

Université Paris Ouest Nanterre La défense
Master M1 MIAGE : Mémoire de fin d'année
Maître de Stage : Jean-Claude BONCOEUR
Tuteur universitaire : Emmanuel HYON

Mémoire Master M1

Utilisation de l'Urbanisme des SI pour la recette des applications

Stéphanie BAUCHE

Année 2009-2010



Remerciements

Je tiens à remercier l'ensemble des personnes qui, de près ou de loin, ont participé à la réalisation de cette mission, et notamment :

- M. Christian DENIS, chef du pôle EFG, pour son accueil.
- M. Jean-Claude BONCOEUR, pour son accueil, sa confiance, sa disponibilité, et son implication dans mes travaux.
- Toute la cellule DSIT-EFG/URB (Martine BESSIERE, Christophe DUEZ, Vanontiana RAJOELINA et Bahi AYAD), qui m'ont parfaitement intégré dès le début, et qui n'ont pas hésité à me transmettre leurs connaissances.
- Tous ceux qui m'ont aidé pour mon étude : Paulo DE OLIVEIRA, Lydie BRANLARD, ..., et d'autres.
- M. Stéphane DUCORNET de la société Dalisys, qui a assuré ma formation aux API QUALITY CENTER.
- M. Sergueï MILECHINE de la société MEGA, qui a assuré ma formation à MEGA API.
- M. Emmanuel HYON mon tuteur enseignant, qui a su veiller au bon déroulement de mon apprentissage au sein de la SNCF et qui m'a conseillé pour ce mémoire.
- Le CFA AFIA et l'université de Nanterre qui m'ont permis de faire ce Master MIAGE.



Utilisation de l'Urbanisme des SI pour la Recette d'applications

Introduction

Depuis quelques années, la plupart des entreprises se sont engagées dans la voie de l'urbanisation de leurs Systèmes d'Informations (SI), dans le but de mieux les organiser et les optimiser. Le retour d'expérience a permis de définir et d'ouvrir les possibilités de l'urbanisme.

En définitive, les données résultantes de l'urbanisation du Système d'Information forment un référentiel. Ce référentiel est un ensemble de bases de données contenant les « références » d'un système d'information. Il doit donc être clair, logique et précis. Enfin, les applications de l'entreprise se basent sur ce référentiel pour utiliser les données du Système d'Information. Il est donc stratégique.

Dans le cycle de vie d'un projet, les exigences relatives à l'application sont définies dans le référentiel d'urbanisme durant la phase d'analyse des besoins. Elles doivent ensuite, et en théorie, être vérifiées lors de la phase de tests. Nous constatons néanmoins qu'il n'y a pas de lien concret entre toutes ces données.

D'autre part la recette d'application utilise les données métiers du système d'informations pour créer les exigences de tests. Il est donc important de créer une synergie plus forte entre l'urbanisme du SI et la recette d'applications. Mon étude est centrée sur cette problématique.

En un premier temps, nous définirons avec plus de précision ce que sont l'Urbanisme des SI puis la Recette d'applications.

Ensuite, nous confronterons les données pour en déduire des passerelles. Ces liens seront à la source de l'application que j'ai développée durant ce stage et que je vous propose de détailler en dernière partie.

Développée pour la SNCF, je vous propose tout d'abord la fiche d'identité de cette société que l'on ne présente plus.

Sommaire

Introduction	1
I. L'entreprise.....	4
1) <i>Structure de la SNCF</i>	<i>4</i>
2) <i>La DSIT (Direction des Systèmes d'Information et des télécommunications).....</i>	<i>4</i>
3) <i>Contexte actuel.....</i>	<i>5</i>
4) <i>La cellule urbanisme de DSIT-EFG.....</i>	<i>5</i>
II. Présentation de la mission	6
1) <i>Problématique</i>	<i>6</i>
2) <i>Objectif de la mission.....</i>	<i>6</i>
3) <i>Champs d'actions.....</i>	<i>6</i>
III. L'Urbanisme des SI.....	7
1) <i>Qu'est ce qu'un système d'information ?.....</i>	<i>7</i>
2) <i>Définitions de l'urbanisme</i>	<i>8</i>
3) <i>La cartographie.....</i>	<i>9</i>
4) <i>L'outil de modélisation.....</i>	<i>9</i>
IV. La Recette d'applications	13
1) <i>Définition.....</i>	<i>13</i>
2) <i>L'outil de recettes.....</i>	<i>14</i>
V. Le lien entre l'Urbanisme SI et la Recette	16
1) <i>Un référentiel commun.....</i>	<i>16</i>
2) <i>Utilisation de l'Urbanisme SI pour la Recette d'applications.....</i>	<i>16</i>
3) <i>Les données de MEGA</i>	<i>17</i>
a) <i>Les données utilisées</i>	<i>17</i>
b) <i>Description des données.....</i>	<i>17</i>
4) <i>Les données de Quality Center.....</i>	<i>18</i>
a) <i>Les données</i>	<i>18</i>
b) <i>Description des données.....</i>	<i>18</i>
5) <i>Correspondances MEGA et QC</i>	<i>19</i>
VI. La construction d'un outil	22
1) <i>Les étapes</i>	<i>22</i>
2) <i>Etudes des différentes solutions</i>	<i>22</i>
3) <i>Comparaison des solutions</i>	<i>24</i>
4) <i>La solution retenue.....</i>	<i>24</i>

a) Généralités.....	24
b) Conception	25
c) Choix techniques	27
d) Limites.....	28
e) Axes d'amélioration	28
Conclusion.....	29
a) Intérêts professionnels.....	29
b) Intérêts personnels.....	29
Abréviations.....	30
Références bibliographiques	31

I. L'entreprise

1) Structure de la SNCF

La SNCF (Société Nationale des Chemins de Fer) est l'une des principales entreprises publiques françaises. Ses activités sont centrées sur le transport ferroviaire de voyageurs et de marchandises, sur la gestion d'infrastructures ainsi que sur l'ingénierie ferroviaire.

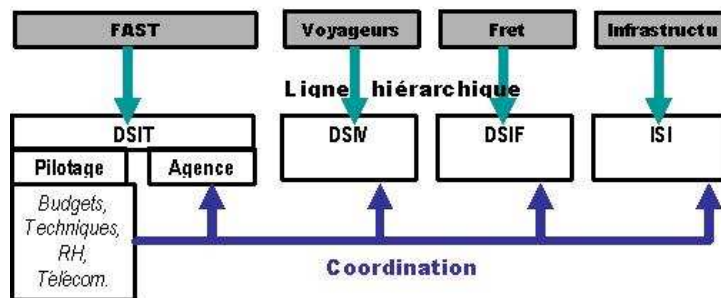
Forte de son chiffre d'affaire conséquent (24,9 milliards d'euros en 2009), la SNCF est actuellement présidée par Guillaume Pepy. Elle compte environ 201 000 salariés répartis sur 23 régions et 330 établissements.

2) La DSIT (Direction des Systèmes d'Information et des télécommunications)

En 2009, la SNCF a dépensé environ 500 millions d'euros pour ses systèmes d'information, soit 2 % de son chiffre d'affaires.

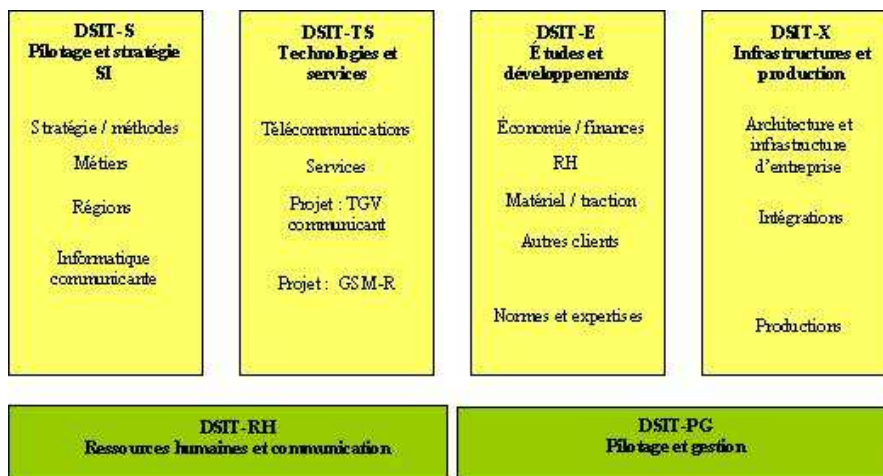
La mise en œuvre de ces SI se fait au sein de 4 directions (cf. schéma suivant) :

- 3 entités rattachées chacune à une activité de l'Entreprise, dont elles portent le nom : Voyageurs, Fret et Infrastructure,
- la Direction des Systèmes d'Information et des Télécommunications (DSIT), est rattachée à la Direction des Finances, des Achats et des Systèmes d'information et de Télécommunications (FAST).



Structure des Systèmes d'Information de la SNCF

La DSIT emploie 650 agents. Elle est divisée en 4 départements, et 2 départements transversaux, comme le montre le schéma ci-après :



Structure de la DSIT (Direction des Systèmes d'Information et des Télécommunications)

3) Contexte actuel

L'ouverture à la concurrence du fret européen est effective depuis mars 2003 et en fret intérieur depuis avril 2006.

Depuis le 1er janvier 2010, le marché des lignes ferroviaires internationales est ouvert à la concurrence, mettant fin au monopole de la SNCF.

L'évolution du marché, la recherche de compétitivité et l'évolution des technologies de l'information nécessite que la SNCF fasse évoluer ses Systèmes d'Information.

Mais le système existant, résulte de l'accumulation successive d'applications, souvent imbriquées, comportant des redondances et manquant de cohérence. La complexité de ce patrimoine existant génère des difficultés pour le faire évoluer avec les attentes des différents métiers de l'entreprise.

L'implémentation de l'urbanisme des SI (définition de l'urbanisme SI dans le chapitre II) à la SNCF s'est fait en 2003 par le biais d'une gouvernance SI, afin de garantir la bonne mise en place des différents axes nécessaires au développement de son référentiel de cartographies.

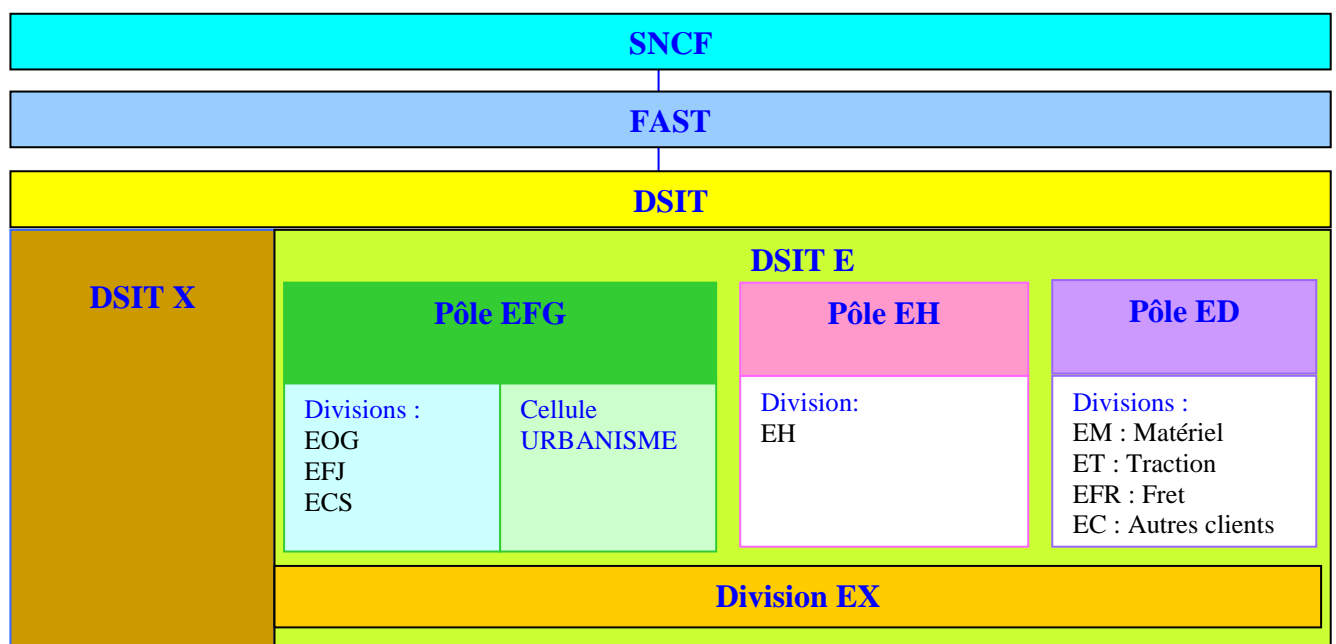
4) La cellule urbanisme de DSIT-EFG

Pôle EFG (Etudes, FAST et Gestion)

Ce pôle est une entité qui a pour mission de concevoir et maintenir les applications contribuant au SI Gestion de l'entreprise. Le pôle intervient ainsi auprès de ses clients internes et les accompagne tout au long du cycle de vie d'une application en tant qu'Assistance à Maîtrise d'Ouvrage (AMOA), en tant que Maîtrise d'œuvre (MOE) et pour les Projets et applications ERP.

La cellule urbanisme

La cellule Urbanisme DSIT-EFG/URB a en charge l'Urbanisme du Pôle DSIT-EFG pour le périmètre du SI Gestion. Elle réalise et maintien les modélisations métiers, fonctionnelles et applicatives des projets SI. Afin de présenter la vision du système d'information existant, de définir le SI cible et d'orienter les projets informatiques pour atteindre cette cible.



Détail de la DSIT-E (Etudes)

II. Présentation de la mission

1) Problématique

A chaque évolution fonctionnelle et/ou évolution de versions de l'ERP, des tests doivent être réalisées, à partir des exigences de tests déclinées dans le logiciel Quality Center.

Or, il est constaté du fait de la carence de ressources et par manque de temps, que les exigences ne sont pas toujours créées et mises à jour.

Les exigences sont pourtant importantes car elles permettent un suivi immédiat de la couverture des fonctionnalités durant la période de validation (taux de couverture) et l'affichage direct des anomalies associées à une exigence non couverte. Elles sont accessibles par tous en visualisation. Elles se présentent sous la forme d'une arborescence décrite dans REQUIREMENTS, contenant une liste exhaustive des Règles Métier.

Les modélisations des processus-procédures réalisés dans MEGA peuvent servir à créer les exigences de tests dans QUALITY CENTER. Or, MEGA utilise une base propriétaire pour stocker ses données, on ne peut donc pas récupérer facilement les données des cartographies. Aujourd'hui le service de la recette pour remplir les exigences doit donc visualiser les cartographies et saisir les données manuellement dans le logiciel de tests sous la forme d'arbre des exigences selon l'arborescence Processus/Procédures/Opérations des modélisations.

2) Objectif de la mission

Ma mission consiste à la réalisation et à la mise en place d'une passerelle entre l'outil MEGA et la solution QUALITY CENTER afin de reproduire automatiquement des exigences et scénarios de tests à partir des processus métier.

Récupération des données de MEGA (Macro-Processus, Processus, Procédure et Activités) pour créer automatiquement les exigences dans l'outil QUALITY CENTER (Arborescence, Structure et contenu).

Seule l'approche dans le sens MEGA vers QUALITY CENTER, sera étudiée.

3) Champs d'actions

L'équipe de recette d'intégration (DSIT-ECS/INT) a en charge les tests et recette d'intégration applicative des projets ERP.

Le Programme ERP vise à outiller les processus métiers majeurs (actes de gestion) des différentes fonctions, activités et domaines (Achats, Comptabilité, Logistique, Gestion de projet, Contrôle de gestion, ...) de l'Entreprise.

Dans la gestion de projet ERP, l'Intégration Applicative et les Recettes jouent un rôle primordial en aval des phases de conception détaillée, de réalisation et de Maintien en Condition Opérationnelle avant tout déploiement sur les différents sites de la SNCF.

Comme toute mise en place d'ERP, un certain nombre de spécifiques, sont nécessaires à réaliser et donc à intégrer dans les scénarios et fiches de test.

Pour comprendre ma mission et le contexte de mon étude, nous devons voir ce qu'est l'urbanisme des systèmes d'information, première partie de mon étude.

III. L'Urbanisme des SI

L'urbanisme a comme racine étymologique latine « urbs » qui signifie « ville ».

L'urbanisme est un ensemble de mesures techniques, administratives, économiques et sociales qui doivent permettre un développement harmonieux rationnel et humain des agglomérations.

L'urbanisation est une démarche de reconstruction utilisée pour l'intégration et la prise en compte des progiciels, des logiciels historiques et des composants hétérogènes. Le but est donc de réorganiser le système d'information pour le simplifier.

C'est une méthode qui fait référence au monde de l'architecture cadastrale. C'est une méthode qui propose de concevoir un système d'information de la même manière qu'un plan d'urbanisme. Elle repose sur l'agencement des fonctions informatiques les unes par rapport aux autres comme les zones et quartiers d'une ville. Le plan de construction d'une ville est élaboré en fonction des besoins des habitants. C'est la même chose pour le système d'information. Le plan du Système d'Information (SI) devra être réalisé en fonction des besoins des utilisateurs de ce SI et de la stratégie de l'entreprise.

Par extension au SI, l'urbanisme répond à la problématique d'insertion de sous-systèmes d'information dans un système d'information existant.

1) Qu'est ce qu'un système d'information ?

Un système d'information est un ensemble d'éléments qui recueillent de l'information, la traitent, la stockent et la diffusent afin d'aider à la prise de décision, à la coordination, et au contrôle d'une entreprise.

Le système d'information est utilisé par des acteurs (utilisateurs, administrateurs, managers) pour manipuler (consulter, modifier, communiquer) des données à l'aide de procédures.

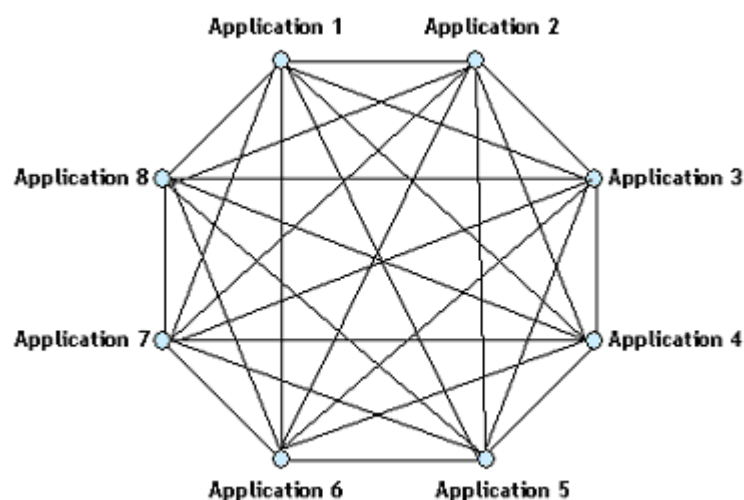
Au sein d'un système d'information informatisé on trouve des éléments applicatifs récurrents:

- L'intranet : Utilisé en interne, il est la principale source d'information pour les employés et permet la diffusion, sur un site accessible par le réseau, d'information en lien étroit avec la société.
- L'extranet : C'est le portail entre l'entreprise et ses partenaires.
- Plate-forme Web : Accessible par tous par un accès internet, il représente la possibilité pour l'entreprise de présenter son activité, de mettre en avant les atouts de l'entreprise et d'offrir des services. Elle permet de toucher un large spectre de clients et fournisseurs potentiels.
- ERP : Entreprise Ressource Planning, ou Progiciel de Gestion Intégré (PGI) en français. Il coordonne les différents flux des activités de l'entreprise. L'ERP intègre et regroupe toutes les données et les processus au sein d'un même système applicatif pour la même organisation. Décomposé en plusieurs modules (Finance, Industrie, Distribution, RH, Transport...) il permet de s'adapter au mieux à l'activité et au besoin de l'entreprise.
- La messagerie : C'est avec l'usage du téléphone un moyen rapide et efficace d'échanger des informations tout en gardant une trace. Il tend à devenir une référence, la loi considérant même la signature électronique et l'e-mail comme ayant la même valeur juridique qu'un document écrit.

- Développements spécifiques : Ce sont des développements sur mesure qui ne se trouvent pas sur le marché et répondent à des besoins précis et ponctuels de l'entreprise.

Dans la plupart des entreprises, chaque domaine d'activité utilise son propre Système d'Information. Cela leur permet d'être plus autonomes. Les acteurs sont propriétaires des données, ce qui leur permet d'être plus réactifs, de mieux connaître les données (produit, client). Le système d'information hétérogène répond à un besoin fonctionnel précis.

Les entreprises auront souvent négligé l'architecture même du système d'information. Dans ce cas l'évolution du SI croît avec l'ajout de nouvelles applications qui vont successivement développer des interfaces de communication afin de permettre le dialogue inter applicatif. Cette urbanisation dite naturelle est désorganisée. Une fois que la communication est établie, une simple évolution d'un côté ou de l'autre peut remettre en cause cette communication.



Système d'information

2) Définitions de l'urbanisme

L'urbanisme, au sens informatique du terme, est une discipline récente qui consiste à modéliser le Système d'Informations d'une entreprise pour présenter différentes visions nécessaires à la compréhension du fonctionnement de l'entreprise.

Le club des Urbanistes et Architectes des Systèmes d'Information est une association inter entreprise. Elle a été créée en 2000 par plusieurs entreprises. Elle a pour objectif d'échanger et de capitaliser les retours d'expérience entre professionnels de ces nouveaux métiers de l'urbanisation du SI.

Le Club Urba-EA propose la définition suivante:

« Urbaniser, c'est organiser la transformation progressive et continue du système d'information visant à le simplifier, à optimiser sa valeur ajoutée et à le rendre plus réactif et flexible vis à vis des évolutions stratégiques de l'entreprise, tout en s'appuyant sur les opportunités technologiques du marché.

L'urbanisme définit des règles ainsi qu'un cadre cohérent, stable et modulaire, auquel les différentes parties prenantes se réfèrent pour toute décision d'investissement dans le système d'information. »

D'après le CIGREF (Club Informatique des grandes entreprises françaises), l'urbanisme est une « démarche qui consiste à rendre un système d'information plus apte à servir la stratégie de l'entreprise et à anticiper les changements dans l'environnement de l'entreprise ». Elle « consiste à définir un système d'information cible qui puisse s'adapter et anticiper les différents changements ».

Dans cette optique l'urbanisme au travers de cartographies permet de consigner, d'identifier et de présenter les métiers réalisés au sein de l'entreprise pour atteindre ses objectifs. Les cartographies sont détaillées sur différents niveaux, de la vue d'ensemble au plan intérieur de chaque fonction.

Cette vision d'ensemble permet de garantir une cohérence globale pour le pilotage du SI.

3) La cartographie

La cartographie a pour objectif de décrire l'existant, les objectifs finaux et les différentes étapes de transition. Il existe quatre niveaux de cartographie:

- **Le niveau métier:** c'est le niveau stratégique. On y place les modèles d'objet métiers, les processus, les enjeux stratégiques.
- **Le niveau fonctionnel:** contient la définition des fonctions de l'entreprise, sous forme hiérarchique. Il est possible d'enrichir la cartographie fonctionnelle avec des flux d'échange.
- **Le niveau applicatif:** contient les éléments logiciels qui vont servir. Les éléments d'infrastructures seront aussi placés à ce niveau dans le but de supporter les flux d'échanges de données et de contrôle.
- **Le niveau technique:** contient l'environnement technique (machines, réseau, outillage..).

4) L'outil de modélisation

Pour réaliser leur mission, les urbanistes ont besoin d'un outil de modélisation reposant sur un référentiel commun. L'Atelier de Génie Logiciel (AGL) MEGA a été choisi par la SNCF pour la modélisation de son SI.

Sur le marché plusieurs AGL sont disponibles cependant ils ne sont pas exhaustifs par rapport aux besoins des utilisateurs, notamment en matière de lien vers la recette d'applications.

La cellule urbanisme de DSIT-EFG réalise les modélisations métiers, fonctionnelles et applicatives des projets du Programme ERP sur l'AGL MEGA. Les données des objets utilisés dans les cartographies sont stockées sur la base propriétaire de MEGA.

Un Atelier de Génie Logiciel (AGL), est un ensemble de programmes informatiques permettant eux-mêmes de produire des programmes. Un AGL désigne un environnement de conception d'applications. On peut d'ailleurs utiliser le terme équivalent d'environnement de développement.

Le logiciel MEGA

Dans la suite logicielle (qualifiée d'Atelier de Génie Logiciel) de l'éditeur MEGA International, des modules sont mis à disposition des architectes du SI Gestion :

MEGA Process : outil de représentation et de modélisation des architectures fonctionnelles (principalement diagrammes de processus, organigrammes, modèles opératoires et logigrammes).

MEGA Architecture : outil de représentation et de modélisation des architectures applicatives, diagrammes d'urbanisation et architectures techniques-physiques.

MEGA Documentation : solution qui permet de définir des documents-types, des descriptions (reposant sur des sélections) chargées de l'extraction des objets, de leurs liens et des caractéristiques (attributs) qui les composent. Ce module s'adresse à des utilisateurs avancés chargés de réaliser des documents-types ou d'effectuer des extractions de données non graphiques par exemple pour l'exploitation du référentiel (tableaux, représentations arborescentes, etc.).

MEGA Intranet : outil de génération de site web exploitant le référentiel pour accès intranet / extranet et internet aux données. Ce module s'adresse, lui aussi, à des utilisateurs avancés chargés de réaliser des descriptions HTML (des appels à services type JavaScript) pour formater l'information présentée et générée.



Architecture MEGA

D'autres applications, plus orientées modélisation ne sont pas déployées pour le programme ERP :

MEGA Database pour la modélisation des SGBD avec une approche Entité/Relation (rétro-génération possible),

MEGA AA&D et Development pour l'approche Méthode Merise-UML de conception logicielle,

MEGA Code Generation pour la production de code (objet),

MEGA Intégration pour les décrire les applications EAI.

La licence MEGA Exchange (APIs de MEGA) permet notamment des opérations d'import/export dans MEGA. Elle permet également l'intégration de MEGA avec d'autres applications Windows telles que le pack Microsoft Office. MEGA Exchange est accessible via des applications développées avec Microsoft Visual Basic, Windows Scripting Host, ou peut tout simplement être utilisé avec l'éditeur de scripts fourni par MEGA.

Une API (*Application Programmable Interface*), « *interface de programmation* » ou « *interface pour l'accès programmé aux applications* », est un ensemble de fonctions permettant d'accéder aux services d'une application, par l'intermédiaire d'un langage de programmation.

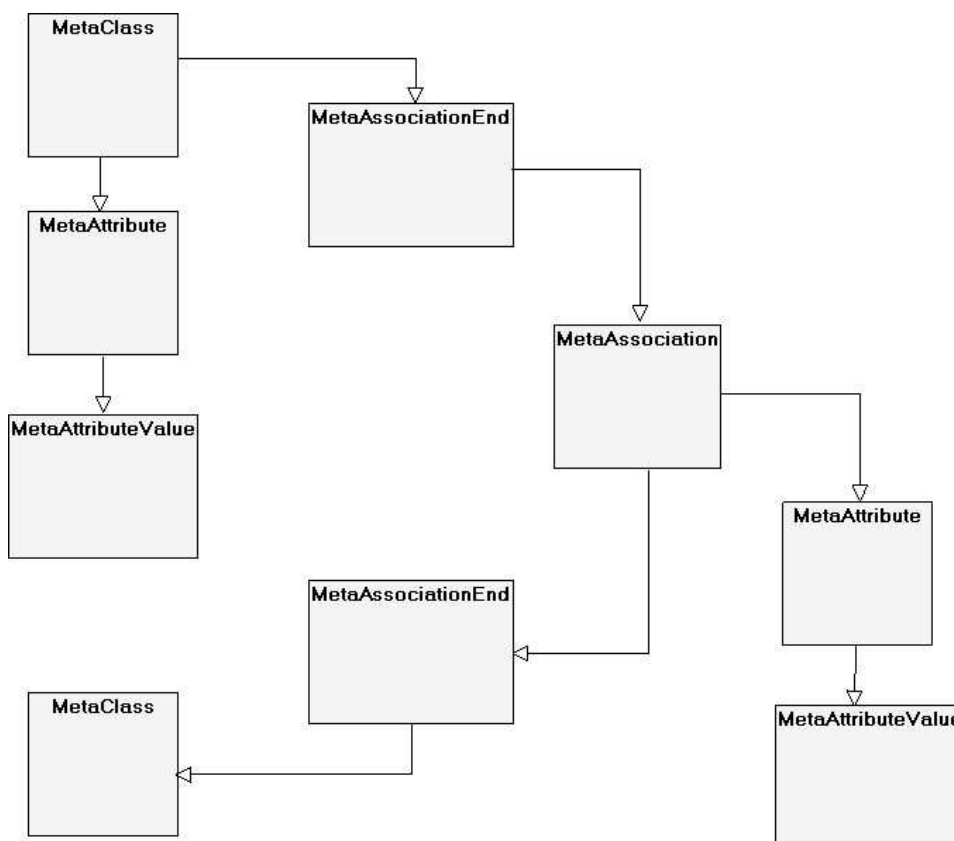
Les applications de MEGA reposent sur un socle commun, le "référentiel MEGA" constitué de sites (un seul pour le Programme ERP), d'environnements (pour le programme ERP nommé "Pole ERP GESTION") et de bases (Une base pour le programme ERP et une base pour chaque domaine) contenant les objets MEGA (diagrammes, objets, caractéristiques, clients, etc.).

A un environnement est associé un méta-modèle ou dictionnaire des objets contenant des extensions consistant en rajout de caractéristiques (attributs des objets et leurs valeurs tabulées), de liens, de formes (représentations de l'objet MEGA au sein des diagrammes).

Le méta-modèle MEGA

La réalisation de traitements pour gérer les données MEGA nécessite de bien connaître le méta-modèle de MEGA et les extensions de la SNCF.

Méta-modèle : fichier de paramétrage des applications MEGA, structure de mémorisation des données dans la base qui regroupe l'ensemble des types d'objets permettant de modéliser un système, ainsi que leurs caractéristiques, représentations et liens. La gestion du méta-modèle (réalisation des "extensions") est une tâche d'administration MEGA.



Extrait diagramme de classe méta modèle MEGA

Concept	Syntaxe
type d'objet	MetaClass
lien	MetaAssociation
patte	MetaAssociationEnd
type de lien	MetaAssociationType
caractéristique	MetaAttribute
groupement de caractéristiques	MetaAttributeGroup
valeur de caractéristique	MetaAttributeValue

Description des classes du méta modèle MEGA

Définitions :

Metamodèle : structuration des données MEGA manipulables

Metaclasse : objets manipulés pour la modélisation

Metaattribut : propriétés des objets et des liens entre objets

Metaattribut-valeur : valeur des propriétés des objets

Metaassociation : lien entre les objets (Metaclasse)

Metaassociationend : extrémité d'une Metaassociation (orientation du lien détermine le caractère mineur ou majeur de la patte)

Extensions du méta-modèle MEGA de la SNCF

Il s'agit d'ajout de nouvelles caractéristiques (attributs) des objets standard de MEGA, de nouvelles possibilités de liaisons entre objet (non graphiques), de formes (représentations graphiques) manuelles ou automatiques en fonction d'une ou deux valeurs de caractéristiques, de création de valeurs tabulées (valeurs discrètes) pour les attributs (caractéristiques) des objets.

Des requêtes sont possibles (requête simple par le langage de recherche ERQL proche du SQL et constitution/extraction par le biais des fonctions MEGA Matrice et Dialogue).

L'architecture MEGA est client-serveur mais asynchrone et transactionnelle (par système de journalisation). La base de données est propriétaire.

Fin 2010 la version 2005 MEGA ne sera plus supportée par le constructeur MEGA ce qui présage de la mise en place d'une migration vers MEGA 2009.

Nous venons de voir le premier contexte de ma mission, je vais vous présenter le deuxième contexte, la recette d'application.

IV. La Recette d'applications

Les spécifications d'un système informatique, sont les caractéristiques attendues du système. Elles portent sur :

- les Règles Métier permettant de décrire précisément les critères de décision et les actions à mener pour un processus donné. Ces règles, formalisées ou non, existent dans chaque cœur de métier. Elles capitalisent la connaissance, le savoir-faire d'une entreprise et traduisent sa stratégie
- les fonctionnalités du système
- son architecture,
- sa capacité maximale en termes de volumes : nombre d'unités d'œuvre fonctionnelles (contrats à gérer, clients, factures...), nombre d'utilisateurs, ...
- son implantation géographique
- ses conditions d'accès par les utilisateurs
- ses conditions d'exploitation et de maintenance,
- le niveau de qualité du système (performances, fiabilité, maintenabilité, exploitabilité...)
- le niveau de sécurité du système (durée maximale d'interruption du fonctionnement, protection contre les accès non autorisés...)

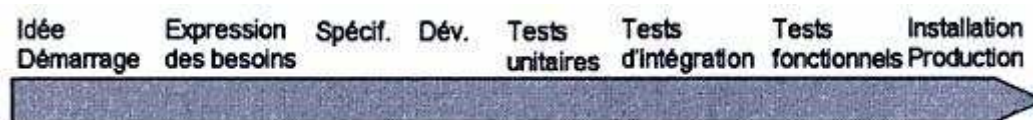
1) Définition

Le développement d'un projet comprend une phase finale de test. Cette étape est nécessaire pour s'assurer de sa conformité et du bon fonctionnement de l'application, qui consiste à mettre en situation des utilisateurs et repérer les dysfonctionnements pour les corriger. Dans cette optique, ceci se traduit par l'élaboration de scénarios. L'environnement de test comprend tous les éléments nécessaires pour l'exécution du projet en conditions réelles.

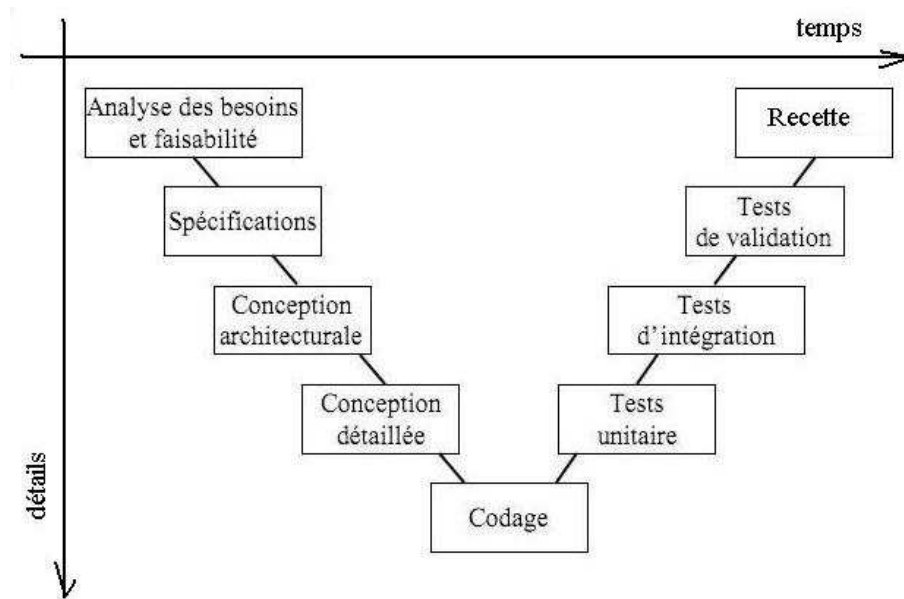
La recette d'application est un processus qui permet de vérifier les différentes exigences métiers décrites dans les spécifications fonctionnelles. Elle a pour but la validation des fonctionnalités avant la mise en production de l'outil. Les tests se composent de scénarios usuels du métier.

Les différents tests consistent à reproduire l'utilisation de l'application pour vérifier son adéquation fonctionnelle, organisationnelle (habilitations, profils) et technologique par rapport aux spécifications.

Il y a différents types de tests selon le cycle de vie du projet :



Cycle de vie d'un projet



Cycle de projet en V

Les différents types de tests dans la problématique de l'ERP :

- Test unitaires de développement

Il s'agit pour le programmeur de tester un module, indépendamment du reste du programme, ceci afin de s'assurer qu'il répond aux spécifications fonctionnelles et qu'il fonctionne correctement en toutes circonstances. Il s'accompagne d'une vérification de la couverture de code, qui consiste à s'assurer que le test conduit à exécuter l'ensemble des instructions présentes dans le code à tester.

- Test d'intégration

Les tests d'intégration ont pour but de valider le fait que toutes les parties développées indépendamment fonctionnent bien ensemble de façon cohérente.

- Test de validation

Les tests de validation permettent de vérifier si toutes les exigences du client décrites dans le document de spécification d'un logiciel, écrit à partir de la spécification des besoins, sont respectées.

- Test de performance

Tests dont l'objectif est de déterminer la performance du système informatique. Ils ont pour objectif de mesurer les temps de réponse d'un système applicatif en fonction de sa sollicitation.

- Test de non régression

Tests d'un programme, après une modification, pour s'assurer que des défauts n'ont pas été introduits ou découverts depuis les derniers tests. Ces tests sont effectués quand le logiciel ou son environnement est modifié. L'intérêt de ces tests est de limiter les anomalies relevées lors de la recette de l'application.

2) L'outil de recettes

Pour réaliser les tests, l'entreprise utilise l'outil Quality Center (QC) de l'éditeur MERCURY. Il permet de mettre en relations des exigences (spécifications), des plans de test, des scénarios, et d'y associer des fiches d'anomalies.

Le logiciel de recettes, Quality Center, permet d'avoir une démarche structurée en 4 phases :

- Gestion des besoins (Exigences métiers)
- Spécification de tests
- Définition des scénarios pour l'enchaînement des exécutions
- Suivi des anomalies

En amont, l'équipe de test part des spécifications qu'ils ajoutent dans la partie « REQUIREMENTS » et établit les points à valider dans le « TEST PLAN ». Puis les tests spécifiques (saisir un fournisseur, une facture) sont déroulés avec le « TEST LAB ». En cas d'anomalie, le test est mis en échec et l'utilisateur décrit son action et le blocage qui a suivi dans la partie « DEFECTS ».

Le contexte vu précédemment me permet maintenant d'expliquer l'étude effectuée pour la réalisation de ma mission.

V. Le lien entre l'Urbanisme SI et la Recette

1) Un référentiel commun

Fréquemment, dans les entreprises, il existe plusieurs bases de données relatives à un même ensemble d'informations. Lorsque il y a plusieurs ensembles de données relatifs au même objet métier, cela implique des problèmes de cohérence, d'exhaustivité et de passerelles.

L'urbanisme du Système d' Information permet de réunir toutes les informations du système d'informations de l'entreprise. Donc les données résultantes de l'urbanisation forment un référentiel pour l'entreprise.

Les Méta-données sont le regroupement de l'ensemble des informations concernant les données et leurs processus de mise à jour associés. Elles permettent de connaître l'information contenue dans les bases de données ; elles sont intégrées dans le référentiel.

Le référentiel est la source unique des informations à partager. Il apporte l'unicité des données, la cohérence des informations et la réduction des erreurs opérationnelles. Cela entraîne une plus grande qualité des informations saisies et mises à jour.

Il met en cohérence un sous-ensemble d'informations commun, dans l'ensemble du système d'information. Il contribue donc à la constitution d'une vision partagée des informations essentielles de l'entreprise. Il permet de partager les données de plusieurs applications. Le développement des systèmes d'information par la réutilisation de « composants » référentiels, est plus réactif.

Le référentiel réduit les erreurs opérationnelles, en évitant les redondances ou les incohérences de données clés, donc il gagne en efficacité. Il contribue à la synergie entre les données de l'Urbanisme du SI et de la Recette d'applications par une représentation commune des exigences métiers, donc il améliore la productivité.

2) Utilisation de l'Urbanisme SI pour la Recette d'applications

Comme les démarches d'urbanisme métier-fonctionnelle et d'intégration applicative sont actives depuis quelques années à la SNCF, des points d'adhérence ont été identifiés entre les données de l'urbanisme et celles de la recette. L'équipe de recettes consultent les modélisations identifiant les procédures métiers réalisées sous MEGA (Intranet ou base MEGA) pour créer l'arbre des exigences ou le détail des fiches et scénarios de test.

Afin de diminuer les temps de saisies pour la rédaction des fiches et plan de tests il faut étudier la mise en place d'une intégration de données MEGA vers le logiciel de recettes (QUALITY CENTER).

Tous les processus métiers de l'ERP doivent être présents et traduits dans MEGA par la cellule Urbanisme du SI Gestion. Or, tous les projets de l'ERP doivent être testés et « recettés » par l'équipe d'intégration de DSIT-ECS.

Donc établir le lien QC/ MEGA semble être judicieux.

Afin de réaliser ce lien, il est nécessaire d'inventorier les données présentes dans chacun des outils.

3) Les données de MEGA

On distingue des cartographies qui se rapportent à la description :

- des métiers utilisant le SI et donc répondant au « POURQUOI » du Système d'information.
- des procédures de travail mises en œuvre par le SI représentant le « QUOI » du Système d'information.
- des applications, ses sous-systèmes et ses constituants logiques (services, IHM, bases, flux applicatifs et communications), les référentiels, les flux et les échanges d'information qui répondent au « COMMENT » du Système d'information.
- des sites et acteurs (entités et rôles) utilisateurs du SI identifiant le « QUI » du Système d'information.
- de l'environnement technique permettant de réaliser les fonctionnalités gérées répondant aux questions « QUAND » et « OU » du Système d'information.

Parce que, les données fonctionnelles permettent d'identifier les différentes fonctions de l'entreprise (les différents métiers) et les données métiers se rapportant à l'exécution et à la réalisation d'un métier, ce sont ces données qui seront utiles à la création d'exigences, plan de tests et scénarios pour la recette d'application. Les données MEGA sont regroupées en projets.

a) Les données utilisées

La modélisation fonctionnelle - métier dans le référentiel Méga:

- Les processus du périmètre fonctionnel du projet représentent le métier concerné,
- Les procédures 'métier' implémentées dans le SI ERP avec un détail assez fin sur:
 - ✓ Les opérations réalisées (transactions, tâches manuelles, traitements batch planifiés ou pas).
 - ✓ Les règles de gestion (contraintes) de ces opérations.
 - ✓ Les référentiels et objets métiers supports de l'information.
- Compléments: références externes.

b) Description des données

Processus : Un processus permet de décrire ce que fait une entreprise, son métier. Un processus est une chaîne de valeur fournissant un bien ou un service à un client interne ou externe à l'entreprise. Il est décrit essentiellement par les échanges de flux entre les différentes activités de l'entreprise. Il est mis en œuvre par des procédures.

Procédure : Une procédure décrit comment est organisée l'entreprise pour réaliser les opérations nécessaires à son fonctionnement. Elle détaille la marche à suivre pour mettre en œuvre tout ou une partie du processus d'élaboration d'un produit ou d'un flux. Une procédure est représentée par la séquence des opérations réalisées par les acteurs de l'entreprise.

Opération : Une opération est une étape d'une procédure correspondant à l'intervention d'un acteur de l'organisation dans le cadre d'une des activités de l'entreprise. Ce peut être une opération industrielle (usiner une pièce) Ou logistique (réceptionner une livraison), ou un traitement d'information (enregistrer une commande). Une opération peut être décomposée en Tâches élémentaires.

Contrainte : Une contrainte représente un contrôle ou une règle de gestion qui doit être appliquée lors de l'exécution d'un traitement.

Référence Externe : Une référence externe permet d'associer un objet à un document qui provient d'une source extérieure à MEGA. L'emplacement de ce document peut être indiqué comme le chemin d'un fichier ou l'adresse d'une page WEB, par l'intermédiaire de son URL.

Vue d'ensemble des Processus : La vue d'ensemble des processus représente les principaux processus de l'entreprise et les messages qui permettent de les enchaîner entre eux.

Diagramme fonctionnel d'un processus : Le diagramme fonctionnel d'un processus permet de montrer les flux échangés entre les activités qui représentent l'intervention de chacun des métiers de l'entreprise au processus.

Logigramme : Un logigramme est une représentation graphique utilisée pour illustrer le déroulement d'une procédure. Il décrit les traitements effectués à l'intérieur d'une entreprise lors de la mise en œuvre d'une procédure. Il permet de préciser les opérations effectuées pendant son déroulement et les messages échangés.

4) Les données de Quality Center

Afin de constituer un référentiel de test pour les versions de l'ERP, une arborescence des exigences est construite par projets. Elle permet de constituer une référence, commune pour chacun des intervenants qui sont l'équipe de Recette d'Intégration applicative ainsi que les équipes de recette projet et celle de la Direction de Programme ERP.

Des scénarios et des fiches de tests intègrent les modes opératoires avec les règles de gestion sont ensuite enrichis pour être joués ou simulés manuellement sur l'ERP pour validation. Tous les incidents sont consignés par l'outil.

Chaque module QC est découpé par type de test puis par Projet.

a) Les données

- Arborescence(s) et description des exigences,
- Fiches de test,
- Scénarios de test.

Après étude, seules les exigences seront réalisées par l'outil d'automatisation contrairement aux fiches et scénarios de tests qui sont soumis à la traduction de la perception humaine.

Exigence de tests

- Une exigence doit répondre aux questions suivantes :
 - Est-elle unique ? Est-ce la seule exigence qui définit cet objectif particulier ?
 - Est-elle précise ? Contient-elle des termes imprécis qui sont difficiles à interpréter ?
 - Est-elle limitée ? Les objectifs sont-ils bien limités ?
 - Est-elle testable ? Peut-on créer au moins une fiche de test vérifiant absolument tous les aspects de cette exigence ?

b) Description des données

- **Exigence:**

Les exigences sont les contraintes relatives aux règles de gestion du métier. Elles doivent correspondre à une cible documentée que l'application doit atteindre. Elle est vérifiable et se rapporte à un métier, un indice de performance ou une caractéristique technique.

- **Fiche de tests:**

La fiche de test est constituée d'une suite d'étapes successives pour lesquelles on attend une réponse du système visant à prouver que le respect des exigences est correct.

Une fiche peut prendre en compte des tests passants ou non passants.

- **Scénario de tests:**

Il représente un processus applicatif (fonctionnel, technique ou transverse).

Il se compose de une à plusieurs fiches de tests, ordonnancées et séquencées de façon procédurale.

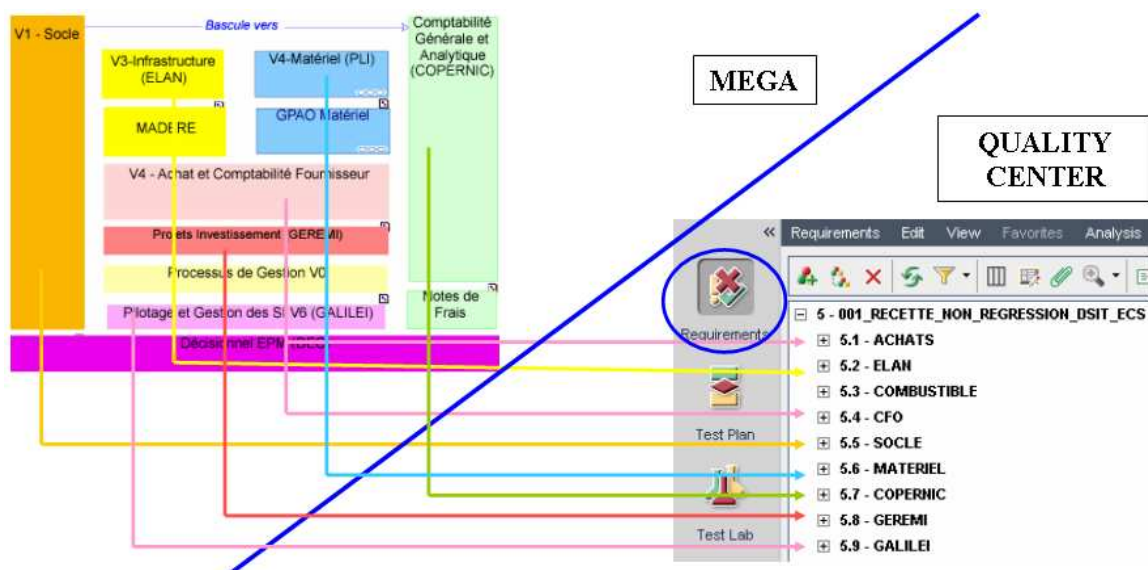
Nous avons posés ici les données des deux outils, nous allons voir maintenant les correspondances entre les deux.

5) Correspondances MEGA et QC

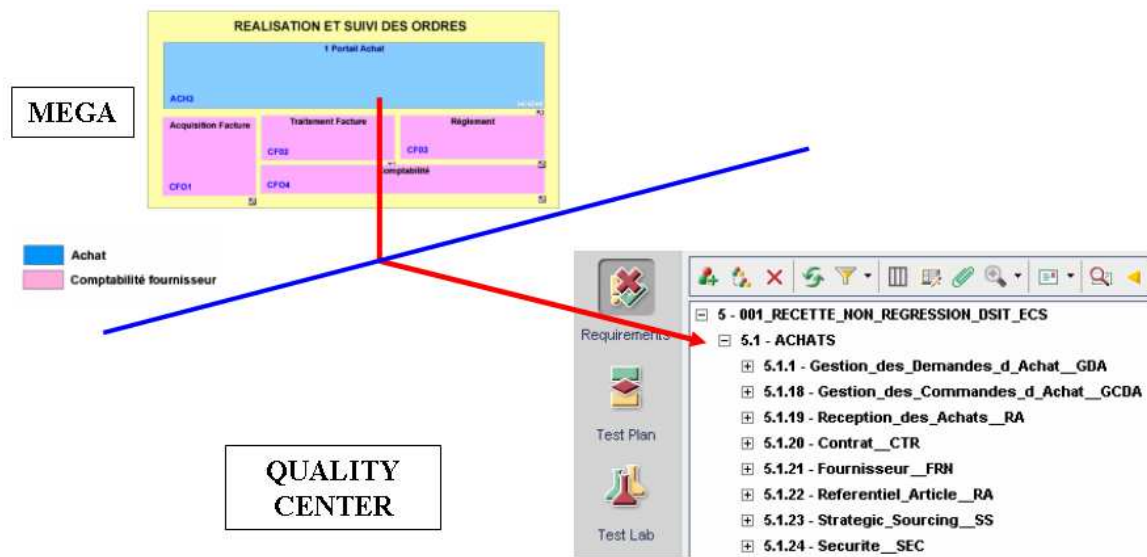
Les objets métiers présents dans MEGA deviennent des exigences de tests dans Quality Center.

Ci-dessous les différents niveaux de MEGA correspondants aux différents niveaux de QC :

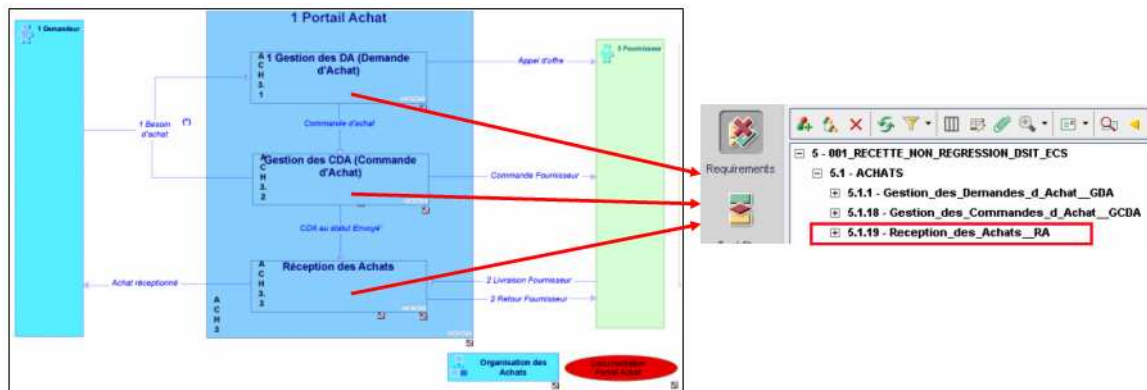
- Le premier niveau de l'arborescence de QC représente les différents types de recettes (Non régression, performance, développement, ...) décrits dans la partie sur la recette d'application.
- Les processus dans MEGA visibles sur le diagramme de vue d'ensemble des processus correspondent au deuxième niveau de l'arborescence d'exigences de Quality Center.



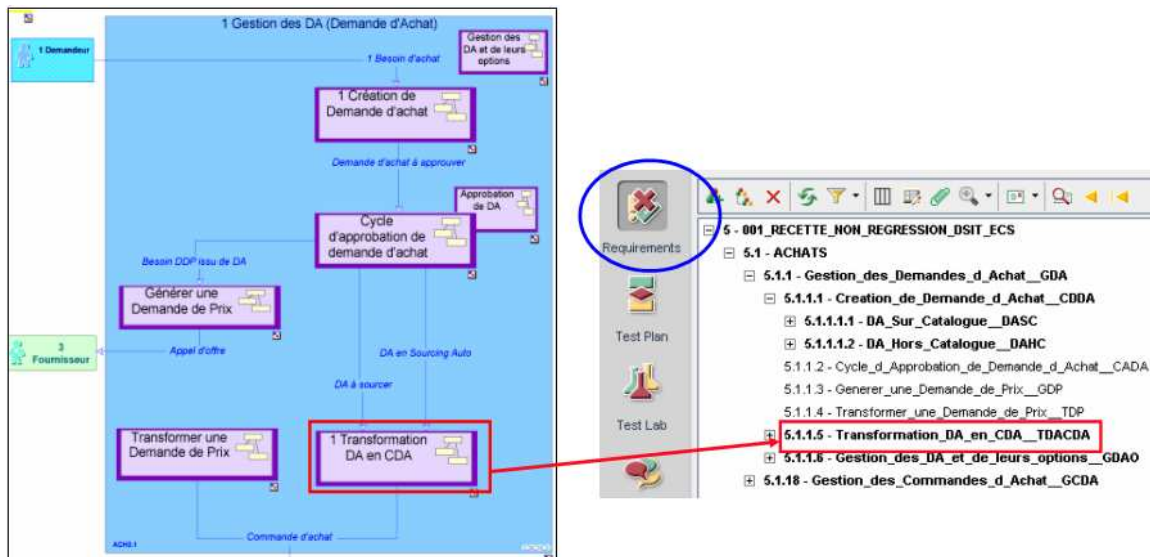
- Le deuxième niveau de granularité du diagramme de vue d'ensemble de processus reprend les processus du premier niveau correspondants aux différents projets.



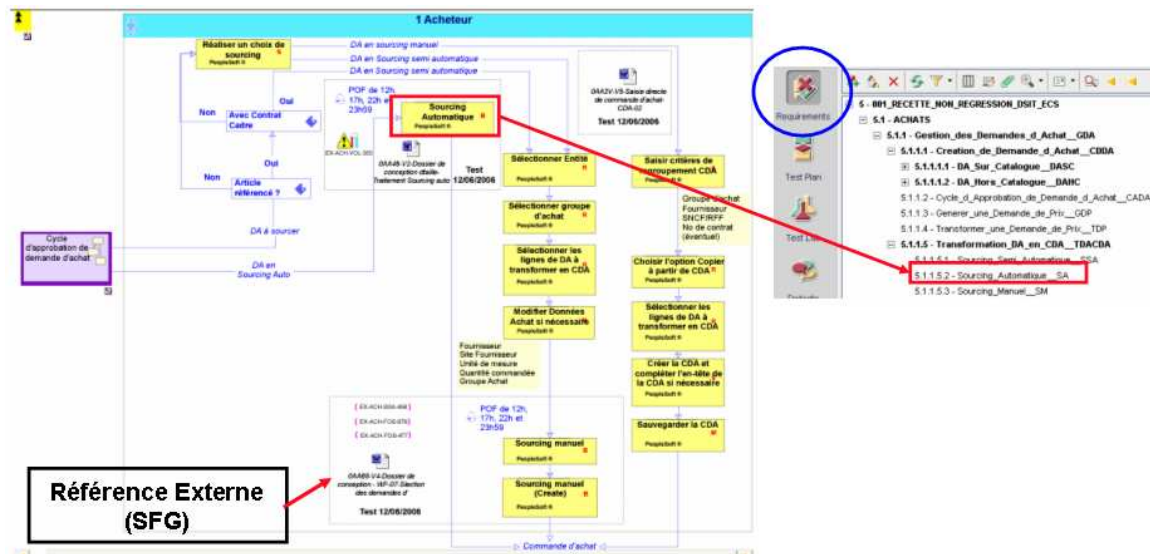
- Les (sous-)processus MEGA dans le diagramme fonctionnel d'un processus correspondent au troisième niveau de l'arborescence d'exigences de Quality Center.



- Les procédures MEGA dans le deuxième niveau de granularité du diagramme fonctionnel d'un processus correspondent au quatrième niveau de l'arbre des exigences de Quality Center.



- Les opérations MEGA présentes dans le logigramme correspondent au dernier niveau de l'arborescence d'exigences de Quality Center, donc à l'existence même.



Le contenu des Référence Externes permet de définir précisément le périmètre des tests en complétant la description des exigences de test.

On peut distinguer jusqu'à 5 niveaux de granularité.

Précédemment, nous avons vu l'étude préliminaire, voici maintenant le cœur de ma mission, la résolution de celle-ci.

VI. La construction d'un outil

1) Les étapes

- ✓ Analyse des tâches d'urbanisme et d'intégration applicative
- ✓ Etude des échanges possibles entre MEGA et Quality Center
- ✓ Evaluation des différentes solutions
- ✓ Prototypage d'une passerelle MEGA → Quality
- ✓ Validation sur un cas test réel de l'apport de la passerelle
- ✓ Ajout méthodologique et rédaction documentaire fixant les éléments de MEGA utilisables par Quality Center
- ✓ Mise en place de la passerelle MEGA → QC en concertation avec les équipes Projets ERP, la Direction du Programme ERP, l'équipe d'intégration DSIT-ECS/INT et DSIT-EFG/URB

2) Etudes des différentes solutions

Les différentes solutions pour initialiser les exigences de la recette automatiquement :

- Gestion des I/O entre les 2 outils avec Excel
- Interface (IHM) en lien avec MEGA et QC
- Ajout d'une nouvelle fonctionnalité dans QC ou dans MEGA

Analyse des solutions

❖ Import/Export par Excel

Les produits Microsoft Office permettent d'un point de vue utilisateur de se servir du logiciel pour accomplir un certain nombre de tâches afin d'arriver à un certain résultat mais offre aussi pour le développeur, une véritable plateforme de développement. En effet, les macros sous Excel possèdent une partie sous-jacente qui permet de contrôler entièrement l'application grâce au langage VBA (Visual Basic for Applications).

Ces Macro-commandes permettent de mettre, sous la forme de lignes de commande, des tâches longues et/ou répétitives. Mais aussi à l'aide de certaines bibliothèques de composants (ici l'ajout des bibliothèques MEGA et QC API), de créer de véritables applications.

Le script VBA Excel se connecte à MEGA pour récupérer les données métiers.

Il transforme ensuite les données extraites pour correspondre aux exigences de QC. Le Script se connecte ensuite à QC pour créer des exigences à partir des données du fichier Excel créé.

Au final, la solution Import/Export Excel s'avère rapide et simple à développer. Cependant elle s'adresse à des utilisateurs avertis car elle est peu interactive. De plus elle présente un script lourd à exécuter et la génération du document Excel n'apporte aucun intérêt, que le stockage des données temporairement entre l'export et l'import.

❖ IHM externe

Une interface homme-machine (IHM) permet de concevoir des systèmes ergonomiques, efficaces et faciles à utiliser, afin que l'utilisateur puisse interagir avec le système ou l'application.

L'application serait réalisée en Visual Basic à l'aide de l'outil Visual Studio. L'application se connecte à MEGA et à QC, elle récupère les données métiers et crée les exigences dans QC avec les données récupérées de MEGA.

Finalement la solution d'un IHM en Visual Basic semble plus longue à développer que la solution précédente mais apporte une interface ergonomique et conviviale pour l'utilisateur, donc une facilité d'utilisation.

❖ **Fonctionnalité intégrée à QC**

Dans le logiciel, un bouton ou un menu peut être ajouté à l'interface actuelle avec un code VB script pour réaliser les traitements.

Un script VB script écrit sous l'éditeur de Scripts de QC qui se connecte à MEGA pour récupérer les données métiers et créer les exigences avec les données récupérées.

La solution d'ajouter une fonction à l'outil de recette permet une connexion vers l'outil MEGA transparente pour l'utilisateur bien qu'il soit plus difficile de se connecter à MEGA depuis la plateforme d'édition de scripts sur QC. L'utilisateur sera familier à l'environnement de la fonction puisque c'est son outil de travail quotidien. L'avantage est qu'il n'y a pas de connexion à QC à prévoir puisque l'utilisateur est déjà connecté avec son login. L'inconvénient pour l'utilisateur est qu'il doit se connecter obligatoirement à QC pour faire la migration, cela lui prend donc plus de temps.

❖ **Fonctionnalité intégrée à MEGA**

Dans le logiciel, un bouton ou un menu peut être ajouté à l'interface actuelle avec un code VB script pour réaliser les traitements.

Un script VB script écrit sous l'éditeur de Scripts de MEGA qui se connecte à QC pour créer les exigences avec les données métiers.

La solution d'ajouter une fonction à l'outil de modélisation permet d'éviter d'ouvrir une session MEGA par le script, par contre l'utilisateur doit donc avoir un compte MEGA et la multiplication des comptes n'est pas vraiment recommandée puisque cela implique la multiplication des licences. Les utilisateurs risquent d'être réticent à devoir se connecter à un outil auquel ils ne sont pas habitués pour lancer l'export. Même problème que la solution précédente, il est plus difficile de se connecter à QC depuis MEGA, de plus il est nécessaire d'utiliser la licence MEGA Exchange pour développer le script dans MEGA.

3) Comparaison des solutions

Caractéristiques remplies par les Solutions	Import/Export Excel	IHM externe	Fonctionnalité QC	Fonctionnalité MEGA
Ergonomie	- -	+ +	+	+
Rapidité d'exécution	- -	-	+	+
Simplicité d'utilisation	- -	+ +	+ +	- -
Maintenance aisée	+ +	+	- -	-
Intégration d'évolution	+ +	+	- -	-

La réalisation d'une IHM externe en Visual Basic est la solution la plus évidente en fonction des caractéristiques analysées dans le tableau comparatif. La plupart des caractéristiques sont remplies mais la rapidité d'exécution reste son point faible puisque la connexion aux 2 outils prend plus de temps qu'une seule connexion. Cependant la rapidité d'exécution n'est pas le critère le plus important car la fréquence des imports dépend des versions donc tout les 2-3 mois minimum seulement.

4) La solution retenue

Nous l'avons vue précédemment, 4 solutions ont été étudiées, il ressort de cette étude que la solution proposant une IHM en dehors des outils de modélisation et de recette est la solution la plus adaptée aux besoins de tous.

a) Généralités

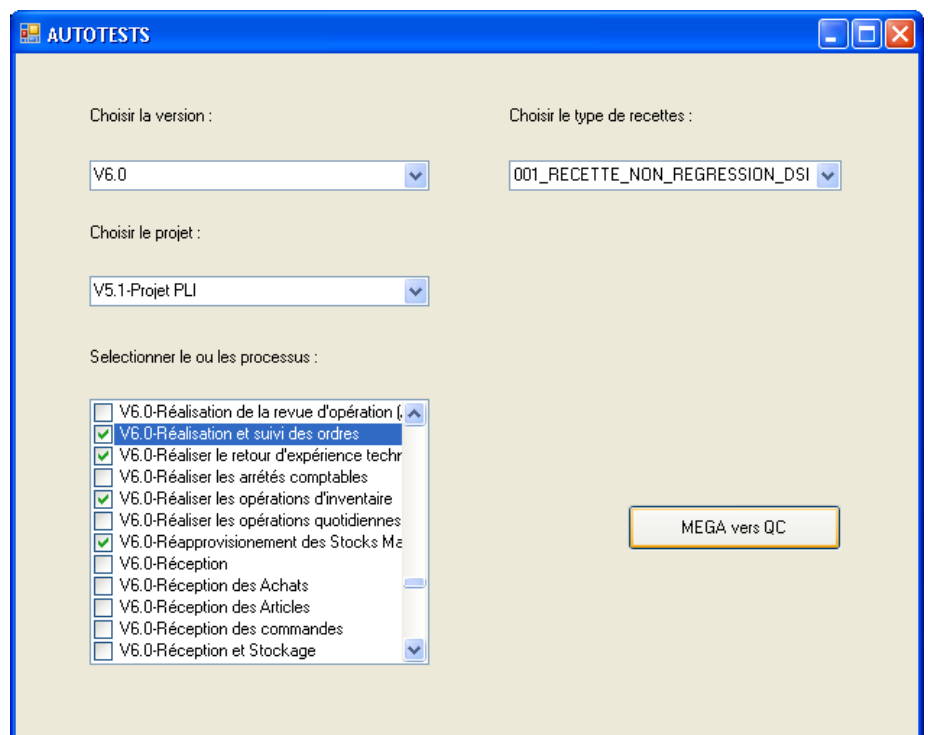
- L'outil permettra donc d'exporter les données des cartographies MEGA pour initialiser les exigences QUALITY CENTER.
- L'objectif principal du logiciel est l'ajout des exigences en fonction des données de MEGA. L'utilisateur peut choisir le diagramme qu'il souhaite pour l'exporter et l'importer automatiquement dans QC. Cela résoudra donc le problème des exigences non remplies dans le logiciel de recette. Il est laissé à la charge de l'utilisateur de les modifier et compléter par la suite puis les relier au plan de tests.
- Le logiciel doit permettre de récupérer un ou plusieurs processus d'un projet ainsi que les objets rattachés à ce processus. Il doit ensuite créer les exigences en fonction d'un type de recette choisi par l'utilisateur dans le logiciel de recettes. Si le processus est importé plusieurs fois dans QC il n'écrase pas la dernière version. Il faut vérifier si l'exigence existe déjà pour ne pas la remplacer sinon le lien avec le plan de tests va disparaître. Les objets qui ont été supprimés dans MEGA depuis le dernier export et les objets dont le nom a été modifié dans MEGA ou dans QC resteront présents. L'utilisateur doit donc vérifier lui-même qu'il n'y a pas de doublons. Les exigences en trop ne sont pas un problème puisqu'elles ne seront reliées à aucun plan de tests donc les statistiques ne seront pas faussées.

○ L'Analyse des tâches d'Urbanisme et de Recette apportent le résultat suivant :
L'arbre Processus/ Procédures/Operations de MEGA peut être assimilé au squelette de l'arbre des exigences de Quality Center.

- Projet MEGA (Diagramme de vue d'ensemble des processus) → Arborescence Exigences QC
- Processus MEGA (Diagramme de vue d'ensemble des processus) → Arborescence des Exigences de Tests QC
- (Sous-) Processus MEGA (Diagramme fonctionnel d'un processus) → Arbre d'Exigences QC
- Procédure MEGA (Diagramme fonctionnel d'un processus) → Arbre d'Exigences QC
- Opération MEGA (Logigramme) → Exigence QC

b) Conception

○ Maquette de l'interface :



Liste des objets de l'interface :

ComboBox : « cbProjet »	Liste des projets (la version est préfixée au projet)
ComboBox : « cbTypeR »	Liste des types de recette (1 ^{er} niveau de l'arborescence de QC)
CheckBoxList : « CkLBProcess »	Liste des processus du projet. Elle ne s'affiche que lorsque l'on a choisi un projet.
Button : « Import »	Exécute le traitement en fonction des paramètres choisis.

○ Mode d'emploi

- Sélection du projet et de la version :

Tous les projets sont ajoutés à cette ComboBox pour pouvoir choisir le projet et la version de ce projet (préfixé au projet)

- Sélection des processus à importer :

Tous les processus sont ajoutés à cette CheckListBox pour pouvoir choisir un ou plusieurs processus (1^{er} niveau dans MEGA) à importer dans QC

- Sélection du type de recette :

Le premier niveau de l'arborescence de QC est ajouté à cette ComboBox.

- Importer les données de MEGA vers QC :

Le traitement pour initialiser les exigences dans le répertoire choisi (si l'exigence existe il ne la réinitialise pas) se réalise lorsque l'on clique sur le bouton « MEGA to QC ».

o Comportement du programme

- Connexion MEGA :
 - Définition de l'environnement, du login, du mot de passe et de la base
 - Ouverture de la base
- Récupérer les versions (Nom)
- Récupérer les projets en fonction de la version sélectionné (Identifiant et Nom)
- Récupérer les processus parents en fonction du projet sélectionné (Identifiant et Nom)
- Connexion QC :
 - Connexion à l'outil
 - Définition du login, du mot de passe, de l'environnement et de la base.
- Récupérer les types de recette (niveau 1 de l'arborescence QC) (Nom)
- Récupérer les processus sélectionnés et les procédures et opérations de ces processus (Identifiant, Nom, Chemin Parent).
- Déconnexion de MEGA
- Ajout des différents niveaux de l'arborescence des exigences dans QC :
 - Pour chaque objet MEGA
 - Dérouler l'arborescence en fonction du chemin parent de l'objet
 - Vérifier si l'exigence est déjà présente ou non en fonction de l'identifiant de l'objet MEGA
 - Si elle n'existe pas, créer l'exigence (initialisation du nom et ajout de l'identifiant MEGA)
- Déconnexion de QC

o Cas de test sur données réelles

Des tests unitaires seront réalisés au cours du développement :

- Tests de récupération des données de production dans MEGA
- Tests des fonctionnalités de l'IHM
- Tests de l'envoi de données tests vers QC dans une base de tests

Une fois l'outil réalisé, des tests seront effectués en utilisant les données de production de MEGA mais sur une base de test de Quality Center.

Après validation de ces tests, juste avant la mise en place de l'outil, des tests seront réalisés dans des conditions réelles d'utilisation donc dans la base de production de Quality Center mais dans une arborescence à part pour ne pas modifier l'arborescence actuelle.

Une présentation sera faite aux utilisateurs pour valider l'outil et le niveau de granularité des exigences, suites aux tests réalisés à la demande des utilisateurs durant la présentation.

- o Documentation et Formation des utilisateurs

Un manuel utilisateur (mode opératoire) sur l'outil sera réalisé mais il y aura aussi une aide contextuelle en ligne sous forme de bulles d'informations directement sur l'IHM.

Un dossier technique sera réalisé pour une maintenance future.

- c) Choix techniques

- Connexion aux outils

La connexion à MEGA et QUALITY CENTER se fait un par login dédié à l'application. Il aurait été possible de demander à l'utilisateur de se connecter avec son propre login mais tous les utilisateurs n'ont pas de login MEGA et il faut que ce soit transparent pour eux la connexion à l'un ou l'autre des outils.

- Sélection des processus

Pour une question de rapidité d'exécution il est demandé à l'utilisateur de choisir seulement les processus qu'il veut importer dans QUALITY CENTER. Il a la possibilité de cocher tout les processus mais l'import peut prendre un certain temps donc s'il n'a besoin que d'une partie du projet seulement l'import est beaucoup plus rapide.

- Sélection du type de recette

L'utilisateur doit choisir un type de recette (le premier niveau de l'arborescence de QUALITY CENTER) pour pouvoir ajouter les exigences importées. La sélection est rendue obligatoire car les différents projets seront importées dans chacune des différentes recettes mais a des moments différents et par d'autres équipes donc il est moins lourd d'importées qu'un seul type de recette en une fois.

- Interface graphique

Le choix de faire une interface graphique a été fait car cela est plus simple à utiliser pour un utilisateur lambda.

- Ajout de l'identifiant MEGA dans QC

Le programme fait le lien entre les données d'un import ultérieur et celle du nouvel import, grâce à l'identifiant MEGA stocké dans les exigences de la base QC. Lors d'un second import, il supprimera automatiquement les exigences qui ne sont plus existantes dans MEGA, complétera les exigences existantes et ajoutera seulement les ajouts. Donc il n'écrase pas les exigences déjà importées puisqu'elles peuvent être relié à des plans de tests.

d) Limites

Les utilisateurs qui avaient pour habitude de ne pas vraiment remplir la partie exigence de la recette ne comprennent pas ce changement de méthodes, ils sont donc réticents à utiliser l'outil bien que la demande vienne de leur direction.

Un autre problème est la migration vers la version de MEGA 2009, cela pourrait engendrer des évolutions du script, de part les modifications du Meta-Modèle et un passage à une base Oracle contre une base propriétaire actuellement.

e) Axes d'amélioration

Le logiciel répond aux spécifications attendues, cependant il y a certains points à améliorer dans l'avenir.

Par exemple créer une gestion dynamique entre MEGA et QC, permettant d'avoir les dernières mises à jour MEGA.

De plus il est aussi intéressant de faire un lien automatiquement entre les exigences, le test plan et les scénarios.

L'outil pourrait proposer aussi une visualisation des cartographies sélectionnées et même une animation représentant le processus a effectué par le testeur.

Conclusion

Au travers de cette étude nous avons constaté qu'il existe un lien entre l'Urbanisme des SI et la Recette d'application mais pas de passerelle. L'objectif était donc de définir les données pour réaliser cette passerelle.

A ce jour, l'application n'est pas encore finalisée, il est donc encore trop tôt pour établir le bilan du projet et en tirer les enseignements. Le projet est cependant en bonne voie avec un maintien de l'objectif pour septembre 2010.

La réussite du projet est centrée sur 3 facteurs clé de succès : une bonne maîtrise de MEGA et de QUALITY CENTER facilitée par des formations approfondies, l'appropriation du contexte au travers d'interviews réalisées auprès des équipes de travail et le support des équipes d'urbanisme en place.

L'urbanisme est au centre de cette démarche de réorganisation du SI mais nécessite une bonne adhésion au changement et des ressources financières.

Sur ce dernier point la crise actuelle n'arrange rien. En effet, plusieurs entreprises jugent cette démarche comme secondaire et réduisent les investissements sur le sujet. A terme, cela pourra entraîner l'arrêt du maintien du référentiel d'urbanisme. A la SNCF une telle situation pourrait engendrer, une non utilisation de l'application.

a) Intérêts professionnels

- Approfondissement de mes connaissances sur le cycle de vie d'un projet informatique. Particulièrement les phases de recette, d'intégration relatives à l'implémentation de la solution ou à l'intégration corrective ou évolutive du produit (la gestion des incidents et problèmes).
- Découverte de l'urbanisme des SI, et de l'intérêt de cette discipline pour optimiser l'organisation d'un SI.
- Découverte de la recette d'application et de l'ampleur et l'importance de ce domaine en entreprise.
- Les formations sur MEGA et QUALITY CENTER m'ont permis d'appréhender de nombreux aspects relatifs à la modélisation de processus/procédures, à la qualité, au « versionning », à la restitution de données sur site web. J'ai également bénéficié de formations externes à MEGA API (Exchange) et QC API.
- Initiation à la programmation VB Script pour le paramétrage avancé des outils QUALITY CENTER et MEGA.
- Observation du déploiement d'un ERP, et comment on implémente progressivement les différentes fonctions de l'entreprise (achat, comptabilité, ...)

b) Intérêts personnels

- Gestion d'un projet en totale autonomie
- Intégration dans une cellule existante. Les nombreux contacts que j'ai eu à nouer au sein de la DSIT et parfois avec des intervenants extérieurs (ex : MEGA) m'ont permis d'acquérir une certaine aisance relationnelle qui reste évidemment à consolider et à renforcer.

Abréviations

AGL : Atelier Génie Logiciel

API : Application Programmable Interface

DSIT : Direction des Systèmes d'Information et des Télécommunications

ECS : Etudes Centre de Solution et d'Intégration

EFG : Etudes Fast et Gestion

EFJ : Etudes Finance et Juridique

EOG : Etudes Outil de Gestion

ERP : Entreprise Ressource Planning ou PGI (Progiciel de Gestion Intégré) en Français.

FAST : Finance Achat Systèmes d'Information et Télécommunications

IHM: Interface Homme Machine

SI : Système d'Information

SNCF : Société Nationale des Chemins de Fer

QC : Quality Center

Références bibliographiques

Livres :

Le projet d'urbanisation du S.I. : démarche pratique avec cas concret.

Christophe LONGEPE. Editions DUNOD, 2004.

Urbanisation et modernisation du S.I.

Bernard LE ROUX, Luc DESBERTRAND, Xavier TANG, Julien TIXIER, Pierre VERGER.
Editions Lavoisier 2004.

Urbanisme informatique et architecture applicatives.

Guy LAPASSAT. Editions Lavoisier 2003.

Pratique de l'Urbanisme des Systèmes d'Information des entreprises.

Urba SI Club. Edition 2003.

Conduite de projets informatiques: Développement, analyse et pilotage.

Brice-Arnaud Guérin. Editions ENI 2009.

Développement de projet en informatique.

Carl-August Zehnder. Edition Presses polytechniques romandes 1990.

Documentation en ligne :

« [Urbanisme des villes et urbanisme des systèmes d'information.](#) »

« [Approche méthodologique pour l'urbanisme des systèmes d'information.](#) »

Véronique Levasseur

« [L'apport historique de l'urbanisme des villes pour l'urbanisme des systèmes d'information.](#) »

« [L'apport de l'ingénierie des besoins à l'ingénierie de l'urbanisme des systèmes d'information.](#) »

Isabelle Contini

« [L'urbanisation du système d'information de Renault.](#) »

Jean-Christophe Bonne

« [Pourquoi urbaniser les systèmes d'information.](#) »

Marc Desreumaux

« [Le système d'information vivant et démocratique.](#) »

Stéphane Grès

« [L'urbanisation des systèmes d'information - le moyen de parvenir.](#) »

Jean Joskowicz

« [Démarche de conception d'une cible urbanisée et du plan de convergence.](#) »

« [L'urbanisme : une opportunité pour réinventer la relation maîtrise d'ouvrage/maîtrise d'oeuvre.](#) »

« [Les limites actuelles de l'urbanisme des systèmes d'information.](#) »

Christophe Longépé

« [Urbanisation du système d'information de l'ANPE.](#) »

« [Evolution du rôle du système d'information - du concept au processus.](#) »

Michel Volle

« [Dossier pratique urbanisation.](#) »

Antoine CROCHET-DAMAIS, JDN Solutions

Sites Internet :

Club Urbanisme des SI – Enterprise Architecture

<http://www.urba-ea.org/>

Club Informatique des Grandes Entreprises Françaises

<http://www.cigref.fr>

Sites d'informations et forums :

www.01net.com

www.benchmark.fr

www.wikipedia.org/

www.developpez.net

www.clubic.com/

www.commentcamarche.net/