



Gestion Projet : Cours 2

Le Système d'Information

« Ensemble d'acteurs humains et/ou applicatifs en interaction les uns avec les autres ayant pour but de traiter, diffuser, persister l'information afin de supporter les activités métiers des utilisateurs »



Sommaire

- L'urbanisation 3
- Découpage du SI : Niveau 1 11
- Les projets informatiques 19



L'urbanisation

Un constat

- 3 acteurs :
 - une agence de voyage
 - Une Hotline de vente par téléphone
 - Un service achat : gère les stocks disponibles
 - Un service facturation : vérifie la comptabilité
- Un processus :
 - vente d'un voyage à un client
- Constat
 - 4 bases clientes
 - Des incohérences entre les différents services
 - Des données au format différent :
 - Au niveau compta : client sur 20 caractères contre 15 dans l'agence
 - Difficultés pour croiser les données :
 - agrégation reporting

Pourquoi ?

- Réduire les coûts des évolutions :
 - Réduire les impacts transverses
 - Ne pas refaire ce qui existe déjà
- Réduire les délais
 - Savoir qui gère quoi
- Améliorer la qualité
 - Cohérence des données
 - Meilleure maîtrise du bout en bout (processus métier)
- Définition de Design Pattern
- Rendu possible grâce aux évolutions techniques

Le point de départ

- Définir le périmètre :
 - Il est impossible d'urbaniser complètement un SI
 - Peut être appliqué à chaque niveau du SI
- Notion de processus métier
 - Refonte du processus : bon point de départ
 - Exemple : UML et les Use Case
 - Découpage en objectifs et sous-objectif :
 - Verbe + objet
 - Exemple : saisir dossier, contrôler adresse, ...
- Notion de KPI :
 - indicateur clé de performance
 - A un objectif est associé un ou plusieurs KPI



MIM

Modèle des informations métier

- Assurer que le langage est standardisé de manière à faciliter le partage, la consolidation et la réalisation de l'information
- Documenter : les règles de gestion et faciliter la traçabilité de leurs implémentations
- Connaître les données : tracer leur traitements et mesurer leur qualité
- Pourvoir un cadre de référence pour évaluer les progiciels ou construire des solutions spécifiques qui répondent aux besoins

Les référentiels

- Si au moins 2 SF gèrent le cycle de vie d'un objet métier, on doit créer un référentiel fonctionnel responsable du cycle de vie de ces objets
- Définition des classes concepts :
 - Niveau d'abstraction le plus élevé
 - Exemple : le client qui est une abstraction des notions de coordonnées bancaires, adresse ...
- Contiennent les fonctions primaires de consultation et de gestion du cycle de vie des données :
 - Consulter, créer, Modifier, Supprimer
- Assurent la cohérence globale de leurs données grâce à l'exposition de services transactionnels :
 - Atomicité, Consistance, Isolation, Durabilité (ACID)

La médiation

- Les Systèmes ne communiquent qu'au travers de leurs interfaces. Chaque interface correspond à une et une seule fonction
 - Une interface appartient donc à un seul SF
- La zone de médiations à pour rôle :
 - De fournir la localisation des appels de service (annuaire UDDI)
 - Gestion des messages (annuaire de diffusion)
- Notion de SOA



BPM

Business Process Manager

- Permet d'automatiser des processus métier
- Apporte une meilleur maîtrise et un meilleur suivi des données
 - Traçabilité
 - Reprise en cas d'erreur
 - Cohérence des données
- Complexité au niveau de la mise en place
 - Obligation d'avoir un SI orienté urbanisation et SOA
 - Gestion des indispos :
 - File d'attente
 - Erreurs techniques



Découpage du SI : Niveau 1

Le Front-Office : Définition

- Ensemble d'applications permettant de s'interfacer avec le SI
- Interactions :
 - B2B : Business To Business
 - B2C : Business To Consumer
 - B2A : Business To Administration
 - B2E : Business To Employees
- Interfaces :
 - Webservice
 - Client lourd
 - Client léger
- Exemples :
 - Guichet automatique dans une banque
 - Site de vente en ligne

Le Front-Office : Impacts Projet

Client	Client final généralement non présent.
Disponibilité	Importante voir indisponibilité interdite
Technologie	Dernières nouveautés
Durée de vie	Courte : 1 à 2 ans
Sécurité	Important car porte d'entrée sur le SI
Évolutions Fonctionnelles	Nombreuses donc devant être peu coûteuses

Le Back-Office : Définition

- L'arrière boutique :
 - accès interdit depuis l'extérieur du SI
- orienté traitement de données :
 - Exemple : facturation, logistique
 - Traitement en masse des données
- mise en place de référentiel d'entreprise
 - base cliente
 - base des offres (catalogue)

Le Back-Office : Impacts Projet

Client	Client Interne
Disponibilité	De courtes indisponibilités sont tolérées tant que les traitements respectent leur temps d'exécution
Technologie	Dernières nouveautés
Durée de vie	longue : 5 à 20 ans
Sécurité	Faible
Évolutions Fonctionnelles	Par lot, afin d'éviter le nombre de mise en production

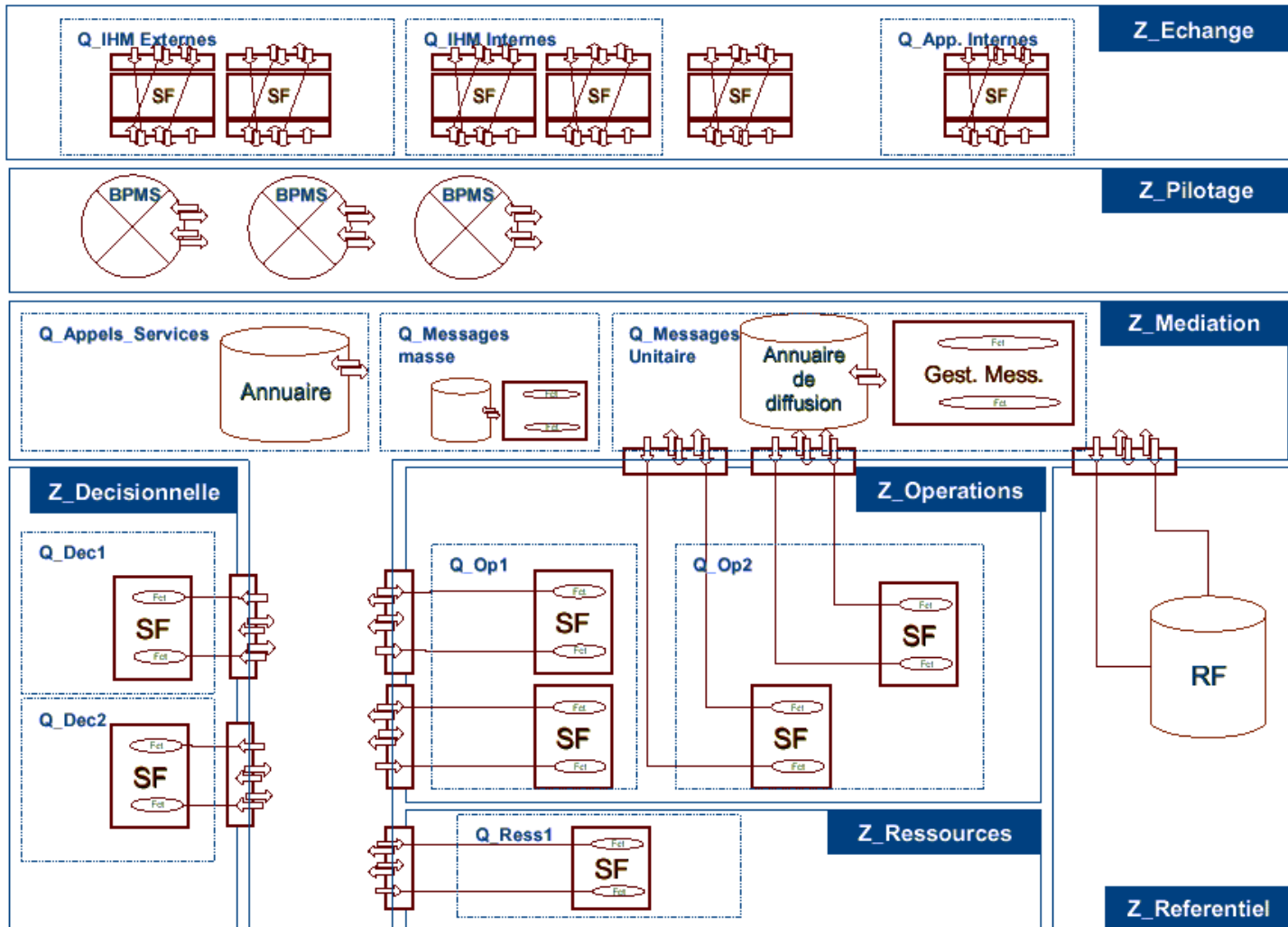
Le Middle-Office : Définition

- Couche Transverse
 - Permet aux différents système du SI de dialoguer
- Orienté contrôle et enrichissement
 - Assure la cohérence de données entre l'ensemble des système
- Vision processus
 - Vue métier

Le Middle-Office : Impacts Projet

Client	Autre application
Disponibilité	Peut-être indisponible s'il ne bloque pas le front et si l'indisponibilité n'entraîne pas de perte de données
Technologie	Toutes
Durée de vie	5 à 10 ans
Sécurité	ST interne sécurité classique
Évolutions Fonctionnelles	Liés au processus. Évolutions liées à celles des STs clients

Découpage final





Les projets informatiques

Nouvelle Application (1)

- Devient assez rare si l'entreprises a un SI avancé au niveau du front-office ou du back-office.
- Ne pas négliger les phases de formation que ce soit au niveau des utilisateurs finaux qu'au niveau des exploitants
- Difficulté d'estimer les futures évolutions (volumétrie des données, nombre d'utilisateurs...) qu'il faudrait prendre en compte dès la conception de la première version
- Les premières phases d'étude et d'analyse sont les plus critiques : exemple de la balançoire vu lors du premier cours

Nouvelle Application (2)

Implication du client	Très présent
Coût	Difficile à gérer
Délais	Client généralement assez compréhensif sur ce point
Qualité	Doit être au rendez-vous si l'on souhaite que cette application évolue (sinon risque de refonte)
Fonctionnalité	Nombreuses, souvent trop nombreuses, et pas forcément utiles.
Test	Total et très long car il faut tester chaque cas d'utilisation de chaque fonctionnalité
Composante Technique	Généralement nouvelle application = nouvelle infrastructure.
Composante Fonctionnelle	Le piège.

Les évolutions Fonctionnelles (1)

- Projets les plus courants
- Cycle de réalisation devant être très court
- Dans certains cas regroupement d'un ensemble d'évolutions fonctionnelles cohérentes les unes par rapport aux autres pour limiter les impacts sur l'application
- Exemple :
 - Front : Ajout d'une nouvelle offre, d'un nouveau champs de saisie
 - Middle : Ajout de nouveau contrôle pour éviter des cas de fraude
 - Back : rajout d'information cliente au niveau des factures

Les évolutions Fonctionnelles (2)

Implication du client	Client présent car à l'origine du besoin
Coût	Dépend de l'évolution demandée. Si le coût est trop important la refonte est souvent proche
Délais	Une évolution fonctionnelle est souvent liée à un besoin commercial
Qualité	Ne doit pas impacter le niveau de qualité actuelle de l'application
Fonctionnalité	Généralement peu nombreuse et émise sur une application existante, il est rarement possible de réduire le périmètre
Test	Au minimum tests de non régression en plus des tests sur la nouvelle fonctionnalité
Composante Technique	Faible hors impacts perf
Composante Fonctionnelle	Classique analyse des nouveaux besoins

Le palier technique (1)

- But : pérenniser une application qui voit son infrastructure technique ou logiciel devenir obsolète
- Effet boule de neige
 - À l'origine migration XP
 - Au final : changement de version oracle, de Delphi, ...
- S'applique aux applications réellement appréciées par les clients
- Projet souvent négligé car non porté par les clients
- Projet qui permet de faire vivre de nombreux informaticiens (merci Microsoft)
- Possibilité d'intégrer du nettoyage applicatif à ce type de projet
 - suppression des fonctionnalités non utilisées

Le palier technique (2)

Implication du client	Nulle si l'on considère que le client est l'utilisateur de l'application
Coût	Le budget est limité car il y a rarement un gros ROI associé à ce type d'opération
Délais	Pas prioritaire
Qualité	Au minimum identique à ce que le client connaît
Fonctionnalité	Il n'est pas recommandé d'ajouter des nouvelles fonctionnalités lors de la réalisation d'un palier techno important
Test	Performance, non régression
Composante Technique	Forte, c'est souvent une personne ayant un profil technique qui gère ce type d'application
Composante Fonctionnelle	Nulle hormis pour la partie teste de non régression. Peut être utile si l'on décide de faire du nettoyage applicatif

Les Refontes (1)

- A partir d'une ou de plusieurs applications existantes réaliser une nouvelle application
- Découle d'une analyse du ROI :
 - Maintenance de l'application trop coûteuse
 - Différence entre la réalisation d'un palier technologique et une refonte pas suffisamment importante
 - Impossibilité de prendre en compte de nouveaux besoins avec un coût de réalisation acceptable
- Peut découler d'un rapprochement de services, d'entreprise, ...

Les Refontes (2)

Implication du client	Très aléatoire. Tout dépendra des raisons qui poussent à la refonte
Coût	Lié au ROI
Délais	Contrainte la plus faible car des applications couvrent déjà en partie le besoin.
Qualité	Au minimum équivalent à l'existant. Souvent devant être meilleure.
Fonctionnalité	La refonte se justifie souvent pas l'ajout de nouvelles fonctionnalités. Elles peuvent être intégrées dans un deuxième lot (voire évolution fonctionnelle)
Test	Mêmes tests que pour la création d'une nouvelle application
Composante Technique	Mêmes impacts que pour la réalisation d'une nouvelle application
Composante Fonctionnelle	Analyse de l'existant pour ne pas perdre des fonctionnalités lors de la refonte

Les Projets Transverses (1)

- Impactant plusieurs applications
- Plusieurs clients
 - Obligation de définir un client principal qui sera le porteur de l'aspect financier
- Pouvant aller du front-office jusqu'au back-office
- Évolution fonctionnelle impactant un processus de l'entreprise (urbanisation)

Les Projets Transverses (2)

Implication du client	Le client principal est impliqué. Par contre les utilisateurs qui subissent les effets de bords ne sont pas forcément moteur
Coût	Gros budget, grosse équipe. La moindre rallonge est importante
Délais	Il est difficile de prendre du retard car on peut bloquer d'autres évolutions
Qualité	
Fonctionnalité	Supprimer en priorité les fonctionnalités qui impact fortement le plus de ST
Test	essentiellement de la non régression.
Composante Technique	Faible hors impacts perf
Composante Fonctionnelle	Classique analyse des nouveaux besoins

Le Décisionnel (1)

- Application permettant aux utilisateurs de consulter et d'agréger des données appartenant à d'autres applications
- Exemple :
 - Connaître le pourcentage de panne sur un modèle bien précis par rapport à la vente de ce modèle
- Mode d'alimentation asynchrone :
 - Alimenté par les autres STs
 - Pas d'interrogation direct des données métiers
- Application Back-Office pouvant remonter des informations au front-office

Le Décisionnel (2)

Implication du client	Exprime des besoins de reporting, une fois ce besoin émis le client est moins présent
Coût	Peu de budget
Délais	Assez souple
Qualité	Grosse contrainte. La politique commerciale de l'entreprise est basée sur les informations fournies
Fonctionnalité	Les fonctionnalités ne sont pas nombreuses. C'est surtout le nombre de données à agréger qui augmente
Test	Test de validation
Composante Technique	Orienté base de données décisionnelle.
Composante Fonctionnelle	Forte car il faut identifier les STs pouvant fournir les informations à agréger

Les Progiciels (1)

- Difficile d'associer une démarche urbanisation avec les progiciels
- Ne peut intervenir que dans les cas :
 - Nouvelle application
 - Refonte application
- Des progiciels différents en fonction du métier cible
- Solutions clef en main : éviter le spécifique
- Exemple :
 - Siebel : CTI (centre d'appel), force de vente
 - SAP : logistique, paye

Les Progiciels (2)

Implication du client	Le client doit connaître le progiciel ou avoir déjà travaillé avec le progiciel. C'est au client de s'adapter au progiciel et par l'inverse.
Coût	Les licences associés à ce genre de logiciel sont élevés. L'obligation de passer par des développeurs spécialisés est également à prendre en compte
Délais	Comme la plupart des fonctionnalités sont déjà implémentées un gain au niveau des délais est attendu
Qualité	Si on limite le spécifique les anomalies sont mineurs (hors anomalies du progiciel lui même)
Fonctionnalité	Nombreuses et facile à prendre en compte tant qu'elles sont fournis avec le progiciel
Test	Phase orientée fonctionnelle + test de perf
Composante Technique	Infrastructure imposée par le progiciel
Composante Fonctionnelle	Il est impératif de connaître les possibilités du progiciel