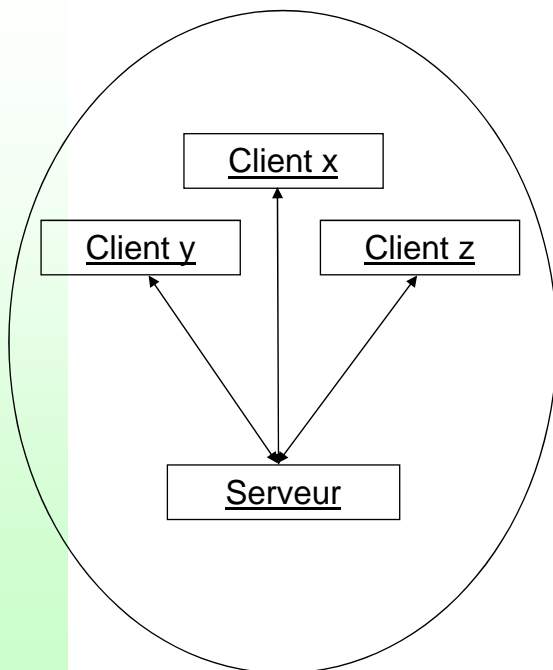
Exemple d'applications

- Diffusion d'information
 - horaires, incidents, états de stocks, etc.
- Documentation automatique
 - librairie électronique, manuels, maintenance, etc.
- Gestion de sites Web dynamiques
 - présentation, marketing, veille technologique, etc.
- Communication et "knowledge management"
 - échanges de données, gestion ressources humaines, etc.
- Commerce électronique
 - présentation, sélection, transaction, médiation, etc.

Page 67

Architecture

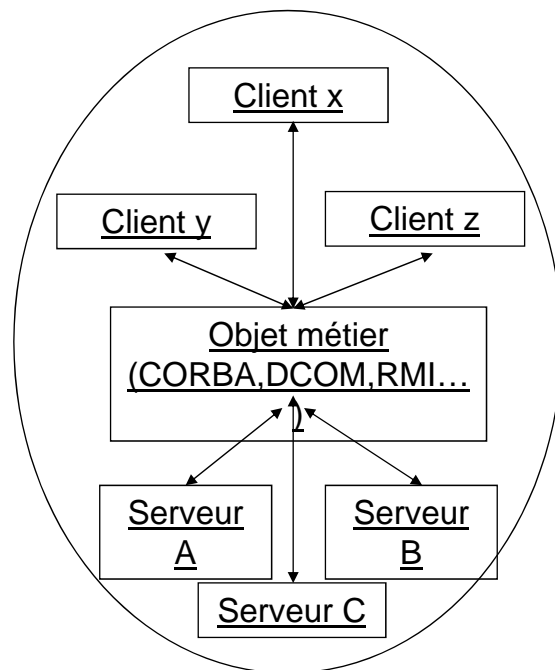
2 tiers



INTRANET

Architecture n

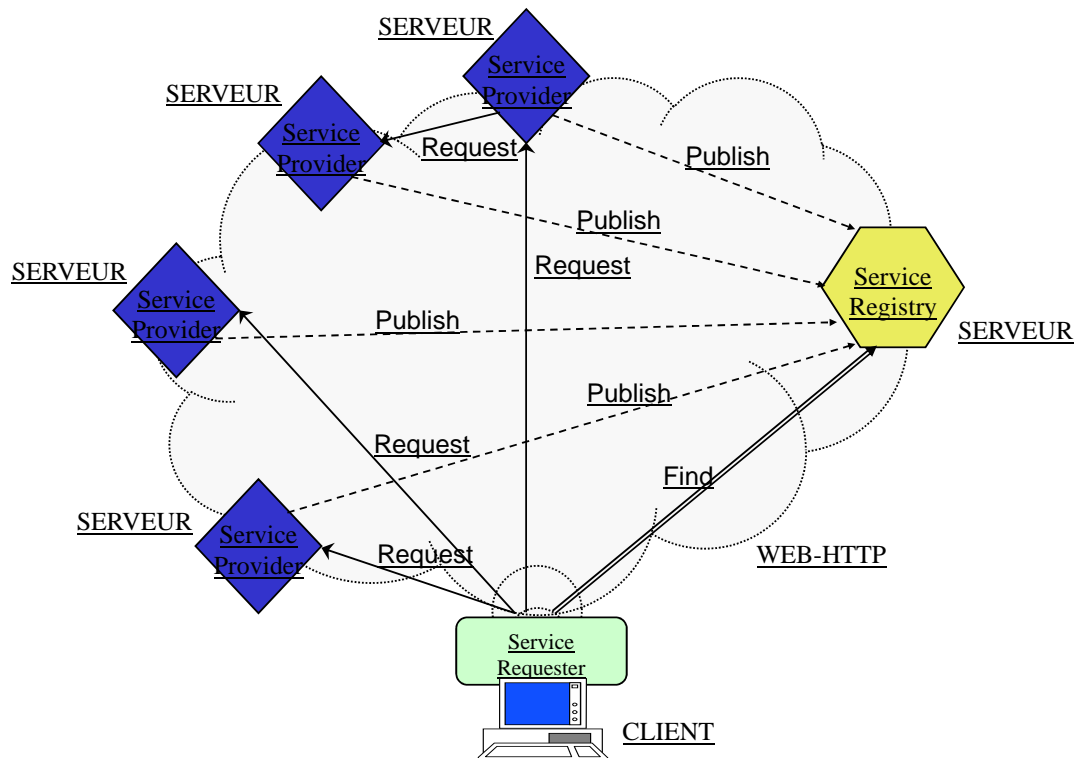
tiers



INTRANET

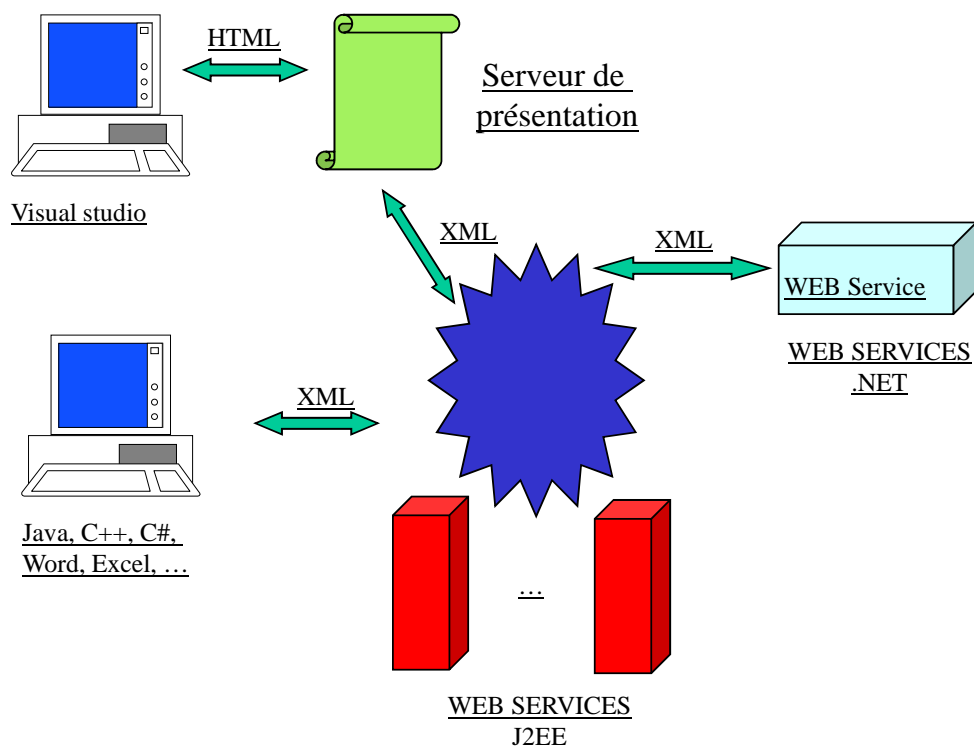
Page 68

Architecture Web Services



Page 69

Architectures inter-opérables



Page 70

Qu'est-ce qu'un WEB SERVICE ?

- Un service Web est une « unité logique applicative » accessible en utilisant les protocoles standard d'Internet
- Caractéristiques:
 - Réutilisable
 - Indépendamment de
 - ◆ la plate-forme (UNIX, Windows, ...)
 - ◆ l'implémentation (VB, C#, Java, ...)
 - ◆ l'architecture sous-jacente (.NET, J2EE, ...)

Qu'est-ce qu'un WEB SERVICE ?

- Une «bibliothèque» fournissant des données et des services à d'autres applications.
- Un objet métier déployé sur le web (vision objet)
- Un « module » ou « composant » (Application avec JAX-RPC : un composant simple avec une interface RMI)

Qu'est ce qu'un Web Service ?

- Définition
 - Application exposée sur le WEB offrant une interface programmable accessible par XML
 - Similaire à un service objet CORBA mais accessible par HTTP / XML
- Exposition
 - Langage WSDL utilisé pour décrire le service
 - Similaire à IDL mais basé sur XML
- Activation
 - Protocole Web au-dessus de HTTP (RPC XML, SOAP)
 - Autres protocoles possibles ...

Page 73

Points communs avec les middlewares objets

Un langage de description : WSDL

Une infrastructure : Le Web et http

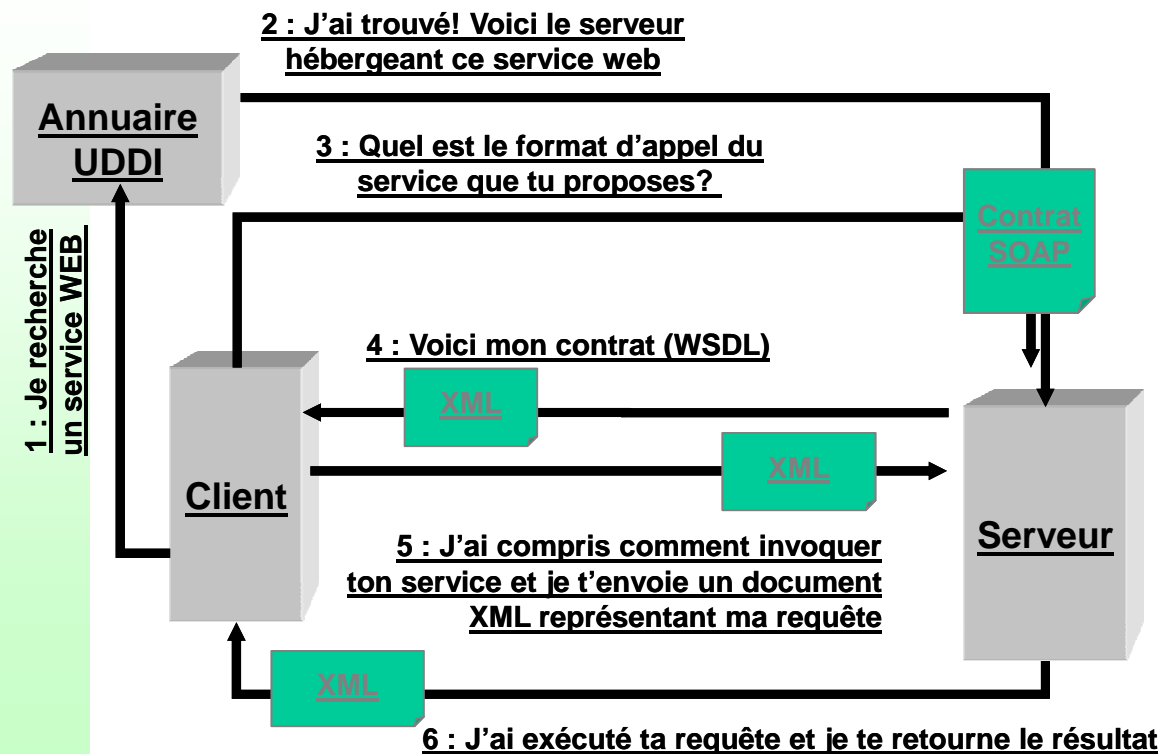
Une communication par envoi de messages : SOAP

Du marshalling : XML

Un service de nommage « dynamique » : UDDI

Page 74

Principe d'utilisation des services WEB



Page 75

Web Services : Place de XML

- Une technologie Description et invocation des services
 - similaire à IDL/IIOP ou DCOM, mais textuel
 - peut être intégrée en sur-couche à l'existant
 - facilite l'interopérabilité et la connaissance
- Encapsulation des protocoles existants
 - échange de données (résultats)
 - échange de requêtes (RPC)
 - contrôle de transactions (AXTP)
- Un protocole du W3C et OASIS
 - SOAP
- Apporte lisibilité, validation, contrôle, interopérabilité

Page 76

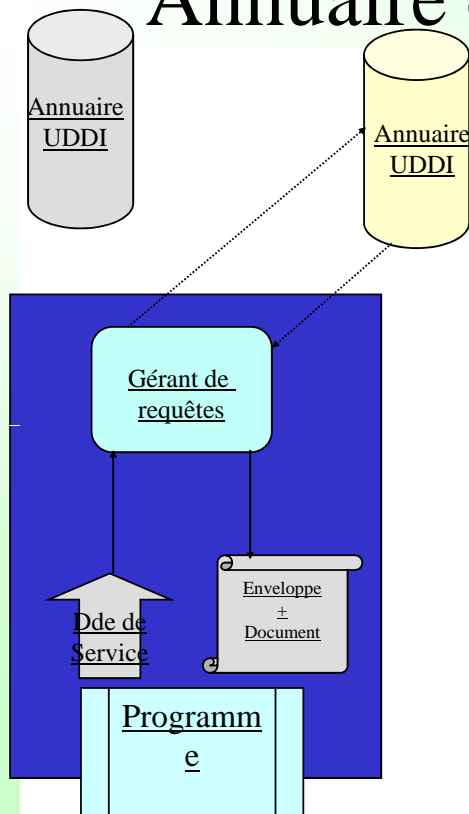
Les Composants

- Service Provider (Fournisseur de service)
 - Application s'exécutant sur un serveur et comportant un module logiciel accessible par IxxNet en XML
- Service Registry (Annuaire de service)
 - Annuaire des services publiés par les providers (UDDI)
 - Géré sur un serveur niveau application, entreprise ou mondial
- Service Requester (Demandeur de service)
 - Application cliente se liant à un service et invoquant ses fonctions par des messages XML (SOAP)

Référentiel UDDI

- Un référentiel UDDI peut être utilisé en :
 - Pages blanches : le référentiel comporte des informations sur les fournisseurs de services.
 - Pages Jaunes : le référentiel comporte des critères de catégorisation de services.
 - Pages vertes : le référentiel comporte des informations techniques (WSDL).
- Le service UDDI est lui-même un WEB service

Annuaire des services: UDDI



Page 79

- Universal Description, Discovery and Integration
- Annuaire des services
 - décrit par un document
 - WSDL, spec. EJB, autre ...
- Accessible en SOAP
- Fonctions
 - Enregistrer votre société
 - Enregistrer des services
 - Enregistrer des opérations
 - Découvrir des services
 - ...

Référentiels UDDI

- Plusieurs types de référentiels
 - Public :
 - Microsoft : uddi.microsoft.com
 - IBM : www.ibm.com/services/uddi
 - HP : uddi.hp.com
 - SAP : udditest.sap.com
 - Privé ou d'entreprise
- Accès défini en WSDL

Page 80

SOA : définitions

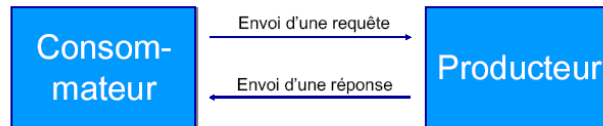
- SOA est un paradigme abstrait, base de l'architecture distribuée sans aucune référence à une implémentation technique
- SOA est une plate-forme formée de Web services
- SOA est une nouvelle plate-forme architecturale pour le développement de solutions distribuées

Définitions (suite)

- SOA représente une architecture ouverte, extensible, fédérée et composable qui promeut une orientation service et qui est composée de services
 - Autonomes
 - Capables de qualité de service
 - Indépendants des vendeurs
 - Intéropérables
 - réutilisablessouvent implémentés sous forme de Web Services (mais pas obligatoirement)

Notion de service et SOA

Une redéfinition des applications, qui se réorganisent en ensembles fonctionnels dénommés services, dont certains sont exposés au monde extérieur, en tant que producteurs de services, pour des consommateurs de services, eux-mêmes services ou applications.



Source: Unilog Management 2005

Un service est un composant informatique délivrant un ensemble cohérent de fonctions sous la forme d'informations échangées via des messages.

SOA (*Service Oriented Architecture* ou Architecture Orientée Services) est un modèle d'interaction applicative qui met en œuvre des **services** (composants logiciels) :

- avec une forte cohérence interne,
- avec des couplages externes « lâches ».

Source: wikipedia

Services et objets

Un service représente une unité autonome de traitement et de gestion de données, communiquant avec son environnement à l'aide de messages ; les échanges de messages sont organisés sous forme de contrats d'échange.

Cette définition emprunte de nombreuses caractéristiques à celle d'un composant objet:

- La combinaison de données et de traitements.
- L'autonomie: un service a sa propre implémentation, son propre déploiement. Il peut être modifié sans que cela affecte les autres services ou partenaires avec lesquels il est en relation.
- Des frontières explicites : on distingue l'intérieur du service, son implémentation, et l'extérieur du service, l'expression sous forme de message de ce qu'il peut échanger et comment il peut être, toujours de l'extérieur, sollicité.

C'est sur ce dernier élément que se joue la différence : par rapport à un composant traditionnel auquel on passe des paramètres, un service échange et fournit de l'information au travers de messages. Chaque service propose un ensemble de messages pouvant être enchaînés pour remplir une fonction déterminée.

Services

- Propriétés d'un service :
 - Un service est autonome
 - Les frontières sont explicites
 - Un service expose un contrat
 - La communication se fait par message

SOA et les WEB SERVICES

- SOA est souvent associé aux standard WEB services :
 - XML
 - WSDL
 - SOAP
 - UDDI

Les prémices des services

RPC (Remote Procedure Call)

Corba (Common Object Request Broker Architecture)

EDA (Event - Driven Architecture, Architecture Pilotées par les Evènements)

EAI (Entreprise Application Integration)

ESB (Enterprise Service Bus)

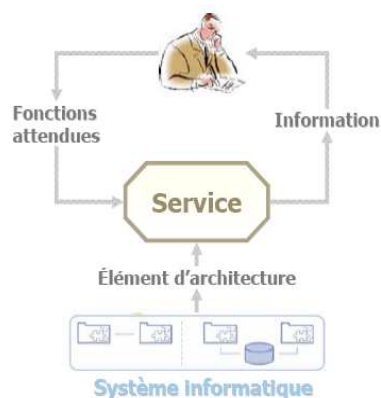
Page 87

Services au sein d'une entreprise

Les consommateurs des services peuvent être différents départements métiers à l'intérieure de l'entreprise, le rôle du **producteur de services** prend le service informatique.

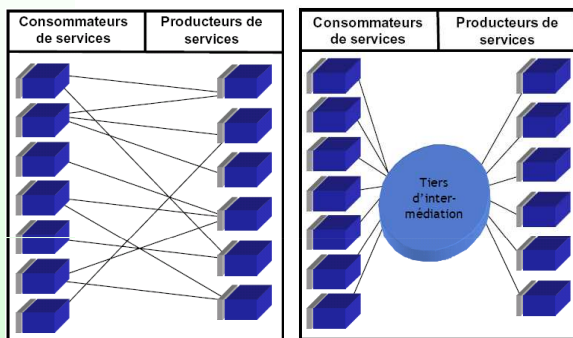
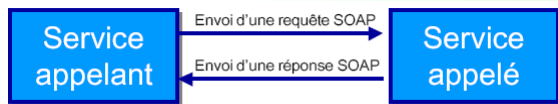
Les départements métiers expriment leur besoin sous forme de listes d'informations qu'ils attendent du système. Ce faisant, ils décrivent les fonctions qui doivent à terme être supportées par un système.

Les informaticiens ont pour tâche de grouper sous forme de service les messages délivrant les informations demandées.



Page 88

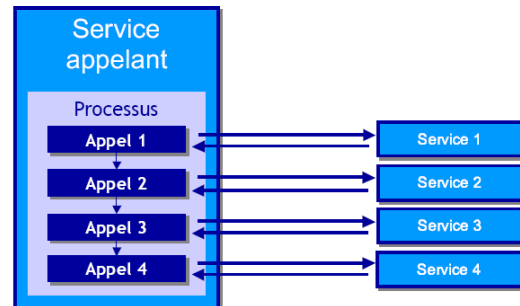
SOA et Processus métier



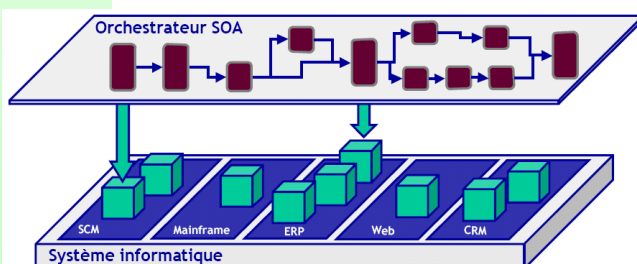
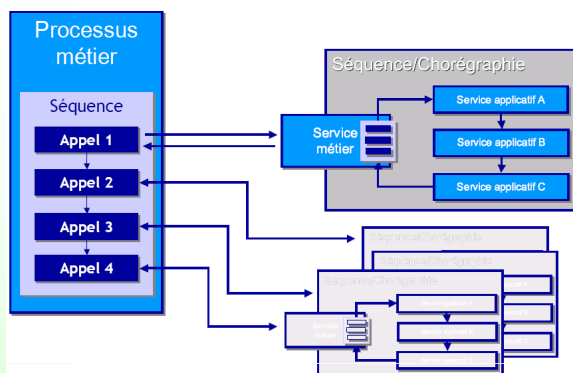
L'intégration d'un ensemble de services pour accomplir une tâche ou implémenter un processus métier.

Le processus métier : c'est l'assemblage de services et/ou de chorégraphies pour décrire des séquences métier fonctionnellement cohérentes.

Pas d'invocation directe de service à service, mais l'utilisation d'un tiers de médiation, qui orchestrera dynamiquement le modèle, sur un plan technique, et assurera un couplage lâche entre les services.



Orchestration



« Chorégraphies »

Introduction des niveaux intermédiaires et pré-assemblage des blocks de services techniques unitaires selon des séquences techniquement et fonctionnellement cohérentes.

Tiers de médiation - Orchestrateur SOA

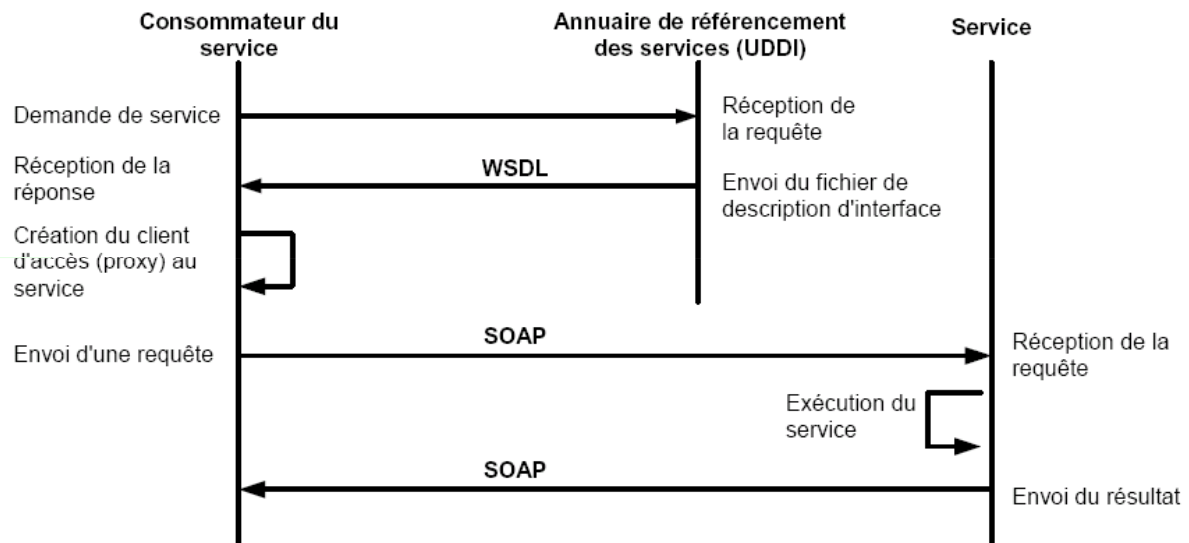
- modélisation/gestion de processus
- gestion de transactions
- gestion de sécurité

...

Tout appel de service étant susceptible de déclencher des appels en cascade et séquentiels d'autres services, le rôle de l'orchestrateur SOA sera de gérer ces enchaînement de services imbriqués de la meilleure façon possible.

A différents niveaux, les processus pourront être modélisés et implémentés depuis l'orchestrateur, qui se chargera d'invoquer les services applicatifs correspondants.

Le schéma de principe



Source: Unilog Management 2005

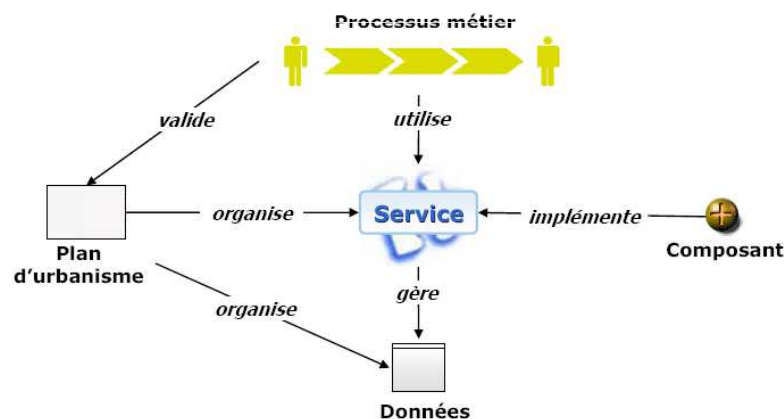
Page 91

SOA et Architecture d'entreprise

SOA - approche multidisciplinaire et multidimensionnelle.

Disciplines concernées:

- la conception orientée objet
- la conception orientée service,
- la structuration de l'information,
- l'analyse des processus métier



Les instruments d'analyse de l'architecture de service

Source: mega 2005

Page 92

Langages

Notation	Execution	Query
BPMN	BPEL	BPQL

Source: Unilog Management 2005

BPMN (Business Process Modeling Notation), concerne la normalisation de la représentation graphique des processus depuis tout atelier graphique;

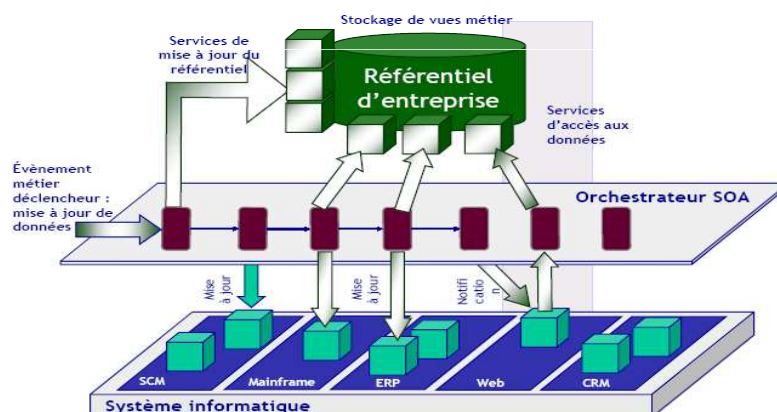
BPEL (Business Process Execution Language), décrit la structure des documents XML où figurent, entre autres informations, les étapes métiers du processus, leur séquence d'enchaînement et les acteurs impliqués;

BPQL (Business Process Query Language), permet l'interrogation et l'extraction de fragments d'information depuis tout document BPEL, pour connaître notamment le statut et l'état d'avancement d'un processus.

Page 93

Référentiels d'entreprise

Référentiels d'entreprise-
vastes ensembles d'objets métier centralisant l'intégralité des informations du modèle public partagé entre tous les départements fonctionnels et techniques.



Répartition Services/Données

Page 94

Source: Unilog Management 2005

SOA et Web Services

SOA \neq Services Web, mais...

Les SOA ont été popularisées avec l'apparition des services Web dans l'e-commerce en mettant en application une partie des principes d'urbanisation.

SOA - un concept d'architecture. Il existait avant l'apparition des services Web.

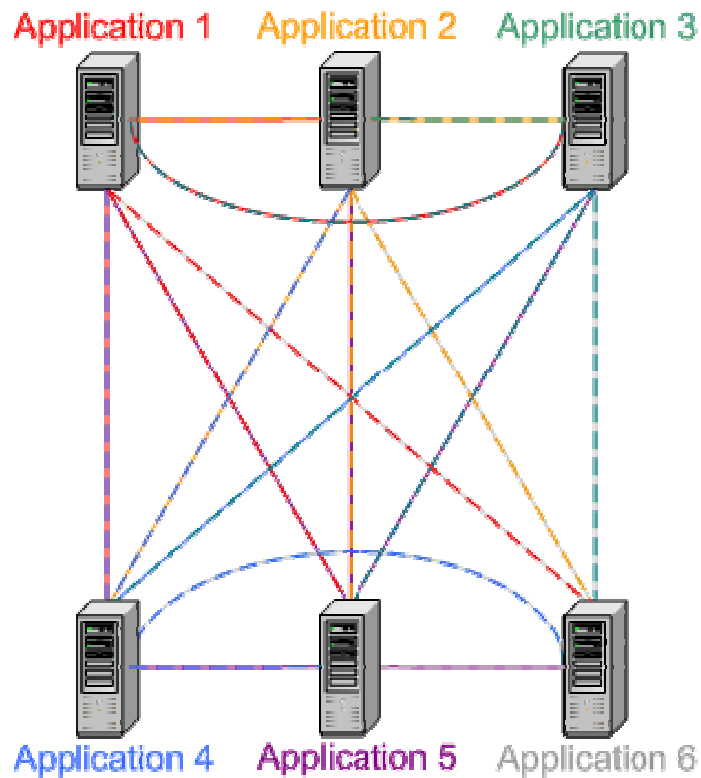
WSOA (WebService Oriented Architecture) est sa déclinaison ou plus précisément son implémentation avec des WebServices, services Web ont standardisé l'accès aux Services pour n'importe quel S.I.

Services Web peuvent cohabiter au sein d'une SOA avec d'autres modes d'implémentation.

Il est possible de s'imaginer une SOA sans aucun service Web!

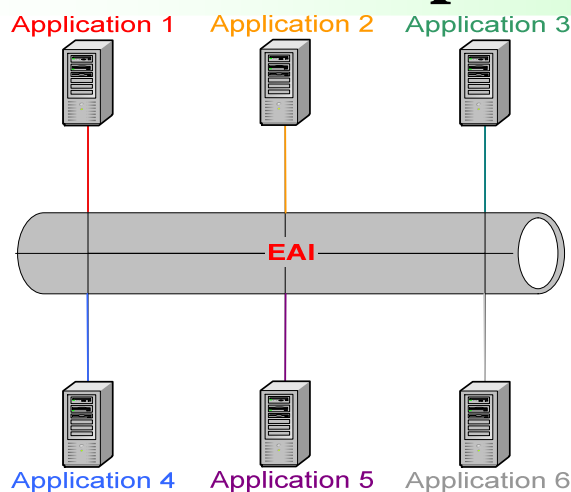
Les EAI Enterprise Application Integration

L'existant : plat de spaghettis



Page 97

Principe de l'EAI



L'EAI est un moyen permettant de passer d'un SI non urbanisé basé sur un modèle de communication point à point à un SI urbanisé basé sur un modèle en bus ou en étoile

Page 98

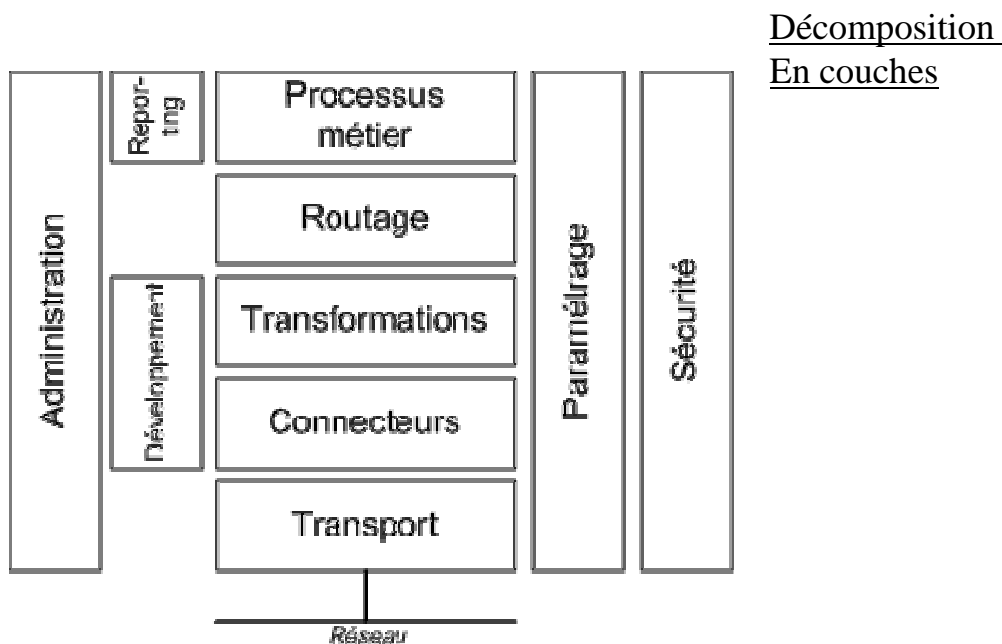
Concepts de l'EAI

- Une offre logicielle et des projets d'intégration d'applications d'entreprise
- Un logiciel EAI permet de modéliser, d'exécuter et d'administrer les flux inter-applicatifs afin de garantir l'intégrité du système d'information à un moment donné
- C'est une « multiprise applicative »

Fonctions de l'EAI

- La fonction connexion :
 - Connexion (non-intrusive)
 - Traitement de l'information indépendant de la plate-forme matérielle
- La fonction routage et transformation :
 - Transport, routage et transformation des données
 - Conversion syntaxique et sémantique des messages
- La fonction gestion des flux et processus métiers :
 - modélisation des activités de l'entreprise ainsi que leurs relations
 - Synchronisation des flux, bufferisation...
- La fonction administration des flux et des processus :
 - maintenance, analyse des performances, sécurité, QoS...

Aspects techniques



Remarque : les couches EAI se positionnent au dessus de la couche 4, la couche Transport, du modèle OSI.

Page 101

EAI : un système en couches

La couche **Connecteurs** a pour responsabilité de connecter les applications à la plate-forme d'EAI. Les connecteurs utiliseront les services de la couche Transport pour extraire et/ou injecter des données depuis ou vers les applications en relation avec l'EAI.

La couche **Transformations** fournit des services de conversion de données afin de permettre les dialogues entre application(s) source(s) et cible(s).

La couche **Routage** assure la gestion des flux de messages entre les applications. Elle route les messages de la source vers la ou les applications cibles.

La couche **Processus métier** gère la logique de l'enchaînement des tâches constituant un processus. Elle contrôle l'exécution des processus métier sans intervention humaine ainsi que les workflow (processus métier comprenant une intervention humaine).

Page 102

EAI : un système en couches

La couche **Paramétrage** sert à adapter le comportement des couches Transport, Connecteurs, Transformations, Routage et Processus métier pour répondre aux besoins de projets d'intégration.

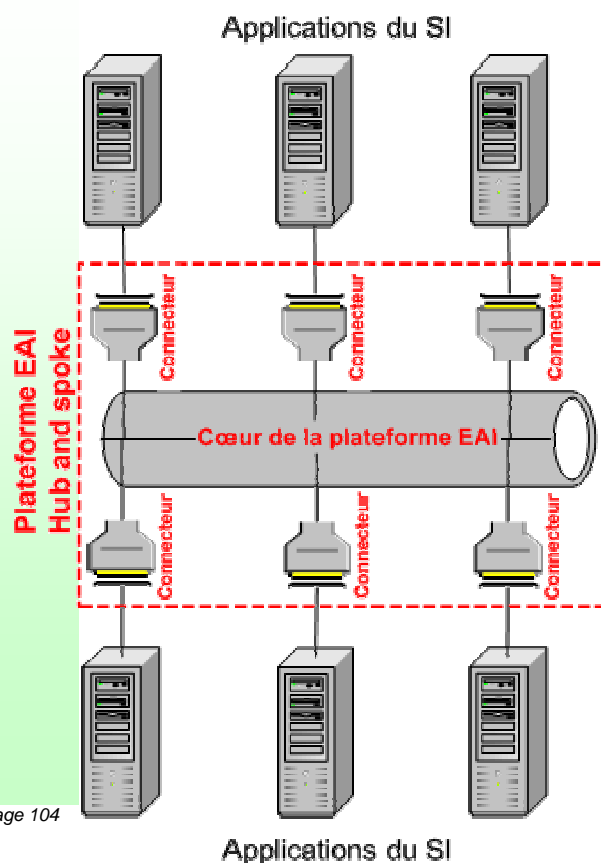
La couche **Développement** donne la possibilité d'ajouter des fonctionnalités à la plateforme EAI en terme de services rendus.

La couche **Administration** permet, de gérer l'ensemble de la solution EAI, tant au niveau technique que fonctionnel.

La couche **Reporting** fournit des informations décisionnelles sur les flux de données métier qui transitent par l'EAI.

La couche **Sécurité** traite les problématiques d'authentification, d'intégrité, de confidentialité et de non répudiation des messages transitant par l'EAI, ainsi que la gestion des accès à la plateforme d'EAI.

Structure HUB and Spoke



L'architecture Hub and Spoke (moyeu et rayon)

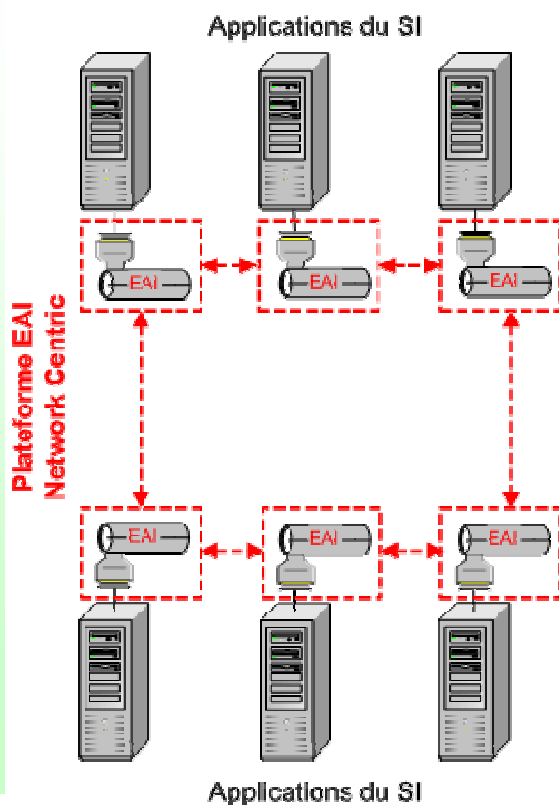
EAI centralisée dans laquelle des branches sont reliées via des connecteurs à un bus unique dont le rôle est d'aiguiller les communications.

Architecture en étoile : le hub (concentrateur) représente le nœud central et les spoke (rayons) permettent de relier les applications au nœud central.

Structure Hub and Spoke

- Avantages :
 - modification minimale dans les applications,
 - une centralisation des règles de routage et de transformations (au sein du hub),
 - une administration simplifiée
- Inconvénients majeurs :
 - le point de faiblesse qu'introduit le hub en cas d'indisponibilité
 - la gestion de la montée en charge qui ne peut se faire que par ajout de hubs supplémentaires.
 - Exemple : webMethods est le représentant le plus populaire de cette architecture. Dans la terminologie webMethods, l'Integration Server représente le hub et les applications connectées au hub représentent les spokes.

Structure Network Centric



Un tel système centré réseau est constitué de plusieurs machines, connectées entre elles.

La charge du système se répartit ainsi sur l'ensemble des processeurs du réseau. C'est donc l'inverse d'un système centralisé.

Pas de Nœud central

Structure Network Centric

- Avantages :
 - une distribution de la charge sur les différents nœuds,
 - une distribution du référentiel sur les différents nœuds,
 - une capacité de support de l'augmentation de la charge grâce à l'absence de goulot d'étranglement central.
- Inconvénients :
 - une administration difficile du fait de la décentralisation des composants,
 - une interpénétration dans les applications plus importante que pour le mode Hub and Spoke car l'intelligence de l'EAI se déporte sur chaque nœud.
- Exemple : ICAN de SeeBeyond est basé sur ce concept.

Les connecteurs

- Les connecteurs permettent aux applications de se brancher sur le bus de communication fourni par la solution d'EAI.
- fournissent le middleware applicatif entre l'application à connecter et la plateforme d'EAI.
- Différents types de connecteurs existent :
 - des connecteurs « fonctionnels » (par exemple connecteurs vers SAP ou PeopleSoft),
 - Des connecteurs techniques (par exemple drivers ODBC ou JDBC),
 - des connecteurs spécifiques (par exemple connecteurs pour l'accès à des applications propriétaires).

Standards connecteurs

Type	Dénomination	Description	Domaine d'utilisation
Fonctionnel	Connecteur SAP	Accès à l'ERP SAP au niveau des BAPI (en synchrone) ou avec ALE (en asynchrone)	Accès aux différents modules de SAP
	Connecteur PeopleSoft	Accès à l'ERP PeopleSoft au niveau des Component Interfaces	Accès aux différents modules de PeopleSoft
	Autres connecteurs ERP		
	Il convient de vérifier quelles versions de l'ERP le connecteur supporte		
Technique	Fichier	Fichier plat et fichier XML	Le connecteur fichier est par définition universel. XML tend à s'imposer comme format.
	ODBC	Open DataBase Connectivity	Middleware d'accès aux SGBDR
	JDBC	Java Database Connectivity	Middleware Java d'accès aux SGBDR
	Accès natif aux SGBD	Protocole propriétaire d'accès au SGBD	SQL*Net par exemple pour Oracle
	JCA	Java Connector Architecture de Sun	Middleware Java d'accès aux ressources du SI depuis le monde J2EE
	JMS	Java Messaging Service de Sun	Middleware Java d'accès aux MOM [1]
	Autres API J2EE		
	EDI	Connecteur supportant un ou plusieurs protocoles et formats EDI	Par exemple connecteur ETEBAC3
	Client Web Services	Accès aux Web Services ou exposition de Web Services	Tend à devenir un connecteur universel
	Connecteur vers un moniteur transactionnel	Accès à des moniteurs transactionnels comme CICS ou Tuxedo	
	L'utilisation de XML et particulièrement des Web Services est à privilégier		
Spécifique	Connecteur mainframe	Accès à des applications fonctionnant sur mainframe	
	Connecteur application propriétaire	Accès au niveau données ou applicatif à des applications propriétaires du SI	
Pour éviter de passer par un connecteur propriétaire (et donc par un développement ad hoc), l'échange de fichiers est une solution éprouvée			

Page 109

Les principaux acteurs de l'EAI

- WEBMETHODS : Enterprise Services Platform
- SEEBEYOND : Ican Suite
- MICROSOFT : Biztalk server
- IBM : Websphere Business Integration Server
- BEA : Weblogic Integration Server
- ORACLE

Page 110

Les processus métiers

- Modélisation de processus métiers
 - Enchaînement d'activités
 - Echange de messages XML
- Intégration des Services Web
 - API standards décrites en WSDL
 - Langages d'orchestration (WFSL, XLANG, BPEL, ...)
- Interpréteur de ces langages
 - Généralement centralisé
 - Pilote les processus et échanges

Page 111

Exemple de processus métier

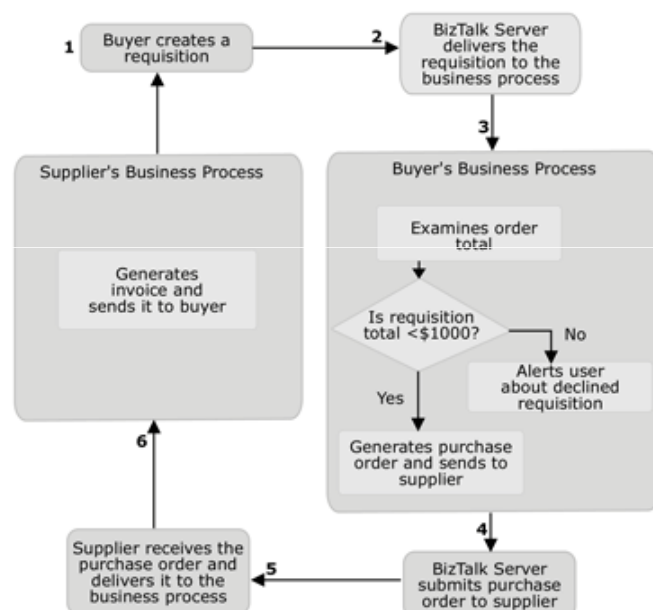
➤ Modélisation d'une activité commerciale

- Process Buyer
- Process Supplier

➤ Génération et échange de message XML

➤ Tests et contrôles

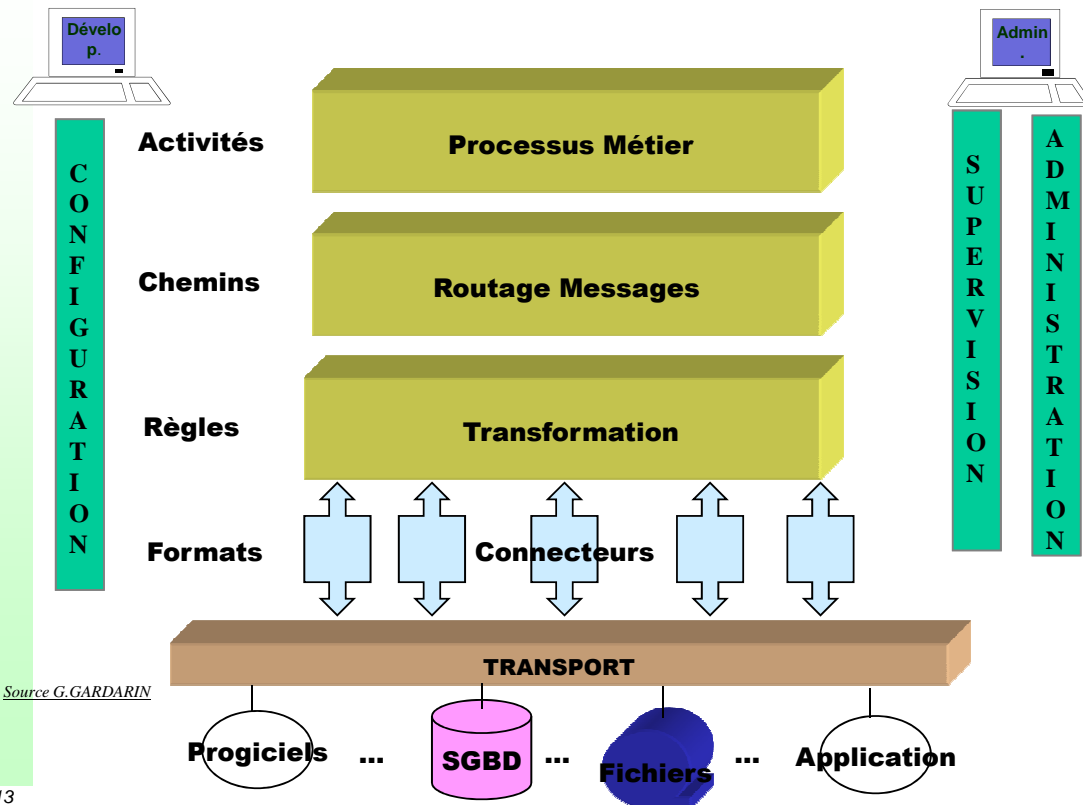
- Total < \$1000



Source G.GARDARIN

Page 112

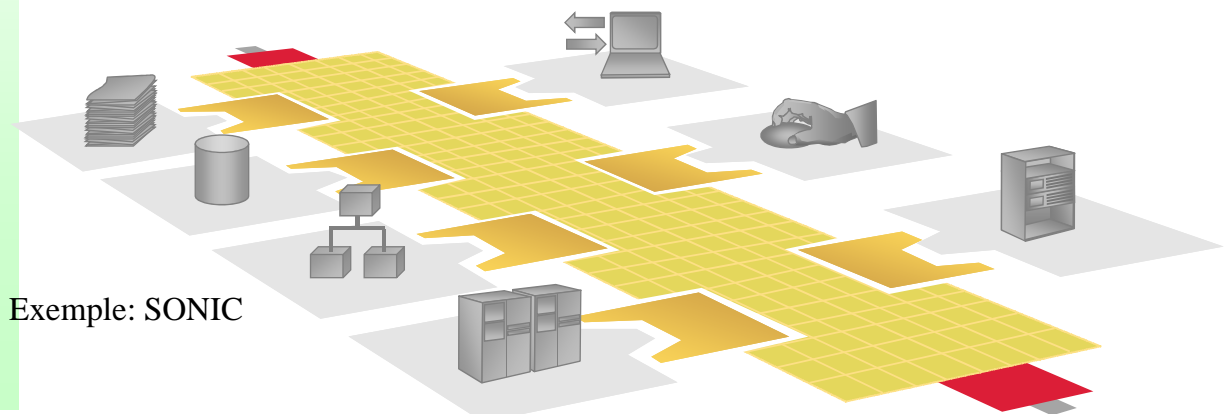
Architecture fonctionnelle



Page 113

L'avenir : Enterprise Service Bus (ESB)

EAI à prises Web services
Bus de services distribués
Backbone pour messagerie fiable
Routage intelligent basé contenu
Coordination des processus
Cadre sécurisé
Transformation XML



Page 114

Bibliographie

- Christophe Longépé
Le projet d'urbanisation du système
d'information, 2ème édition Dunod
- Urbanisation et BPM
Yves Caseau Dunod
- Solutions pour l'EAI
Devin Spackman et Mark Speaker
Microsoft Press
- Enterprise Service Bus
David A Chappell O'REILLY