

Architecture d'Entreprise & Agilité

Livre Blanc

Octobre 2008

Le CEISAR produit des Livres Blancs sur l'Architecture d'Entreprise tous les 6 mois.
Les thèmes sont choisis par les Sponsors du CEISAR : Air France, Axa, BNP Paribas, Michelin et Total.
Pour la période d'Avril à Octobre 2008, le thème traité est **l'Agilité de la Transformation dans l'Entreprise et la Formation des Maitrises d'Ouvrage**.

Ce document résume les recommandations essentielles définies par le Groupe de Travail du CEISAR auquel ont participé des représentants des Sponsors ; nous souhaitons particulièrement remercier, pour leur contribution :

- Christophe Binot (Total)
- Marc Blangy (Axa France)
- Marc Blottière (Groupe Axa)
- Eric Boulay (Arismore)
- Jean-Baptiste Ceccaldi (Vistali)
- Raphaël Derbier (Vistali)
- Jean-Marc Guiol (Groupe Total)
- Jean Christophe Lalanne (Air France KLM)
- Pascal Léautier (Air France KLM)
- Guillaume Lemele (Axa France)
- Christophe Longepé (BNP Paribas)
- Laurent Mondemé (Air France-KLM)
- Lionel Pequignot (Total)
- Pierre Pezziardi (Octo Technology)
- Jacques Printz (Arts et Métiers)
- Danielle Rattier (BNP Paribas)
- Vincent Schattner (BNP Paribas)
- Philippe Tassin
- Bertrand Thyrion (Michelin)
- Michel Volle
- Denis Zandvliet (Value 360)

Le CEISAR assume seul la responsabilité de ces recommandations.

La première partie du document résume les recommandations d'actions concrètes pour accroître l'Agilité, classifiées par niveau de responsabilité: DG, Lignes Métier, Équipe Fondation, DSI, DRH, Achats.

Les parties suivantes approfondissent ces recommandations.

Table des matières:

1	Quelles actions concrètes pour améliorer l'Agilité.....	4
1.1	Ce que la Direction Générale peut faire :.....	4
1.1.1	Demander un bilan de la Transformation.....	4
1.1.2	Créer une Unité « Fondation » qui œuvre pour le « Bien Commun »	4
1.1.3	Fixer à l'Unité « Fondation » deux Objectifs : Agilité et Synergie	4
1.1.4	Définir le Fonctionnement de l'Unité « Fondations ».....	5
1.2	Ce que les Lignes Métier peuvent faire	5
1.2.1	Séparer Opérations et Transformation	5
1.2.2	Approfondir les choix de Progiciels.....	5
1.2.3	Les Équipes Projet-Solutions jouent le jeu avec l'Unité Fondations.....	6
1.2.4	Nommer des Chefs de Projet Gestionnaires et Constructeurs.....	6
1.3	Ce que la « DSI » peut faire	6
1.4	Ce que l'équipe Fondation peut faire.....	6
1.5	Ce que la DRH peut faire :	7
1.6	Ce que les Achats peuvent faire.....	8
2	Modèle du CEISAR et vocabulaire de la Transformation	9
3	Agilité: un avantage concurrentiel majeur	12
4	Processus et Fonctions	15
4.1	Progrès dans la Gestion des Projets	16
4.2	Quelles Fonctions d'Ingénierie des projets ?	16
4.3	Quelques améliorations d'Ingénierie pour accroître l'agilité	20
5	Un seul modèle partagé par Métier et IT	21
6	Définir et partager le langage Métier.....	23
7	Réutilisation de composants.....	24
8	Processus-Métier et Processus-Organisé.....	28
9	Approche Contractuelle ou Approche Coopérative	29
10	Les Rôles Métier de la Transformation.....	32
11	Organisation	35
12	Formation des Transformateurs Métiers	37

1 Quelles actions concrètes pour améliorer l'Agilité

Changer les Processus-Opérationnels via un Processus de Transformation est une tâche difficile. Changer les Processus de Transformation eux-mêmes dans le but de les rendre plus efficace est encore plus difficile parce qu'ils sont **plus complexes**.

Il n'est pas possible de déployer rapidement dans les Grandes organisations la cible que nous allons décrire. Une approche **progressive** est nécessaire. Nous donnons quelques recommandations qui modifient profondément les Organisations et les Méthodes, notamment

- en introduisant une **équipe Fondation** chargée d'accroître Agilité et Synergie,
- en recommandant de passer de l'Approche Contractuelle à l'**Approche Coopérative** qui associe Métier et IT dans la même équipe de projet
- en incitant les Chefs de Projets à se positionner non seulement en **Gestionnaires** de Projet, mais aussi en **Constructeurs** de Solutions d'Entreprise

Il est recommandé à chaque Entreprise de construire sa preuve à travers un **Projet Pilote** avant de les généraliser à tous.

1.1 Ce que la Direction Générale peut faire :

1.1.1 Demander un bilan de la Transformation.

La DG identifie bien des domaines de progression dans les Opérations comme l'optimisation de la Supply Chain, ou les Processus de Vente, ou les Processus de Back Office. Des moyens sont budgétés pour améliorer ces Processus Opérationnels.

Par contre la DG n'accorde généralement que peu de ressources à l'amélioration des Processus de Transformation. D'une part parce qu'elle a rarement conscience du coût global de la Transformation, d'autre part parce qu'elle n'imagine pas qu'il soit possible de fortement progresser dans ce domaine : les projets comportent des risques, on n'y peut pas grand-chose.

Il est donc nécessaire d'effectuer :

- Un bilan de ce que **coûte** la Transformation à l'Entreprise : non seulement les coûts des Développements Informatiques, mais aussi la charge pour les Acteurs Métier, les outils de Transformation, les coûts pour le management et la gouvernance associée, les formations pour les Acteurs opérationnels, les traitements en double, les migrations d'informations, ...
- Un bilan du degré de satisfaction sur les **délais** de création ou modification de Solutions, et sur la **qualité** des Solutions.

Ce bilan fera nécessairement apparaître que le coût global de la Transformation et que la frustration des Métiers vis-à-vis de l'Agilité des Solutions méritent que l'on prenne ce problème à bras le corps.

1.1.2 Créer une Unité « Fondation » qui œuvre pour le « Bien Commun »

Chaque Ligne Métier est jugée sur son efficacité. Il est donc illusoire de lui demander de travailler pour le « bien commun » de l'Entreprise: elle a naturellement un comportement égoïste.

C'est à la Direction Générale d'assumer la responsabilité de ce bien commun. Pour cela elle doit créer la **Structure « Fondations »** qui œuvre pour compte commun des Lignes Métier et qui regroupe d'excellentes compétences à la fois Métier et IT.

1.1.3 Fixer à l'Unité « Fondation » deux Objectifs : Agilité et Synergie

Le Rôle de l'Unité « Fondations » est de faire croître l'**Agilité** dans ses 2 caractéristiques « vite » et « bien », et d'identifier le bon niveau de **Synergie** entre Lignes Métier ou Filiales.

On ne doit pas disperser des équipes « Méthodologie », « Urbanisme ou Architecture d'Entreprise », « cartographie », « Glossaire d'Entreprise », « Sécurité », « Architecture », « Outils de Développement », « Infrastructure technique », « Interfaces », « Composants », ...Il n'existe **qu'une** Unité « Fondation » responsable de tout ce qui est mutualisable entre les Lignes Métier pour que le Chef de Projet Solution n'ait affaire qu'à **un** fournisseur interne dont le rôle est de :

- Construire et supporter une gamme unique de **Processus** et **Outils de Transformation** réutilisables par toutes les équipes de Transformation des Lignes Métier.

- Construire et supporter les **Modèles Réutilisables** par les Lignes Métier : définition d'un vocabulaire commun Métier pour faciliter dialogue Métier/IT, principes d'organisation, normes de sécurité, interface utilisateur, Composants Réutilisables, Infrastructure informatique ...
- Établir des instruments de mesure pour suivre l'évolution des **indicateurs d'Agilité** (rapidité du changement et qualité des Solutions) et de **Synergie** (partage de référentiels, réutilisation de Solutions ou de Composants).

1.1.4 Définir le Fonctionnement de l'Unité « Fondations »

- Nommer un **responsable Fondation** compétent et **reconnu** par les Lignes Métier : son rôle est extrêmement difficile.
- Lui demander de définir un **plan d'actions** qui soit **adapté** à l'Entreprise : on n'a pas le même besoin de Synergie dans l'Industrie ou dans les Services. En déduire les **moyens** à affecter à cette structure.
- Établir des règles de **refacturation** qui ne dissuadent pas les premières équipes Solution de réutiliser les Fondations.
- Définir des Règles de **Gouvernance** qui conduisent les Lignes Métier à réutiliser ce qui est proposé par l'Équipe Fondation : le Contrôle de conformité des Projets-Solutions vis-à-vis des Fondations doit avoir lieu **avant** validation du projet et du budget (voir livre blanc sur la Gouvernance), faute de quoi le chef de projet Solution expliquera que la réutilisation des Fondations est incompatible avec les planning et budgets qui ont déjà été décidés.
- Définir des Règles de **Gouvernance** qui conduisent l'Équipe Fondation à se comporter comme un **Fournisseur** des Lignes Métier et non comme une structure hiérarchiquement supérieure.
- Inciter l'Unité Fondation à **recupérer** et à packager des Composants issus des Lignes Métier.
- Suivre la **progression** des indicateurs d'agilité.

1.2 Ce que les Lignes Métier peuvent faire

1.2.1 Séparer Opérations et Transformation

Les préoccupations court terme l'emportent toujours sur le long terme : un incident de fonctionnement dans les Opérations nécessite une réaction **prioritaire** par rapport aux incidents qui peuvent affecter des Projets de Transformation en cours. Si une Ligne Métier souhaite favoriser l'Agilité, elle doit donc impérativement **séparer** les responsabilités de Transformation des responsabilités Opérationnelles.

1.2.2 Approfondir les choix de Progiciels

Une Solution déjà disponible est toujours plus séduisante qu'une Solution qu'il faut Construire. C'est ce qui explique le succès des Progiciels :

- Les responsables Métier peuvent visualiser au moment du choix une Solution **concrète** déjà disponible qui offre une partie des Processus souhaités. Si les Directions Métier ont subi des échecs dans le passé à l'occasion de projets mal conduits, ils peuvent se sentir soulagés et **rassurés** de s'appuyer sur une Solution qui s'exécute déjà avec succès dans d'autres Entreprises.
- Les responsables Métier voient s'**alléger leur tâche de définition des nouveaux Processus**.
- **Coûts** et **délais** devraient être réduits par rapport à une Solution indépendante puisque l'investissement et l'évolution du Progiciel sont mutualisés entre plusieurs Entreprises.

Certains Processus Métier sont stables : traiter un chèque, comptabiliser, établir un bulletin de paye,

On appelle **Solution de Commodité** une Solution dont les besoins sont **prévisibles** parce qu'elles touchent la Production, les Back-Offices, le réglementaire, ... Une « Solution de Commodité » va, bien sûr, évoluer au gré des changements d'organisation, des changements réglementaires, mais elle va évoluer dans un périmètre connu.

On appelle **Solution Concurrentielle** une Solution dont les besoins ne sont **pas prévisibles** comme : CRM, Business Intelligence, relations partenaires, Product Factory... A l'usage les acteurs Opérationnels font progresser leur Solution dans des directions qu'ils ne pouvaient prévoir a priori.

Pour chaque Solution, peser objectivement les 2 scénarios, **Solution-Progiciel** ou **Solution Spécifique** : il faut prendre en compte non seulement les Fonctionnalités disponibles, mais aussi coûts et délais globaux (en particulier pour les adjonctions ultérieures), possibilités de se différencier, rapidité d'évolution, insertion de l'Architecture du Progiciel dans l'Architecture d'Entreprise (duplication de données, interface utilisateur spécifique, spécialisation des Transformateurs...).

La Solution-Progiciel ne peut être une réponse aux Solutions Concurrentielles que si elle est Construite à partir de Composants réutilisables.

1.2.3 Les Équipes Projet-Solutions jouent le jeu avec l'Unité Fondations

- Jouer le jeu de la **coopération** avec l'Équipe Fondation : ne pas la considérer comme l'œil de la DG, mais comme un contributeur qu'il faut écouter et solliciter. **Utiliser positivement** ce que livre l'Équipe Fondation : avant de faire du spécifique, commencer par **réutiliser** les Composants existants, quitte à demander des améliorations ou des Composants complémentaires.
- Se comporter en force de **proposition** vis-à-vis de l'Équipe Fondation. Savoir « perdre du temps » à l'issue de chaque Projet-Solution pour **recupérer** ce qui peut devenir Composant réutilisable pour les autres Projets. Généralement un travail de l'Équipe Fondation est nécessaire pour packager le composant avant diffusion aux autres Lignes Métier.

1.2.4 Nommer des Chefs de Projet Gestionnaires et Constructeurs

Choisir et Former des chefs de Projet qui ne sont pas que des **Gestionnaires**, mais aussi des **Constructeurs**. Protéger les chefs de Projet d'un excès de tâches de Gestion

1.3 Ce que la « DSI » peut faire

La DSI doit appuyer son équipe Fondations : voir paragraphe suivant.

En outre la DSI-Groupe, de part son rôle transversal, doit appuyer la Direction Générale dans la recherche du Bien Commun : elle a un rôle clé pour définir le bon niveau de Synergie entre Lignes Métier, pour promouvoir des économies d'échelle, pour communiquer et former.

Elle doit aussi éclairer les choix **d'externalisation** des développements informatiques : ils s'appuient sur un coût inférieur du jour-homme, mais ils empêchent l'Approche Coopérative et entraînent souvent la perte de connaissance du Modèle de l'Entreprise.

Elle doit lutter contre la « procéduralisation » excessive qui laisse à penser que la bonne Gestion garantit la bonne Construction.

Ce n'est pas parce que « qualité », « sécurité », « ergonomie », « urbanisme », « organisation », « méthode », « intégration », « performance », « tests »...sont importants qu'il faut avoir autant de Rôles différents. La **multiplication des Rôles** accroît les problèmes de suivi, de coordination et d'intégration, et **déresponsabilise** les équipes. Chercher plutôt des intervenants de qualité **capables d'assumer plusieurs rôles**.

1.4 Ce que l'équipe Fondation peut faire

- Se comporter vis-à-vis des Lignes Métier comme un **fournisseur** vis-à-vis de ses clients et non comme une autorité supérieure.
- Définir l'**Approche** « Construction de Solution » qui recommande les points suivants :
 - **Grouper** Constructeurs Métier et IT dans une **seule Équipe Projet-Solution**.
 - Définir une **seule** Approche partagée par Métier et IT, ce qui n'est pas encore le cas dans toutes les Entreprises.
 - Privilégier l'**Approche Coopérative** chaque fois que possible : plutôt que le recensement **exhaustif** des besoins, privilégier la Construction d'une **Architecture** de Solution capable d'accueillir des incréments successifs au fur et à mesure de la maturation des besoins.
 - Vérifier que le **Problème** est bien défini : pour que tous les Acteurs d'un Projet prennent les bonnes décisions quotidiennes, il faut systématiquement se référer au Problème à résoudre, ce qui suppose que chacun ait compris et retenu le Problème pour bien **aligner** la Solution sur le problème. Il faut donc en donner une **explicitation simple (une page)** et la communiquer à tous, par exemple en l'affichant dans tous les bureaux des équipes de Transformation.

- Établir un **glossaire rigoureux des Entités du Métier** réutilisable d'un projet à l'autre. Demander à tous les intervenants de l'utiliser et à l'affiner progressivement. Ce Glossaire est **prolongé par le Modèle** des Entités du Métier (relations, héritages, identifiants, cycle de vie de chaque Entité) qui représente l'Architecture des Solutions.
- **Dissocier** Cœur-Métier et **Organisation** dans l'analyse des Processus pour bâtir des Solutions pérennes qui supportent des Organisations successives.
- Rechercher des **Outils** de Modélisation « Round-Trip » **communs** au Métier et à l'IT (pour Modèles d'Informations, Modèles de Processus, Moteur de Règles, Interface Utilisateur) : on Modélise le Métier et il génère automatiquement une partie du Logiciel, on Modifie le Logiciel et il met à jour automatiquement le Modèle Métier.
- Elle gère les **Solutions Progiciels réutilisables** entre différentes Compagnies d'un Groupe : choix, suivi des Versions, construction d'outils pour Interfaces et Migration.
- Pour les **Solutions Spécifiques**, elle construit progressivement des **Composants Réutilisables**
 - Se donner une **large ambition**, même si on commence petit : on peut atteindre des taux de réutilisation de 70% pour les Solutions Spécifiques
 - Prendre en compte **tous** les types de Composants : accès aux Informations, flux entre Solutions, composants SOA, patterns, utilisation de Moteur de Règles, de moteur de workflow...
 - Acquérir le **savoir-faire** dans la construction de Composants
 - Rendre les Composants aisément **accessibles** et **compréhensibles**
- **Supporter** les Composants Réutilisables auprès des Constructeurs de Solutions comme le ferait un éditeur de logiciels et non comme un « gardien du temple »
 - **Former** et **Aider** (coaching, assistance, revues) les Constructeurs de Solution : Métier et IT
 - Animer une **hotline**
 - Recueillir les **demandes** de correction et d'évolution
 - **Observer** l'utilisation des Composants pour en améliorer concrètement l'usage pratique
- Définir la **road-map** pour **simplifier l'Architecture d'Entreprise globale**. Un projet sera plus simple s'il s'inscrit dans une Architecture d'Entreprise bien structurée. La clarté du périmètre, la précision des interfaces avec les autres Solutions, la réutilisation de Fonctions d'accès aux Informations sont autant d'atouts pour concentrer l'énergie du chef de Projet sur le Modèle de Solution, et non sur son environnement. Toute la difficulté consiste à établir une stratégie de simplification progressive pour que chaque Projet contribue à cette simplification globale. Cette première piste a été définie dans le livre blanc du CEISAR « **Simplify Legacy Systems** » : nous proposons au lecteur intéressé de télécharger ce livre blanc à partir de www.ceisar.org

1.5 Ce que la DRH peut faire :

- Si l'on souhaite affecter des personnes talentueuses à la Transformation **Métier**, il faut **valoriser** (salaire, formation, reconnaissance) **la fonction de Transformation** et ne pas définir comme critère essentiel de reconnaissance la taille des équipes (Opérationnelles) que l'on dirige.
- **Sélectionner** des Acteurs de la fonction « Constructeur de Solution-Métier » de grande qualité, capables d'effectuer un maximum de tâches de Transformation pour éviter la multiplication des Rôles (voir ci-dessus).
- Offrir des **Formations** adaptées à ces Constructeurs pour les faire murir rapidement :
 - Comprendre ce qu'est **l'Architecture d'Entreprise** : alignement avec la Stratégie, décomposition entre Opérations et Transformations, décomposition entre Monde réel et Modèle, recherche de Synergies par Réutilisation de Modèles et Partage de Ressources
 - Comprendre ce qu'est un **Modèle d'Entreprise** : Modèle d'Informations, Modèle d'Actions (Processus et Fonctions) et Modèle d'Acteurs (Acteurs humains et ordinateurs)
 - Savoir exécuter un Projet : **décomposition en 8 Fonctions d'Ingénierie** (Comprendre le Contexte, Définir le Problème, Construire la Solution, Vérifier la Solution, Adapter la Solution, Configurer la Solution, Déployer la Solution, Opérer la Solution).
 - Comprendre **Approche Coopérative** et **Approche Contractuelle**

- **Savoir être** : préparer les Transformateurs Métier à séparer l'essentiel, la structure du détail, et à ne pas compter que sur le respect « scolaire » du Processus de Transformation pour faire réussir leur projet.
- **Connaissance des Fondations** : il faut d'abord expliquer qu'il est possible de Construire des Solutions pour des besoins très **spécifiques** à l'aide de Fondations **communes**. Une fois ce principe acquis il faut expliciter la variété des Composants que l'on peut réutiliser.
- Détailler chaque **Fonction d'Ingénierie** dans sa composante Métier: définir le Problème, construire les Entités-Métier, les Fonctions, les Processus, vérifier, configurer, former,...
- Un bon chef de Projet doit avoir à la fois des qualités de Gestionnaire et de Constructeur. On a davantage valorisé les premières parce qu'elles suffisaient dans une Approche Contractuelle adaptée aux **Solutions de Commodité**. Elles sont aujourd'hui insuffisantes pour appliquer une Approche Coopérative adaptée aux **Solutions Compétitives** : la qualité de l'Architecture de la Solution est clé. Comme les bons Constructeurs sont plus rares que les bons Gestionnaires, nous recommandons de choisir comme « Chef de Projet » des Solutions Compétitives non pas les meilleurs gestionnaires, mais les **meilleurs Constructeurs**, quitte à faire épauler ces Constructeurs par des personnes chargées de l'administration du Projet. C'est un changement culturel important dans les Organisations qui ont plutôt mis en avant les Rôles de Gestionnaires.
- Maintenir un noyau d'excellentes **compétences IT** qui permettent à l'Entreprise de conserver la connaissance de son Modèle. L'externalisation croissante de la Modélisation IT va être un frein à la mise en place d'une Approche Coopérative dans les Entreprises. Il serait souhaitable que toute Entreprise conserve un **noyau de compétences IT reconnues**, faute de quoi elle se retrouvera un jour dépendant de fournisseurs qui auront fait évoluer leurs Solutions IT sans que l'Entreprise ne conserve la connaissance de son Modèle : le Modèle Métier d'origine n'a pas été mis à jour et a perdu sa valeur, alors que les **Logiciels** ont été mis à jour successivement par le Fournisseur qui est le seul à les comprendre.
- Favoriser les **passerelles** entre Équipe Fondation et Lignes métier.

1.6 Ce que les Achats peuvent faire

- Mettre en valeur la **Qualité**, et pas seulement le prix, dans les procédures de sélection des fournisseurs qui contribuent à la Transformation : contrairement aux Opérations, les écarts de productivité dans la Transformation peuvent aller de 1 à 10.
- Favoriser les partenariats avec les petites structures **innovantes**. Les grands fournisseurs ont le plus souvent des Approches qu'ils ont rendues **conformes** à ce que demandent les grandes Entreprises. Certains petits fournisseurs atypiques peuvent apporter un vent nouveau.
- Trouver de nouvelles formes de partenariat :
 - la **régie** est peu satisfaisante : on achète de la ressource humaine au plus faible coût, indépendamment du résultat, ce qui a conduit certains sponsors à interdire cette forme de partenariat
 - le **forfait global** n'est plus satisfaisant : il nécessite de modéliser **toutes** les exigences métier avant de contracter. C'est de plus en plus difficile lorsqu'on passe des Solutions de Commodité à des Solutions Concurrentielles dont le périmètre est variable et qui vont nécessiter une approche par Versions.
 - Il faut imaginer des partenariats basés sur les principes suivants :
 - la Solution doit être **bien construite** pour accepter les adjonctions ultérieures
 - le coût et le délai sont maîtrisés pour chaque **Version** successive : elles peuvent constituer une succession de forfaits intermédiaires
 - le périmètre fonctionnel de chaque Version est décomposé en une partie obligatoire et une partie **variable** ajustable au fur et à mesure de l'avancement du projet pourvu que coûts et délais soient bien respectés.

2 Modèle du CEISAR et vocabulaire de la Transformation

Le CEISAR a défini un Modèle d'Entreprise (voir www.ceisar.org) dont nous rappelons les grandes lignes.

L'Architecture d'Entreprise définit comment une Entreprise **Opère** (le « présent ») et se **Transforme** (le « futur ») : comment des **Acteurs** (Humains ou Ordinateurs) effectuent des **Actions** (Processus, Fonctions) à l'aide **d'Informations**.

L'Architecture ne définit pas la **stratégie** de l'Entreprise (quels produits, pour quels marchés, avec quels partenaires, sur quels territoires), mais cette stratégie est un « input » pour l'Architecture d'Entreprise.

Lorsque le fonctionnement de l'Entreprise devient trop complexe il faut **Modéliser ses Opérations** pour qu'elles s'effectuent efficacement: le Modèle permet de comprendre, mémoriser, communiquer, former, guider les Acteurs.

La modélisation peut être **globale** via des Cartes (cartes d'Entités du métier, cartes de Processus, cartes de Fonctions, cartes de Blocs, cartes de Services...) ou **détaillée**.

Le modèle détaillé comprend 3 parties :

- Le **Modèle d'Acteurs** : le Modèle des Acteurs-Humains est appelé « **Rôles** » (Vendeur, Producteur, Administratif), le Modèle des Acteurs-Ordinateurs est appelé « **Configurations-IT** » (Matériels, Logiciels, réseau).
- Le **Modèle d'Actions** décrit la liste des instructions pour bien exécuter une Action : c'est un **Guide Utilisateur** pour les Acteurs-Humains et un **Logiciel** pour les Acteurs-Ordinateurs. On distingue
 - le Modèle de **Processus** (« Vendre », « Produire », « Gérer »)
 - le Modèle de **Fonctions** qui composent les Processus (« Tarifier », « Imprimer »).
- Et les **Modèles d'Informations** des Client, Produit, Contrat, Compte...

Métier = Cœur-Métier + Organisation.

On appelle « **Cœur-Métier** » tout ce qui définit le Métier indépendamment de l'Organisation choisie par l'Entreprise : définition des Produits (Biens ou Services), Informations sur les Clients, choix des Partenaires, règles de tarification.

On appelle « **Organisation** » l'organigramme, le Rôle des Acteurs, les autorisations, les devoirs, la répartition des Activités.

Cette distinction fondamentale doit aider à construire une Solution qui s'appuie sur le Cœur-Métier et supporte des Organisations différentes (successives ou parallèles).

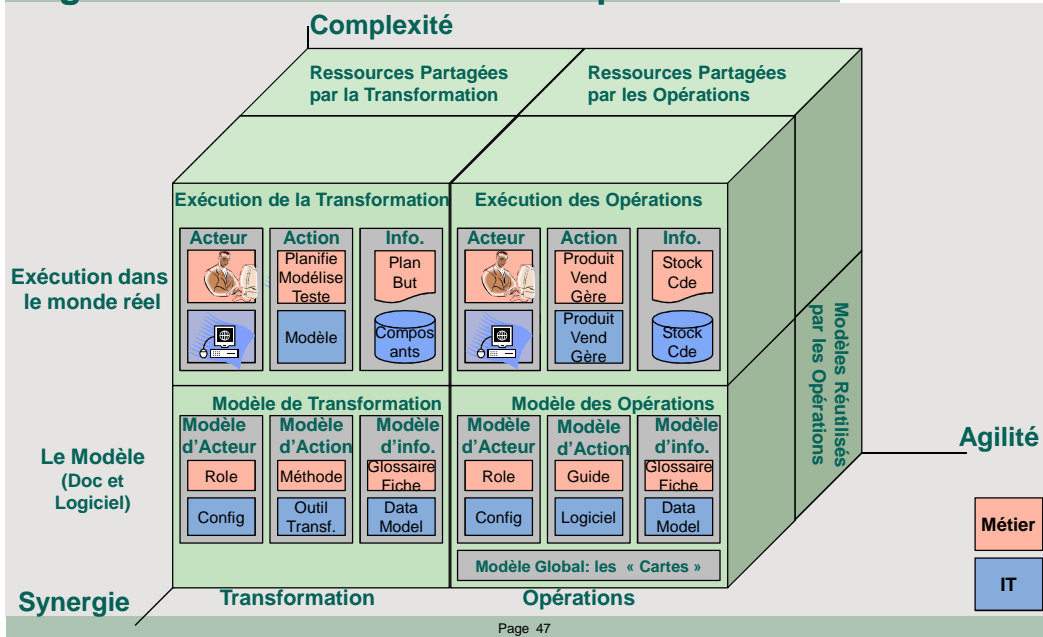
L'Entreprise doit faire évoluer son Modèle d'Opération pour faire face aux évolutions.

La Construction (création, acquisition ou modification d'un Modèle d'Opération) et son Déploiement s'appelle la **Transformation**. Il ne s'agit plus de Vendre ou de Produire, mais de gérer des Projets, de définir une « Road-Map », de changer un tarif, de modifier un Processus, d'ouvrir une agence...

De même que les Opérations sont exécutées grâce à un Modèle d'Opération, de même la Transformation est exécutée à l'aide de son **Modèle de Transformation** qui décrit :

- Le **Modèle d'Acteurs** : Rôles de « Chef de Projet », « Architecte », « Maîtrise d'Ouvrage »...pour les Acteurs Humains
- Le **Modèle d'Actions** : Modèle de **Processus** (ex : « Construire une Solution ») ou Modèle de **Fonctions** qui composent les Processus (« décrire un Problème », « Analyser un Processus », « Tester une Fonction »)
- Et les **Modèles d'Informations** (Projet, Test, Planning)

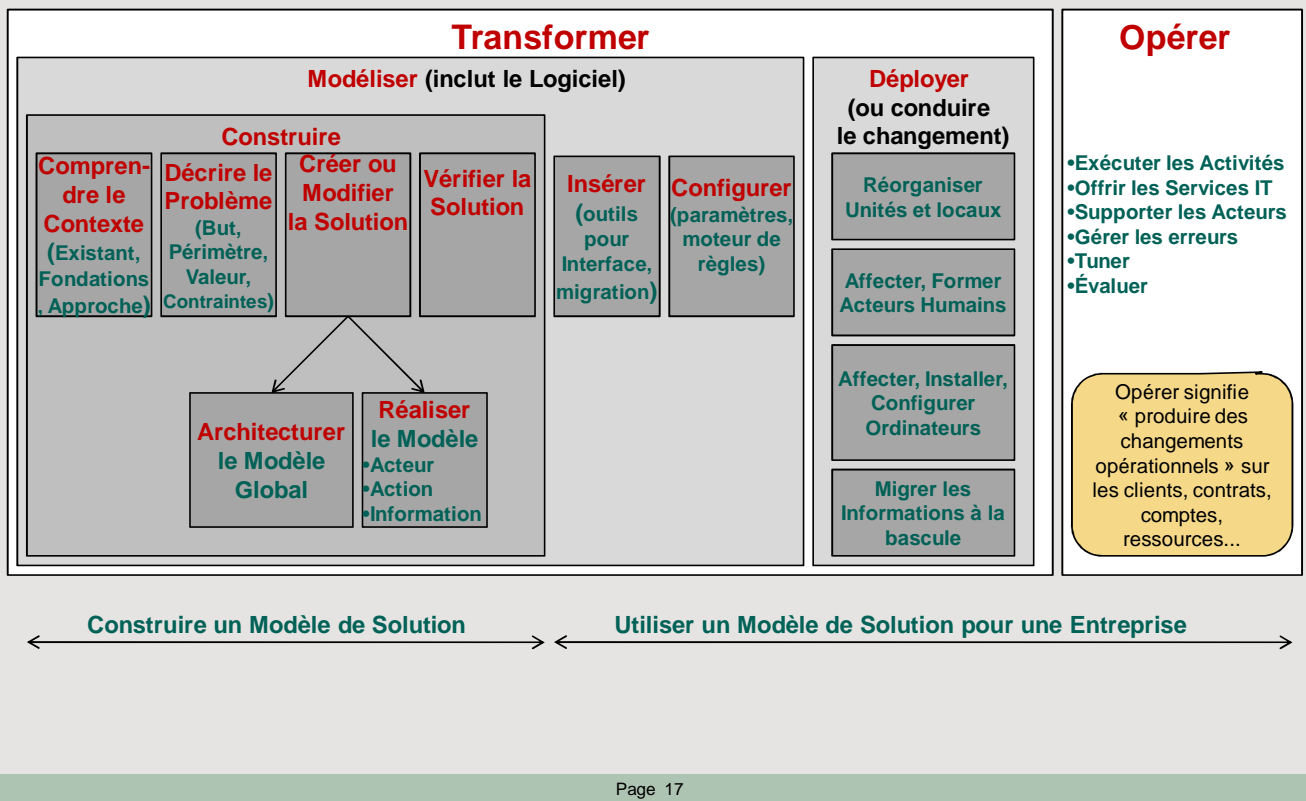
Un seul Modèle d'Entreprise pour Métier, Organisation et IT avec vues adaptées.



Le Cube du CEISAR est un cadre de rangement dont les 3 dimensions correspondent aux 3 plus grands challenges des Entreprises d'aujourd'hui : **Complexité**, **Agilité** et **Synergie**.

- gérer la Complexité croissante nécessite de séparer le monde réel de sa **Modélisation**
- gérer l'Agilité nécessite de séparer Opérations et **Transformation**
- gérer la Synergie nécessite de **Partager** des Ressources (ressources humaines, ressources informatiques ou référentiels d'informations) et de **Réutiliser** des Modèles (Modèles de Solutions ou Composants ou Approches)

Dans ce Livre Blanc, nous nous intéressons à la partie gauche du Cube : la Transformation.



La Transformation inclut la **Modélisation** et le **Déploiement** du nouveau Modèle dans les Opérations. A titre d'exemple, ouvrir une nouvelle Agence à l'aide d'un Modèle déjà existant fait quand même partie de la Transformation parce qu'il s'agit de Déploiement d'un Modèle.

La Modélisation inclut non seulement la **Construction** du Modèle, mais aussi l'**Insertion** du Modèle dans l'Architecture d'Entreprise (Interfaces et outils de migration) et la **Configuration** de la Solution à l'aide de paramètres et de moteurs de règles.

3 Agilité: un avantage concurrentiel majeur

Réactivité=Opérer vite et bien et **Agilité**= Transformer vite et bien.

Servir rapidement un Client, faire rapidement tourner son stock, régler rapidement ses fournisseurs... s'appelle « Réactivité des Opérations » : une Entreprise est réactive si son Modèle d'Opérations lui permet de réagir vite et bien aux sollicitations de ses Clients, Partenaires, Fournisseurs...

Mais si le Modèle lui-même doit être transformé vite et bien, il ne s'agit plus d'Opérations mais de **Transformation** : la **Réactivité** est l'Art **d'Opérer vite et bien** alors que l'**Agilité est l'Art de Transformer vite et bien**.

Vite et Bien

Pour répondre aux **Problèmes** de l'Entreprise, on crée ou on modifie des **Solutions** : comment des **Acteurs** Humains ou des Ordinateurs doivent **Agir** à l'aide **d'Informations**.

La Création de la Version 1 de la Solution est plus coûteuse que chacune des Versions suivantes puisqu'elle inclut la Construction de l'**Architecture** de la Solution qui sera conservée pour les Versions suivantes. Mais, compte tenu de la durée de vie des Solutions, la somme des coûts des Versions 2 à n représente 2 à 3 fois le cout d'origine de la Version 1. L'Agilité est donc non seulement l'art de Construire **rapidement** la première version de Solution, mais aussi l'art de Construire, dès cette 1^o Version, une Architecture de Solution **efficace** pour que les Versions suivantes en bénéficient. L'Agilité à l'instant T ne doit donc pas hypothéquer l'agilité à l'instant T+1 : c'est la définition du développement durable appliqué aux Systèmes d'Information.

La Transformation est plus complexe que les Opérations

On sait Modéliser le Processus Opérationnel de « Commande », mais on sait mal Modéliser le Processus de Transformation « Construction d'une Solution ».

Les incertitudes sont nombreuses :

- Incertitudes sur le livrable
- Incertitudes sur l'Architecture de la Solution
- Incertitudes sur le déroulement du Projet
- Incertitudes sur le degré d'acceptation par les Acteurs Opérationnels.

Elles nécessitent que le chef de Projet ait les talents nécessaires pour prendre des décisions dans un environnement incertain.

On a progressivement réussi à Modéliser des Fonctions de **Gestion** : gouvernance, planning, budget, ressources, communication...

On n'a pas encore bien réussi à Modéliser les Fonctions **d'Ingénierie** : comment définir le Problème, comment Architecturer une Solution, comment réutiliser des Composants, comment Construire une Solution qui supporte différentes Organisations.

Agilité = qualité majeure d'une Entreprise

L'Agilité est recherchée par toutes les Entreprises parce qu'elles sont confrontées à un environnement de plus en plus mouvant qui les contraint à un changement continu.

Comme il est impossible de garantir que les innovations initiées dans l'Entreprise seront meilleures que celles des concurrents, il semble que la stratégie la plus sûre pour accroître l'efficacité soit non seulement d'innover par soi même, mais surtout de mettre au point des Processus de Transformation plus **agiles** que ceux des concurrents, pour implémenter rapidement les innovations venues d'ailleurs et corriger ses points faibles plus rapidement que ses concurrents. En ce sens, l'Agilité devient la **qualité majeure d'une Entreprise** : si la productivité ou la qualité ou la fiabilité sont faibles, si les produits sont trop classiques ou trop chers, ... l'Agilité permet de les améliorer rapidement. Il suffit d'identifier ses faiblesses et de bénéficier de son Agilité pour corriger le tir.

Si la Grande Entreprise est agile, elle peut acquérir des start-up, copier des innovations réussies par certains concurrents, et obtenir un véritable avantage concurrentiel **non pas parce qu'elle « voit loin »**, **mais parce qu'elle « bouge vite »**. La Stratégie n'est plus basée sur la prévision de plus en plus aléatoire du futur, mais sur la saisie d'opportunités dont l'Entreprise tire rapidement parti parce qu'elle s'appuie sur des Fondations qui la rendent plus agile que ses concurrentes.

Pour se doter de Solutions de Commodités, les Entreprises ont développé une Approche Contractuelle qui privilégie la sécurité ou la fiabilité à l'agilité.

On a commencé à déployer des « **Solutions de Commodité** » qui automatisaient les Processus bien définis de Production, de back office, de réglementation. Les objectifs essentiels étaient la productivité, la fiabilité, la visibilité.

Les Processus de Transformation de l'Entreprise qui sont adaptés aux « Solutions de Commodité » ne sont pas toujours suffisamment agiles, souvent parce qu'ils ont été définis à une époque où la pression était moins forte, où les besoins étaient plus facilement prévisibles, où les technologies balbutiantes nécessitaient de lourds contrôles, où l'usage des outils de développement informatiques nécessitaient de définir tous les besoins avant de faire.

Cette lenteur de la Transformation interne, qui a pour conséquences un accroissement des coûts associés, est caractérisée par l'établissement d'un Contrat qui contient le Modèle Métier à traduire en Modèle IT, la séparation des équipes Métier et IT, et une multiplicité de Rôles qui nécessitent de gérer leurs relations,

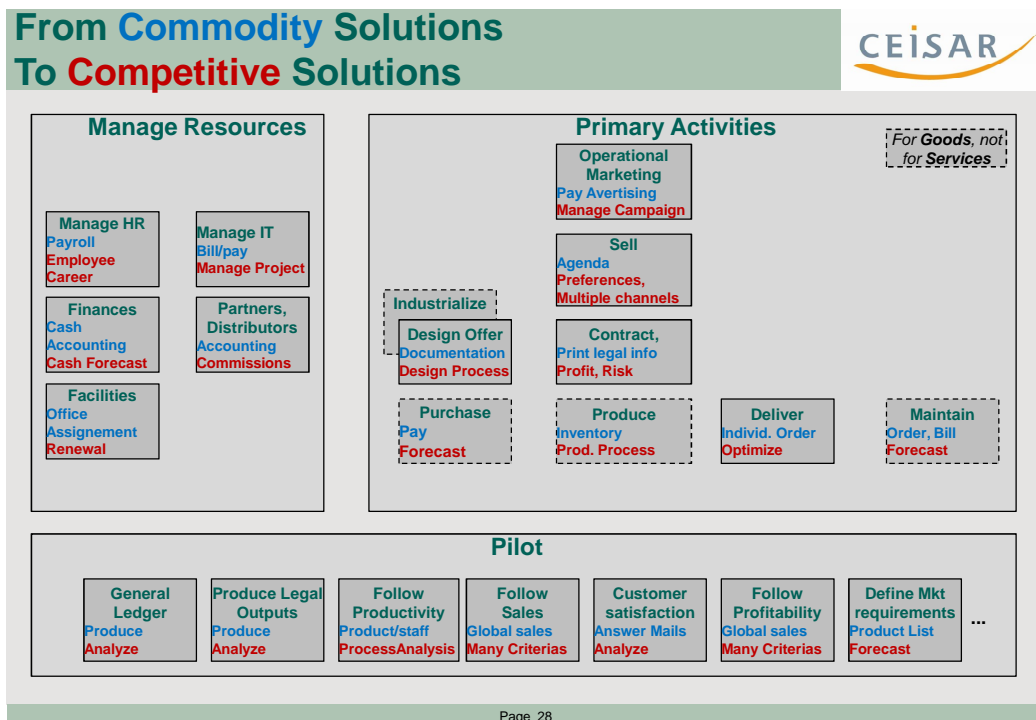
Cette lourdeur a permis à une industrie du **Progiciel** de se développer rapidement en se présentant comme une alternative aux Solutions spécifiques. Les ERP ont résolu les besoins des financiers, des ressources humaines, de la gestion de production, de la productivité des back office, du reporting,... parce que les besoins de commodité n'étaient pas très éloignés d'une Entreprise à l'autre, et que l'on pouvait donc construire une Solution commune à plusieurs Entreprises.

L'approche Contractuelle n'est plus adaptée aux nouvelles Solutions Compétitives

On bascule progressivement vers des « **Solutions Compétitives** » qui peuvent donner un avantage concurrentiel à l'Entreprise. Business Intelligence et aide à la décision, CRM et nouveaux outils de front-office, conception de produit et time to market, processus de bout en bout et intégration des partenaires ou des clients (« l'Entreprise étendue »), multiplicité des canaux de distribution sont autant de domaines où les Entreprises souhaitent davantage d'agilité.

Elles sont caractérisées par l'impossibilité a priori de définir tous les besoins. Il faut

- une **nouvelle approche** : ne pas contraindre le Métier à tout spécifier, procéder par versions successives, mettre l'accent sur la qualité de la Construction de la Solution pour qu'elle accepte les Versions ultérieures.
- une **nouvelle forme de réutilisation** : non plus réutiliser des Solutions-Progiciel que l'on peut adapter à la marge, mais réutiliser des Composants que l'on peut assembler pour construire des Solutions innovantes, différentiantes.



Mais, de façon surprenante, les mêmes Entreprises qui ont su si bien optimiser leurs **Processus Opérations** de back office ou de Production, n'imaginent pas qu'il soit possible de progresser fortement dans l'agilité et beaucoup n'envisagent pas de refondre des **Processus de Transformation** qu'elles ont eu tant de mal à faire respecter.

L'objectif de ce document n'est pas tant de faire le bilan des Processus de Transformation actuels sur lesquels beaucoup a déjà été écrit, que de convaincre qu'il est possible de progresser soit par des actions progressives, soit par un « bond en avant » pour se mettre en position de Construire plus rapidement des Solutions Compétitives.

4 Processus et Fonctions

Différents Processus de Transformation

A côté des Projets Fondations et des Projets Solution, un de nos Sponsors a créé la catégorie « Projet Innovant ». Ces Projets comportent une part de **risque** plus importante et ne peuvent réussir s'ils doivent appliquer à la lettre les Processus de Transformation classiques. Qui plus est, ce sont des Projets qui sont généralement écartés par les Processus de Gouvernance classiques qui ne favorisent pas la prise de risque (voir livre blanc sur la Gouvernance).

Aujourd'hui 10% de leurs Projets sont des Projets innovants. Ils procèdent d'une gouvernance différente et appliquent un Processus de Transformation plus souple.

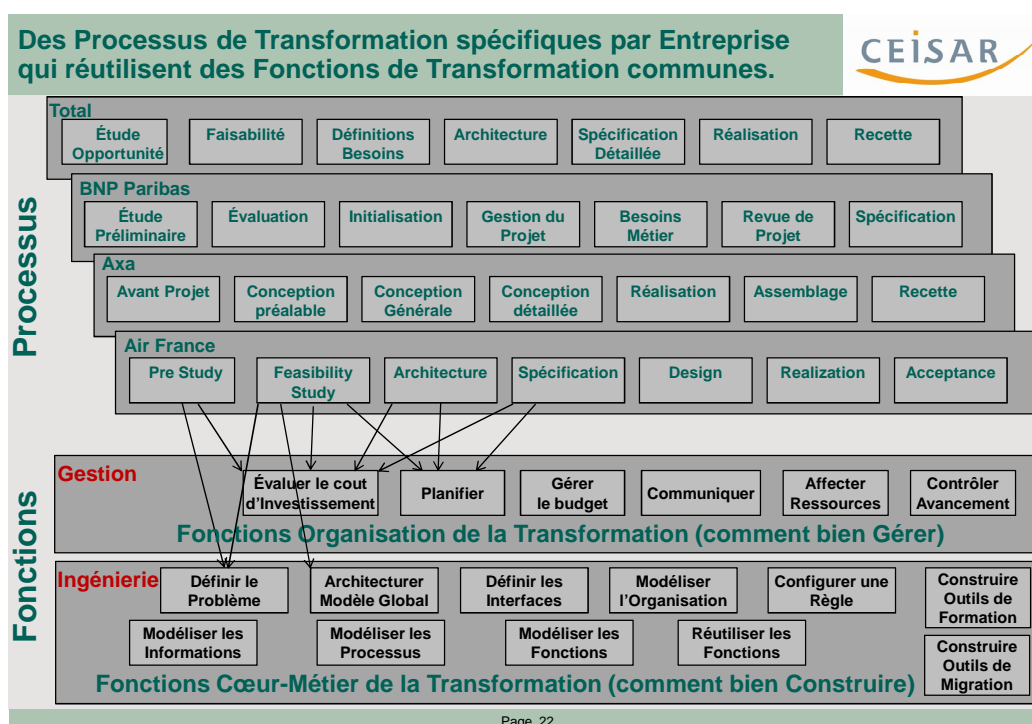
Il existe donc **différents Processus de Transformation** : Construire une Solution, Construire une Solution innovante, mais aussi Modifier une Solution (évolution ou correction de « bug »), Construire des Fondations, Définir une Road-Map,...

Ce document développe surtout le Processus de Construction ou de Modification de Solutions.

Un Processus de Transformation est décomposé en Phases

On sait bien formaliser les Processus **Opérationnels** comme le Processus de commande et on les automatise progressivement: c'est beaucoup plus difficile pour les Processus de **Transformation** comme le Processus de Construction d'une Solution, compte tenu des incertitudes de tout projet.

De nombreux progrès ont néanmoins été accomplis : chaque Entreprise a défini son propre Processus de Transformation en le décomposant en **Phases** progressives.



Chaque Phase fait appel à des Fonctions

Chacune des Phases de Processus utilise des **Fonctions** de Transformation. Par exemple la Phase « Pre Study » d'Air France utilise les Fonctions « Définir le Problème » et « Évaluer le coût d'investissement ».

Chaque Fonction peut être activée **plusieurs fois** dans le même Processus: par exemple la Fonction « Évaluer le coût d'Investissement » est affinée dans chaque Phase pour donner des résultats de plus en plus précis. De même la Fonction « Modéliser les Processus » est utilisée dans plusieurs Étapes pour détailler progressivement les Processus concernés.

Fonctions d'Ingénierie et Fonctions de Gestion

On distingue deux types de Fonctions :

- les **Fonctions d'Ingénierie** (comme « Modéliser les Processus ») pour bien **Construire** la Solution
- les **Fonctions de Gestion** (comme « Planifier ») pour bien **Gérer** le Projet.

Les Fonctions d'Ingénierie sont les **Fonctions Cœur-Métier** de la Transformation : elles doivent être accomplies quelque soit l'Organisation en place. Les Fonctions de Gestion sont les **Fonctions Organisation** de la Transformation : elles dépendent étroitement de l'Organisation. Elles sont donc **spécifiques** à chaque Entreprise, alors que les Fonctions d'Ingénierie sont **universelles**.

4.1 Progrès dans la Gestion des Projets

De nombreux projets souffrent d'une mauvaise gestion de projet : les Phases ne sont pas formalisées, les livrables ne sont pas fournis, le planning est incomplet, les affectations d'acteurs ne sont pas anticipées, le reporting est oublié, les décisions ne sont pas formalisées...

Les organismes de définition des méthodologies comme CMMI ou Open Group (Togaf) ont bien approfondi les Fonctions de Gestion de la Transformation. Les Entreprises ont sensiblement progressé en tirant parti de cette tendance et en maîtrisant de mieux en mieux la Gestion de Projet. On est passé d'une période d'improvisation à une période d'amélioration continue qui est bien illustrée par les niveaux de maturité du CMMI qui définissent comment une Entreprise progresse dans sa gestion de projet.

D'après les résultats des études publiées par le Standish Group (<http://www.standishgroup.com>), le taux de succès des projets informatique suit une tendance à la hausse depuis les 10 dernières années. En effet, près de 30% des projets sont qualifiés de succès : ils sont livrés à temps, à l'intérieur des budgets et des spécifications originales. Cependant, plus de la moitié des projets sont qualifiés de mitigés : ils sont livrés et opérationnels, mais ont moins de fonctions que prévues dépassent les budgets et/ou les échéanciers. Une bonne nouvelle à l'horizon, la tendance des projets qualifiés d'échecs, c'est-à-dire abandonnés en cours de route ou les résultats livrés ne sont jamais utilisés, est en baisse et se situe maintenant à moins de 20%.

Les statistiques des 10 dernières années sont les suivantes :

1995 : Succès 16% ; Mitigés 53% ; Échec 31%
 2000 : Succès 28% ; Mitigés 49% ; Échec 23%
 2004 : Succès 29% ; Mitigés 53% ; Échec 18%
 2006 : Succès 35%, Mitigés 46% ; Échec 19%

Les Progrès ont surtout porté sur la **Gestion**. A titre d'exemple, les recommandations du Standish Group sont essentiellement des recommandations de bonne Gestion :

- L'engagement de la direction
- L'implication des utilisateurs
- L'expérience du chef de projets
- La formulation des objectifs métier
- Une envergure limitée aux besoins essentiels
- Une infrastructure technologique normalisée
- Des spécifications précises et stables
- Des méthodologies formelles et utilisées
- Des estimations fiables et rigoureuses
- Autres : découpage des livraisons, compétence du personnel, etc.

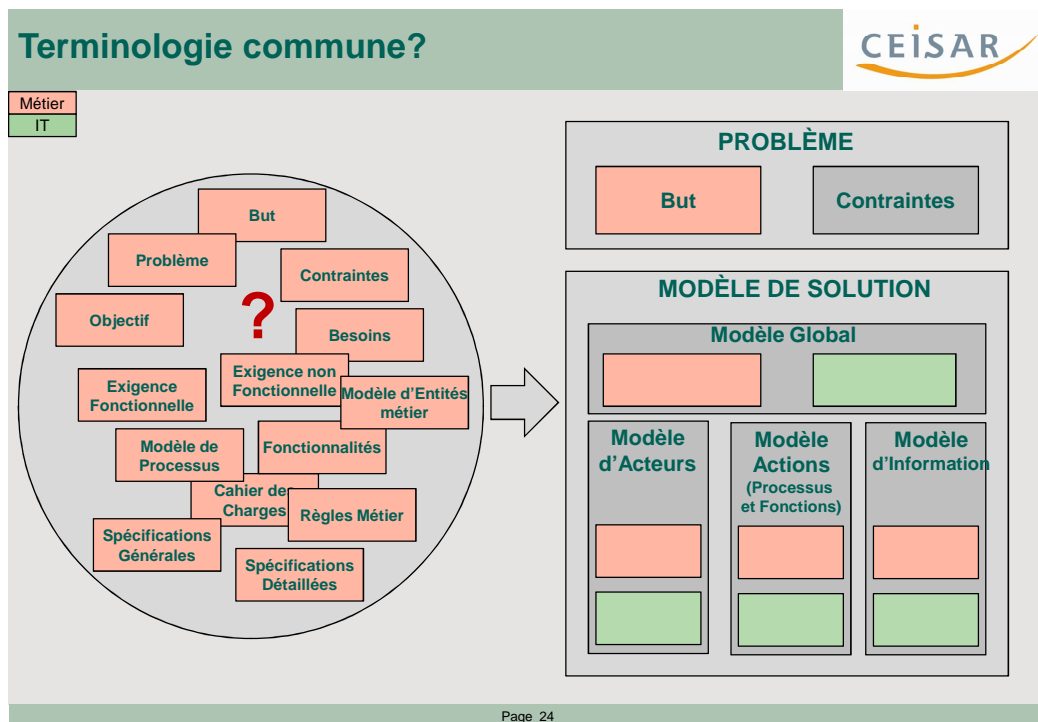
Il faut poursuivre dans cette voie d'amélioration de la Gestion, mais ne pas tomber dans l'excès : trop de tâches de Gestion empêchent le chef de projet de se consacrer aux Fonctions d'Ingénierie. On peut **bien Gérer** le projet d'une Solution **mal Construite**.

4.2 Quelles Fonctions d'Ingénierie des projets ?

Nous proposons de donner quelques pistes pour améliorer l'**Ingénierie** des Projets parce que ce domaine est aujourd'hui moins développé que celui de la **Gestion**.

Commençons par définir les **Fonctions d'Ingénierie** classifiées de 1 à 7 (voir plus loin) : elles décrivent le **Processus d'Ingénierie** et non le **Processus Organisé**. Par exemple on ne voit pas apparaître la **Phase Préalable** qui déclenche les Fonctions « Décrire le But » puis « Architecturer le Modèle Global »,

ni la **Phase de Faisabilité** qui déclenche et approfondit les mêmes Fonctions: on ne représente donc les Fonctions d'Ingénierie qu'une seule fois même si elles sont déclenchées plusieurs fois.
Le Processus et les Fonctions d'Ingénierie de Transformation sont **communs** à toutes les Entreprises.

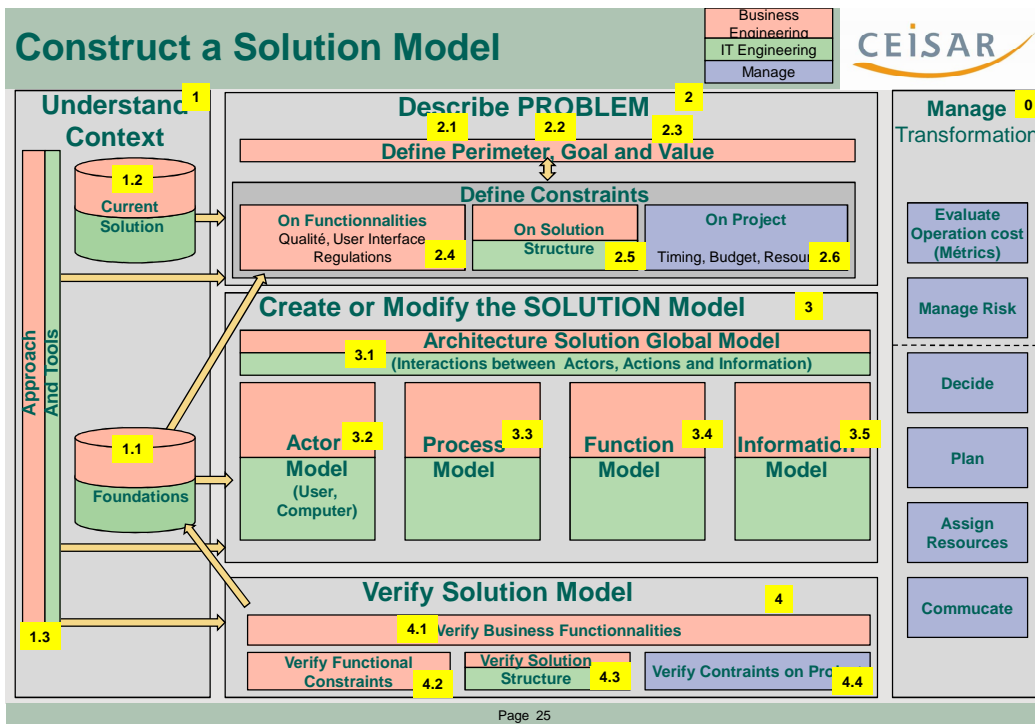


Les termes « Objectif », « But », « Besoin », « Exigence », « Fonctionnalités », « Spécification », « Conception » sont reclassifiés soit dans « Problème » soit dans « Solution » :

- **Problème** : on y retrouve But, Objectif, Contraintes définis par le Sponsor qui décide et paye
- **Modèle de Solution** : on y retrouve ce qui est Construit en réponse au Problème, c'est-à-dire le Modèle Global, les Modèles d'Action (Processus et Fonctions), les Modèles d'Information, les Modèles d'Organisation aussi bien pour la partie **Métier** que pour la partie **IT**.

Le Processus d'Ingénierie ne prend donc pas parti sur le fait qu'il faut ou non séparer le Modèle Métier et le Modèle IT : les Approches Contractuelles les distinguent alors que les Approches Coopératives les confondent (voir plus loin).

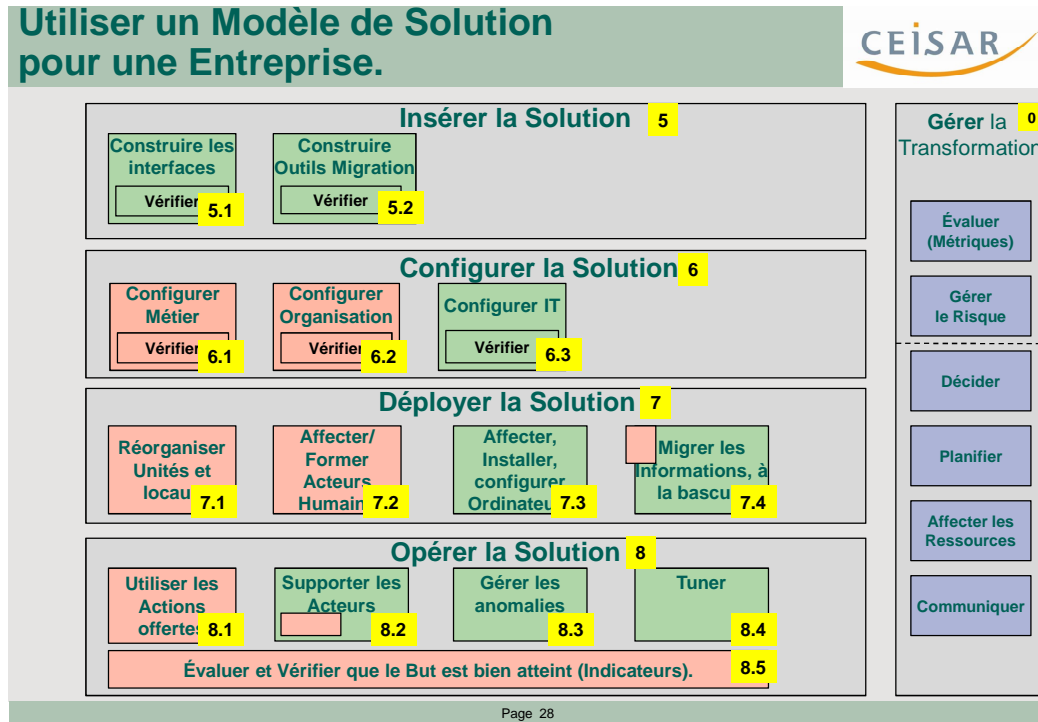
A titre d'exemple, la **définition d'un nouveau Processus** ne fait pas partie du Problème, elle fait partie de la Solution : dans le schéma ci-dessous elle est représenté par la partie rose de 3.3 ; la partie verte représente sa traduction IT.



1. Comprendre le **Contexte** :
 - 1.1. Comprendre les **Fondations** réutilisables pour économiser la charge du Projet et standardiser l'utilisation de la Solution. Les Fondations incluent le Modèle Global (ce que l'on appelle souvent « Cartographie des Solutions ») et les Modèles détaillés (Composants réutilisables, Modèles d'Information réutilisés...).
 - 1.2. Comprendre la **Solution à remplacer** : il n'est pas nécessaire de comprendre son Architecture interne si elle doit être remplacée, mais il faut bien comprendre les Fonctionnalités offertes, pour que la nouvelle Solution les reprenne, et comment elle communique avec les autres Solutions
 - 1.3. Comprendre l'**Approche** et les **Outils de Transformation** préconisés par l'Entreprise
2. Décrire le **Problème** à résoudre :
 - 2.1. Définir le **périmètre** (géographique, Produit, domaine de Processus, segment de clientèle...),
 - 2.2. Définir le **But** à atteindre en terme métier : gagner 10% de productivité, lancer une nouvelle gamme de Produits, diviser par 2 le « time to market » organiser un nouveau partenariat avec un distributeur, lancer une filiale à l'étranger
 - 2.3. Définir les **indicateurs** de Valeur pour vérifier le résultat obtenu à l'issue du projet
 - 2.4. Définir les **Contraintes** sur les Fonctionnalités : cadre réglementaire, volumétrie, performances, confort d'utilisation nécessaire pour inciter à l'utilisation, etc.
 - 2.5. Définir les **Contraintes** sur la Structure du Modèle et ses Interfaces avec les autres Solutions
 - 2.6. Définir les **Contraintes** sur le Projet : délai, budget maximum d'investissement, politique de ressources
3. Créer ou Modifier le **Modèle de Solution** : inclut à la fois **Modèle Métier** et sa traduction en **Modèle IT** :
 - 3.1. Construire le **Modèle Global** : définir les interactions entre Modèles d'Acteurs, Modèles d'Actions et Modèles d'Informations
 - 3.2. Construire le **Modèle d'Acteurs** : Organisation, Rôles, affectation des Activités aux Rôles
 - 3.3. Construire le **Modèle de Processus** : définir les Processus Cœur-Métier, les décomposer en Fonctions Cœur-Métier, puis définir les Processus Organisés et les Activités
 - 3.4. Construire le **Modèle de Fonctions**
 - 3.5. Construire le **Modèle d'Informations** : définitions Métier et IT des Entités, des Relations, des Héritages, des Attributs, des Types
4. **Vérifier** que le Modèle de Solution résout le Problème (et que le Problème n'a pas changé en cours de route):
 - 4.1. Vérifier les Fonctionnalités offertes (c'est le « Modèle d'Action Métier »)
 - 4.2. Vérifier que les Contraintes sur les Fonctionnalités (2.4) sont bien respectées

- 4.3. Vérifier que les Contraintes sur la Structure du Modèle (2.5) sont bien respectées, et récupérer des Composants réutilisables pour d'autres Modèles
- 4.4. Vérifier que les Contraintes sur le Projet (2.6) ont été respectées

Ces 4 Fonctions permettent de Construire un Modèle de Solution. C'est le travail à effectuer pour une Solution spécifique à une Entreprise, ou par un fabricant de Progiciel, ou par un Groupe qui Construit une Solution destinée à plusieurs Filiales.



Il reste à **installer** ce Modèle de Solution dans une ou plusieurs Entreprises.

Les 4 Fonctions suivantes (Adapter la Solution, Configurer la Solution, Déployer la Solution, Opérer la Solution) peuvent s'appliquer autant de fois qu'il y a d'Entreprises qui **réutilisent** le même Modèle.

- Insérer** la Solution dans l'Architecture d'Entreprise : Construire les **Interfaces** avec les autres Solutions et les Outils d'aide à la **Migration** des Informations provenant de l'ancienne Solution.
- Configurer** la Solution grâce à du paramétrage ou à un moteur de Règles pour qu'elle respecte les caractéristiques de chaque Entreprise. C'est une modification du Modèle, mais elle est parfaitement localisée et ne nécessite pas de Fonctions complexes d'intégration, de tests de non régression, ...Configurer un nouveau Modèle devrait être plus simple que de le Construire, la facilité de Paramétrage est d'ailleurs un critère d'évaluation de la puissance du Modèle.
- Déployer** la Solution configurée :
 - Préparer l'implémentation de la nouvelle Solution par une **Conduite du changement**
 - Réorganiser l'Entreprise, si nécessaire,
 - Affecter et Former les Acteurs qui vont utiliser la nouvelle Solution
 - Affecter, Installer et configurer les Ordinateurs (Serveurs et Stations), si nécessaire
 - Migrer les Informations à l'aide des Outils préparés si nécessaire
- Opérer** la Solution : on n'est plus dans la Transformation
 - Les Acteurs peuvent utiliser les « Actions » offertes (Processus et Fonctions)
 - Une équipe de Support les aide en cas de difficulté
 - On gère les anomalies
 - On optimise la Solution
 - Et lorsque la Solution est entièrement généralisée, on évalue les indicateurs de Valeur définis au départ pour vérifier que le But est bien atteint.

4.3 Quelques améliorations d'Ingénierie pour accroître l'agilité

Nous développons dans la suite de ce document quelques propositions d'améliorations de l'Ingénierie qui peuvent faire progresser fortement l'Agilité :

- Rechercher un **Modèle unique** partagé par Métier et IT : il évitera toutes les erreurs de traduction du Modèle Métier vers le Modèle IT
- Définir un **langage Métier** rigoureux pour faciliter la communication entre Métier et IT
- Réutiliser des **Composants** sous toutes les formes pour diminuer la charge de travail
- Modéliser le Cœur-Métier **avant** l'Organisation
- Appliquer chaque fois que possible l'Approche **Coopérative** plutôt que l'Approche Contractuelle

Pour appliquer ces recommandations, il faut faire évoluer les **Rôles** dans la Transformation et adapter les **Organisations**, ce qui suppose de **Former** les acteurs concernés.

5 Un seul modèle partagé par Métier et IT

Prenons un exemple : si une Entreprise souhaite faire des gains de Productivité de 10%, elle doit transformer un certain nombre de Processus Opérationnels. Elle doit donc procéder à 3 actions :

1. Définir le **Problème** « Faire Croître la Productivité de 10% dans tel périmètre »
2. Construire la **Solution Métier** qui consiste à redéfinir certains Processus Opérationnels.
3. Traduire la Solution Métier en **Solution IT** si l'informatique est utilisée dans la Construction des Processus modifiés.

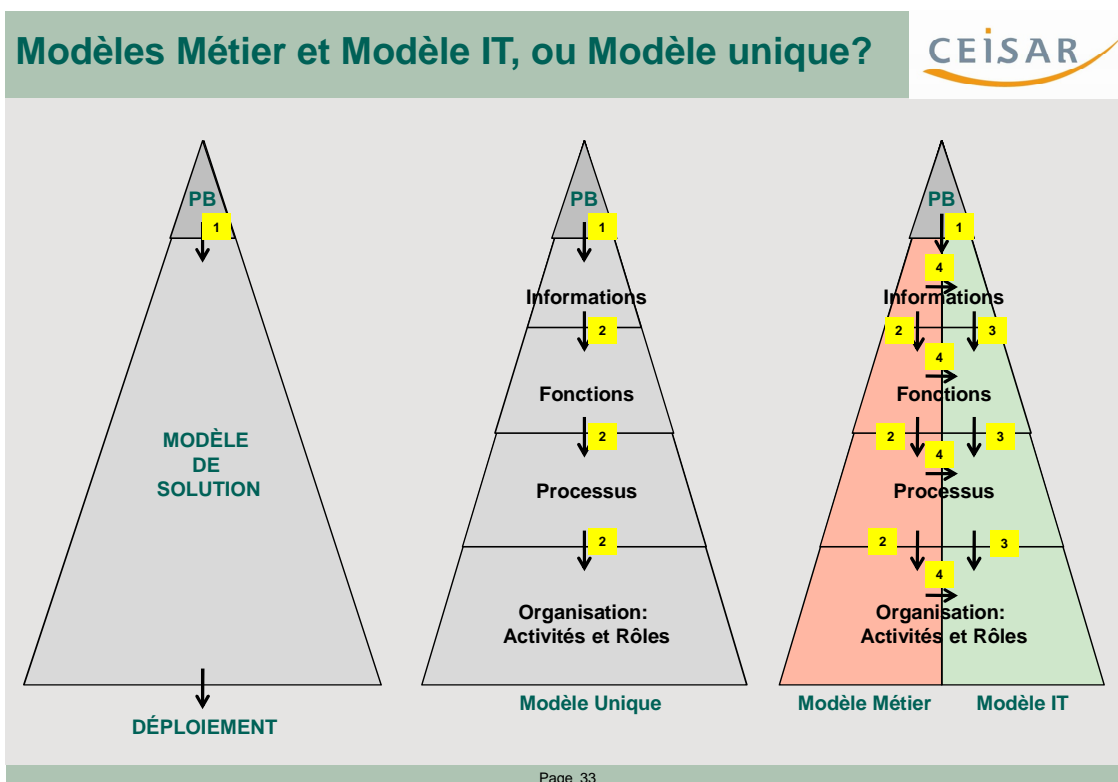
La Fonction 2 « Construire la Solution Métier » peut être effectuée à la main (on produit du Word ou du Powerpoint) ou à l'aide d'outils de Modélisation qui guident le Constructeur Métier et produisent un Modèle Métier formalisé qui utilise un **Méta Modèle Métier** (un Méta Modèle définit les concepts inter-reliés qui sont utilisés pour structurer les Modèles»).

La Fonction 3 « traduire la Solution Métier en Solution IT » est effectuée en concevant et programmant un Modèle IT (le « logiciel ») qui utilise un **Méta Modèle IT**.

Si la Fonction 2 produit un Modèle Métier formalisé et si on peut en déduire automatiquement une partie du Modèle IT, cela signifie qu'une partie du Méta Modèle Métier et du Méta Modèle IT sont **confondus**. On utilise alors un **Méta Modèle unique** plutôt que 2 Méta Modèles séparés pour la partie Métier et la partie IT, et on offre des **Vues** de ce Modèle adaptées à chaque Acteur. On n'a plus besoin de traduire le Modèle Métier en Modèle IT, on n'a plus besoin de vérifier par un plan de tests que les 2 Modèles sont cohérents. Autrement dit l'Acteur Métier définit le Modèle Unique et ce dernier produit automatiquement une partie du Modèle IT exécutable (le Logiciel). C'est un moyen de **Fluidifier la relation Métier IT** et **d'accélérer** la Construction.

Il n'existe pas encore d'Outil universel qui transforme tout un Modèle Métier en Modèle IT, mais des progrès sont faits sur des parties du Modèle :

- Modéliser les Informations et générer automatiquement le Modèle de Données IT
- Modéliser directement des Règles métier dans un moteur de règles
- Modéliser un Processus et générer automatiquement la navigation et l'affectation
- Modéliser la présentation et générer automatiquement le Modèle GUI
- Modéliser des livrables de Business Intelligence et générer automatiquement la production des Informations



Chaque fois que l'on utilise un Méta Modèle unique pour Métier et IT on doit uniquement vérifier que :

1. La Solution répond bien au Problème
2. Le Modèle unique est bien Construit

Tant que l'on utilise des Méta Modèles différents pour Métier et IT on doit vérifier que:

1. La Solution répond bien au Problème
2. Le Modèle Métier est bien Construit
3. Le Modèle IT est bien Construit
4. Le Modèle Métier est bien traduit en Modèle IT

6 Définir et partager le langage Métier

Les termes que l'Entreprise utilise tous les jours tels que Client, Produit, Contrat, Service, Partenaire... sont rarement définis avec précision : mais, pourquoi perdre du temps à définir des Termes que tout le monde connaît ?

Dans la pratique, chaque terme regroupe le plus souvent plusieurs Entités.

Par exemple « Client » signifie :

- celui à qui on vend (pour le commercial),
- celui qui souscrit (pour le juriste),
- celui qui paye (pour le comptable),
- celui qui bénéficie du Produit (pour le service livraison),

On peut faire le même exercice avec « Offre », « Produit », « Service », « Contrat », « Ressource », « Vol » chez Air France ou « Train » à la SNCF...

Lorsque le Métier définit son Problème et son Modèle de Solution il doit utiliser des termes Métier précis qui vont l'aider à préciser son Modèle et à **communiquer plus facilement** avec l'IT.

C'est aussi un moyen de **bien structurer le Modèle** de Solution qui s'appuie avant tout sur le Modèle des Informations et de faire émerger des **Composants Réutilisables**.

Dans la pratique (voir Livre Blanc sur « Entités du Métier »), il suffit de définir une centaine de termes avec rigueur pour améliorer considérablement le dialogue.

Cette définition utilise les verbes « Être » pour les héritages et « Avoir » pour les relations.

C'est le Rôle d'un « Architecte-Métier Fondation » (voir plus loin) de créer, entretenir et communiquer ce Glossaire.

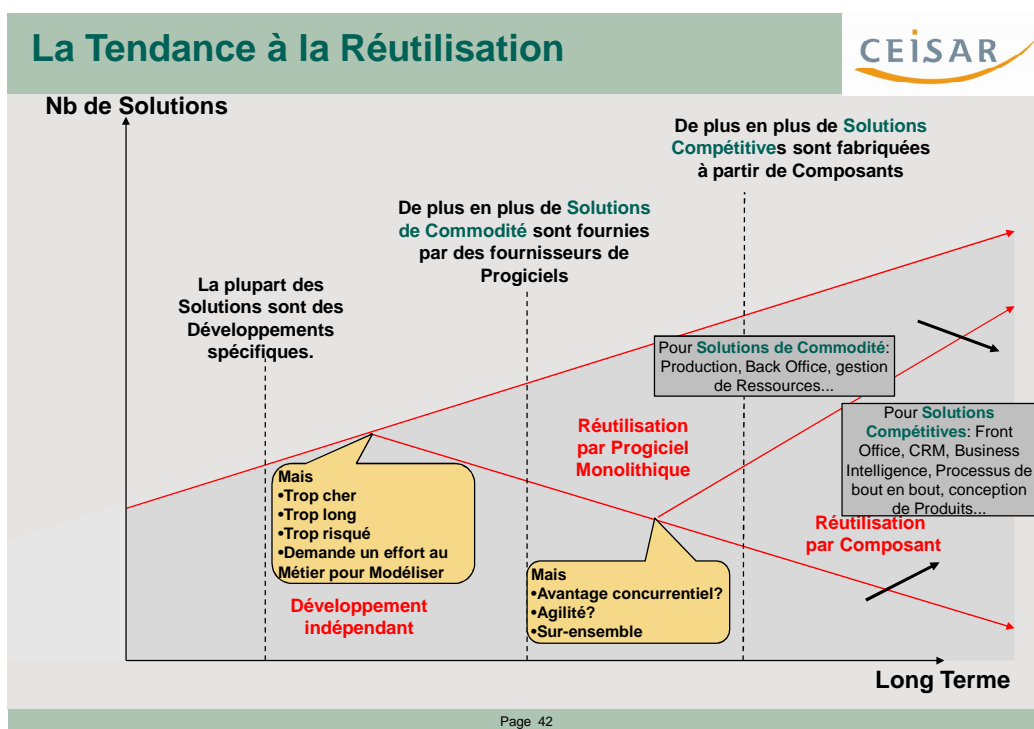
7 Réutilisation de composants

Il existe 2 formes de réutilisation : réutilisation par Progiciel ou réutilisation par Composants.

La réutilisation par Progiciel a eu un succès croissant ces dernières années pour les **Solutions de Commodité** pour lesquelles les besoins sont similaires entre Entreprises.

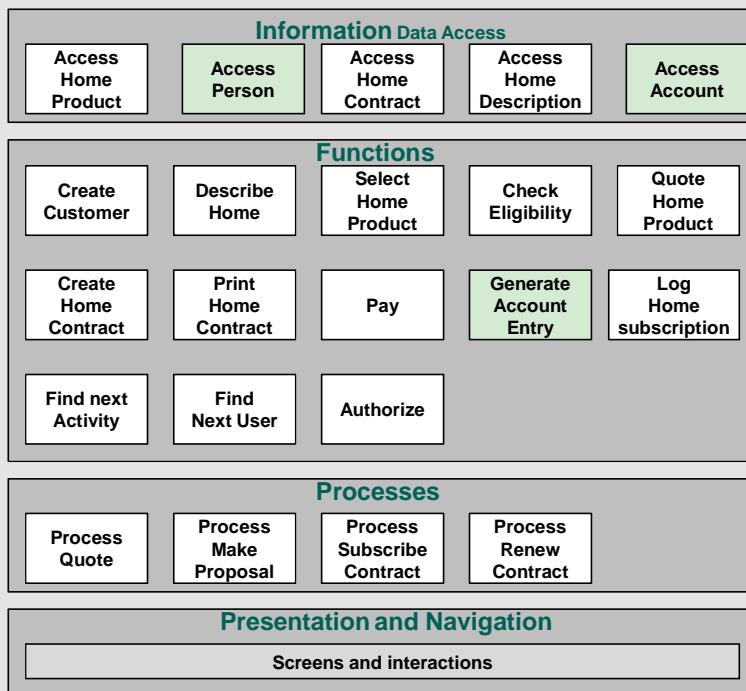
Mais une activité humaine n'atteint la maturité que lorsqu'elle est capable de réutiliser des composants communs pour construire des **Solutions Compétitives** : il a fallu 200 ans à l'Industrie pour aboutir à sa maturité actuelle. Le délai de conception d'un nouveau Modèle automobile a été réduit de 5 ans à 18 mois par la réutilisation de Composants.

Il faudra un certain délai pour que l'industrie du Logiciel en fasse de même. Mais on peut avoir de grands espoirs. Les tendances « **SOA** », « composant réutilisable », « approche Objet » vont toutes dans ce sens, et les résultats obtenus dans un certain nombre de Modèles de Solution prouvent que des taux de 70% de réutilisation sont réalistes, c'est-à-dire que l'on **n'a plus que 30% du Modèle à Construire** pour satisfaire un besoin spécifique. Les fabricants de Progiciels effectuent eux-mêmes cette mutation : les nouvelles offres de Progiciels sont souvent Construites à base de Composants. Cette tendance permet aussi d'accompagner plus aisément les développements de l'Entreprise étendue : un Modèle bien construit prend en compte les relations avec les Partenaires, les Fournisseurs ou les Clients qui deviennent des Acteurs-Organisation de l'Architecture d'Entreprise.



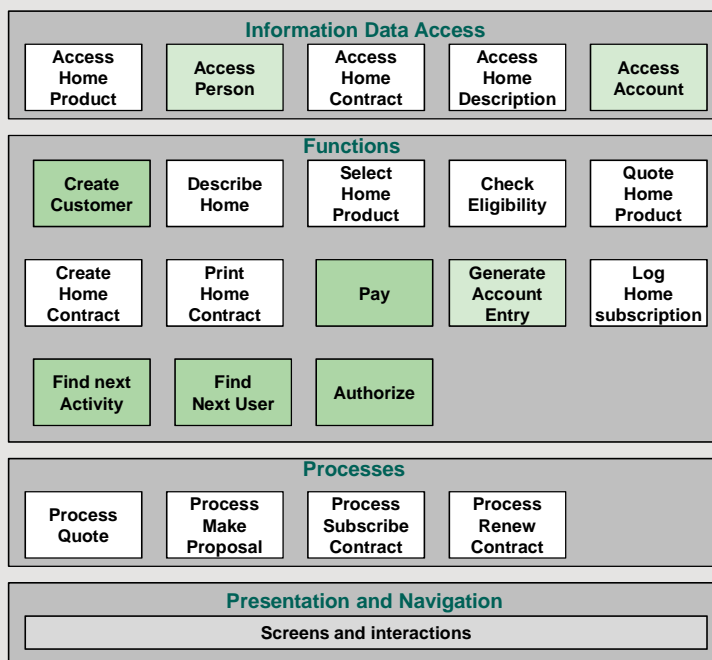
La réutilisation de Composants peut prendre une grande ampleur. Pour illustrer les potentialités nous avons pris un exemple de Solution Assurance décomposée en « 4 quarts » : accès aux Informations, Fonctions (Cœur-Métier ou Organisation), Processus, Présentation et Navigation.

On peut tirer parti de 4 niveaux de Composants Réutilisables que l'on regroupe sous le nom de **Fondations**:

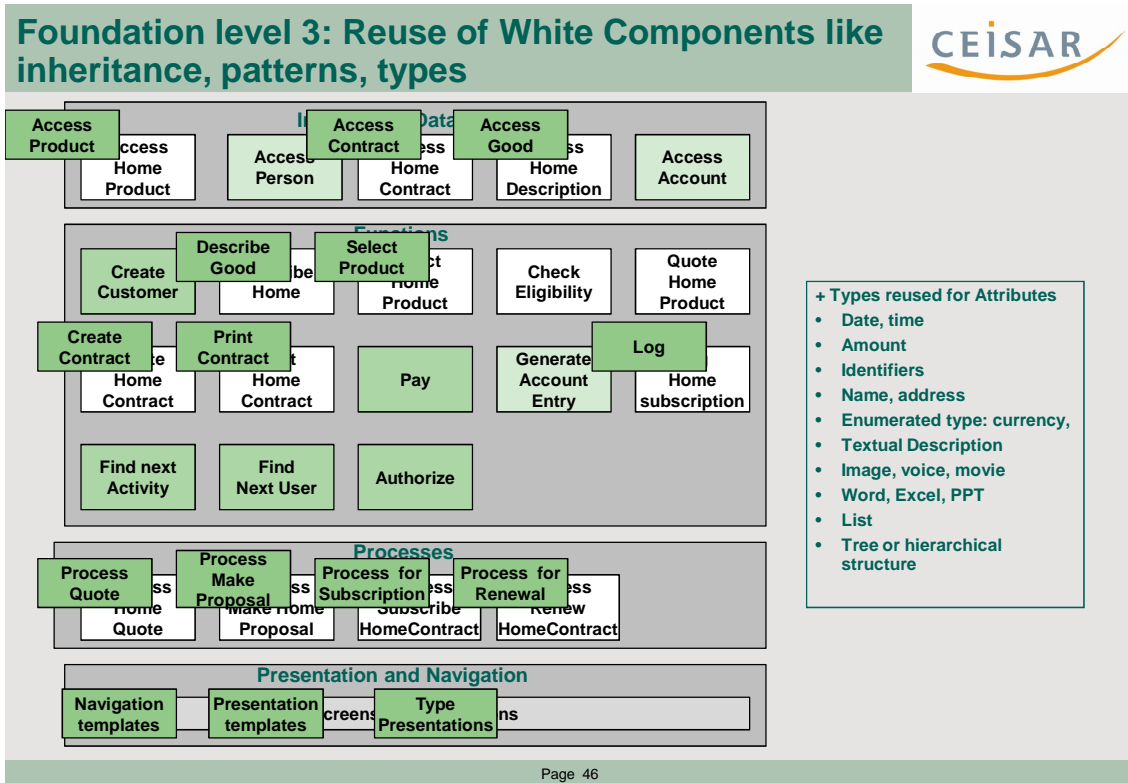


1. Isoler les **accès aux Informations** par des Fonctions partagées d'accès aux Informations permet de rendre indépendants les évolutions du Modèle d'Information et les évolutions du Modèle d'Action. Il permet aussi à des Solutions d'accéder à des Informations gérées par d'autres Solutions. De même isoler les **échanges de Flux** entre Solutions internes ou externes participe à la cohérence globale et minimise le couplage entre Solutions. C'est un moyen **d'isoler chaque Solution** pour lui donner de l'autonomie d'évolution.

Foundation level 2: Reuse of Black Components (Core-Business and Organization Functions)

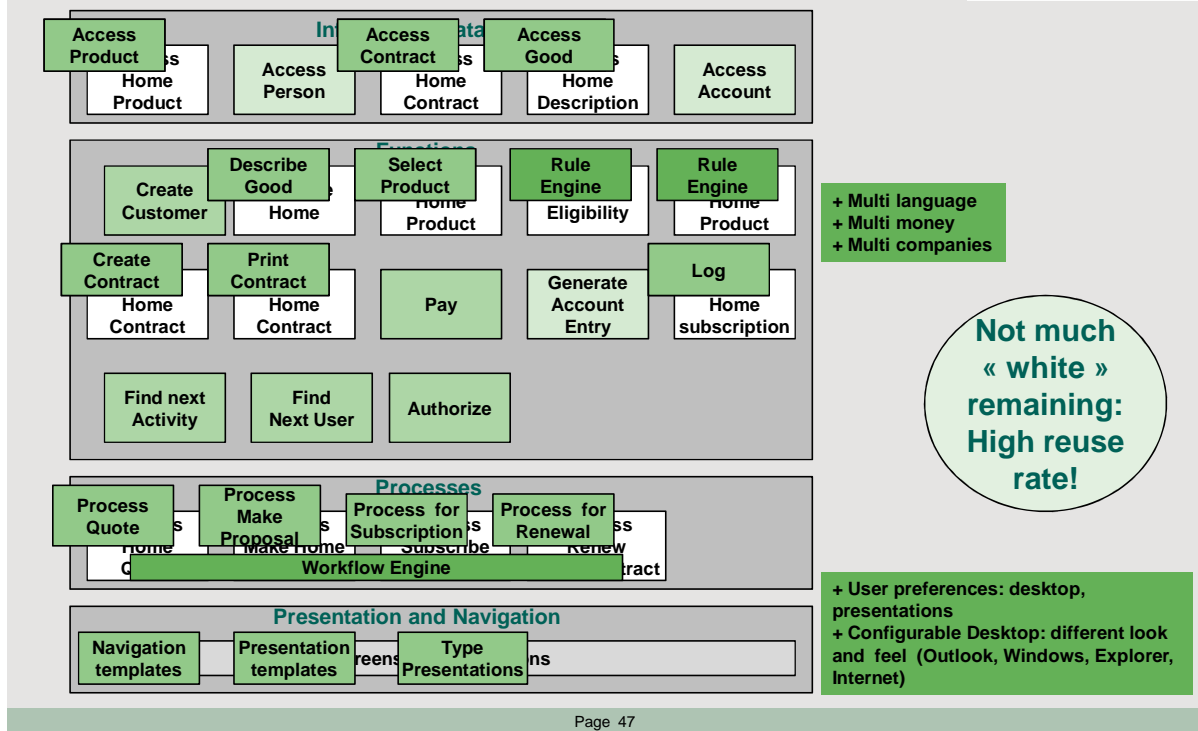


2. Réutiliser des **Fonctions** déjà disponibles pour diminuer ce qu'il y a à Construire : non seulement les Services d'accès aux Informations et les Services de Flux que l'on vient de décrire, mais aussi les Services Métier (comme « Tarifier »), les Services Organisation (comme « Habilitier »). C'est un moyen d'isoler encore davantage les Solutions et d'accélérer la Construction. Les Niveaux 1 et 2 sont appelés « **Composants Noirs** » parce qu'ils se comportent comme des boîtes noires dont on ne connaît que l'interface et non l'implémentation.



3. Réutiliser les **Patterns** ou « **Composants Blancs** » comme un « Modèle de souscription » qui est réutilisé par tous les Processus de souscription. C'est une forme de Réutilisation différente des 2 précédentes puisqu'elle intervient pendant la phase de Construction et non par des appels en cours d'Opérations. Elle peut concerner des Modèles de Processus, des Modèles de Fonctions, des Modèles d'Informations, des Interfaces utilisateur... Elle accélère considérablement la phase de Construction mais nécessite des **outils** adaptés.

Foundation level 4: configurable Component with Parameters and Rule engine



4. Externaliser à l'aide de **paramétrage** et de **moteur de règles** les parties du Modèle qui changent le plus fréquemment pour **Configurer le Modèle** sans passer par le Processus de Transformation complet qui nécessite Intégration, Recette, et mise en Production lourdes. On peut ainsi directement créer de nouveaux Produits, changer les tarifs, adapter.

En appliquant ces différents niveaux de réutilisation, un Progiciel d'Assurance international a pu mettre en commun des Fondations de 6.000 Classes offrant 50.000 Services : le coût complémentaire pour Construire une Solution adaptée à chaque Ligne Métier (IARD, Vie, Prévoyance, Santé...) est d'environ 5% des Fondations. Ce niveau a été atteint en bénéficiant de conditions favorables : pas d'existant qui bride la capacité d'innovation, outillages modernes, équipes expérimentées... Mais cette expérience prouve que l'on peut être extrêmement ambitieux en matière de réutilisation.

Réussir à réutiliser des Composants en masse nécessite des conditions du succès :

- **Isoler** dans une équipe « Fondation » ceux qui construisent, supportent, récupèrent les Composants.
- S'assurer qu'ils ont acquis le **savoir-faire** pour construire des composants : qualité des interfaces, structure de petits composants qui se réutilisent les uns les autres et non liste plate de gros composants, versioning, usage approfondi des Patterns, compatibilité ascendante, gestion de configuration adaptée ...
- Encourager et contrôler la **réutilisation** dans les équipes Solution.

Commentaire SOA

L'approche SOA traite aussi de Composants. Elle est caractérisée par :

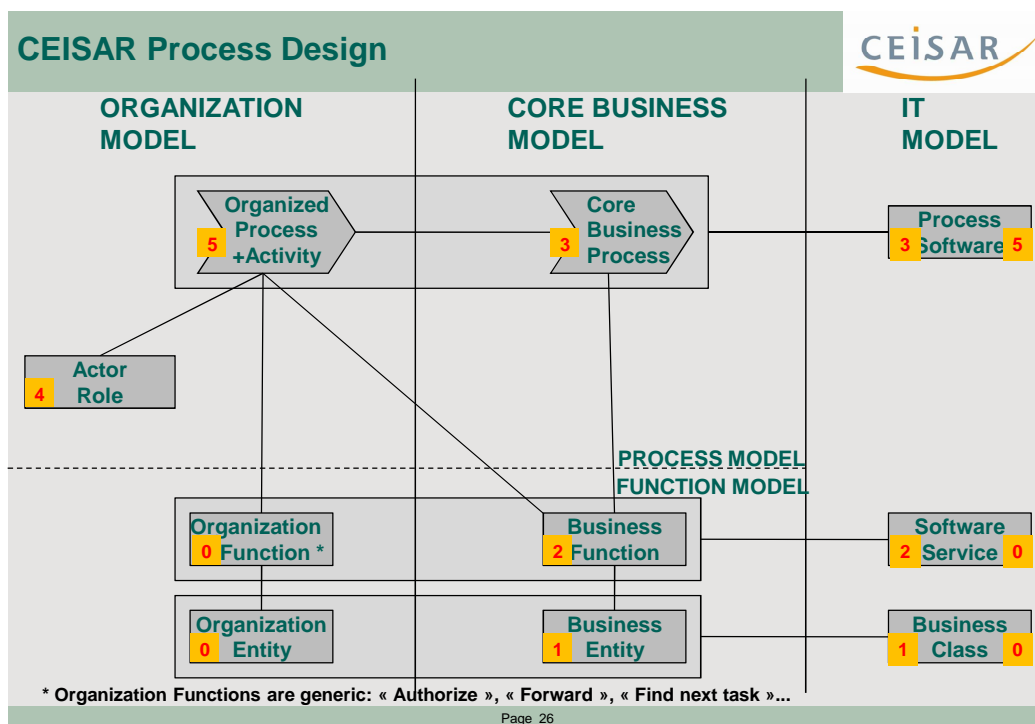
1. L'utilisation de **Composants Noirs** (niveau 1 et 2) et non de Composants Blancs
2. Dont l'**exécution** est interne ou **externe** ; dans ce dernier cas, se posent des problèmes de sécurité, de qualité de service, de fraîcheur d'informations... bref de confiance.
3. qui ont besoin de passer par un **middleware partagé** entre consommateur et fournisseur de Service, le plus simple étant que chacun s'appuie sur les standards du marché (aujourd'hui « Web Services »)

8 Processus-Métier et Processus-Organisé

Une Solution Logiciel dure 20 ans, les Organisations sont en perpétuel mouvement : comment analyser les Processus pour que le Logiciel qui s'en déduit supporte aisément différentes Organisations successives ou simultanées ?

Si vous demandez à un Acteur Opérationnel de décrire la façon dont il travaille ou la façon dont il aimerait travailler, il décrit naturellement ce qu'il vit quotidiennement : des **Acteurs** qui exécutent des **Activités**. Il vous parlera rarement de Processus et d'Informations.

A partir de ce qu'exprime l'Acteur opérationnel, la démarche du Constructeur de Solution doit être la suivante :



Définir le Cœur-Métier

1. Définir les **Entités du Cœur-Métier** telles que « Client », « Produit », « Commande », « Livraison » : ce sont les éléments les plus stables de l'Architecture d'Entreprise.
2. Définir le cycle de vie de chacune de ces Entités pour faire apparaître les **Fonctions Cœur-Métier** : « créer », « contrôler », « modifier », « interroger », « tarifer », « débiter »...
3. Définir les **Processus Cœur-Métier** et les décomposer en réutilisant les Fonctions Cœur-Métier définies dans l'Étape 2 et en rajoutant de nouvelles que l'on ne peut découvrir qu'en analysant les besoins du Processus.

Définir l'Organisation

4. Définir quels **Rôles** sont attribués aux Acteurs ou aux Unités organisationnelles qui les regroupent.
5. Définir les **Processus Organisés** découpés en **Activités** (une Activité n'est exécutée que par un Acteur) pour les affecter aux Rôles définis en 4. Décomposer chaque Processus Organisé en Fonctions Métier et Fonctions Organisation (ces dernières font partie des Fondations)

L'intérêt de cette démarche de Construction est multiple :

- Les Solutions sont basées sur le Cœur-Métier et supportent des **changements d'organisation**
- On fait apparaître des Entités et des Fonctions **réutilisables** parce qu'on analyse les Entités Métier avant les Processus.
- On peut commencer à Construire la Solution **sans attendre** que toute l'Organisation soit définie.
- La Solution peut aisément s'adapter à des partenariats externes.

9 Approche Contractuelle ou Approche Coopérative

Dans l'**Approche Contractuelle**, couramment pratiquée, on effectue toute la Modélisation Métier de la Solution avant de travailler sur sa Modélisation IT. C'est une manière efficace de s'assurer que la Solution IT répond bien aux besoins du Métier, ou que les « Fonctionnalités » sont Modélisées avant de développer le logiciel.

C'est particulièrement le cas lorsque la partie IT est **sous-traitée** à une équipe externe. La **Vérification** est alors une Fonction qui ne peut avoir lieu que lorsque la partie Informatique est terminée. Cette approche conduit à bien séparer les équipes Métier et Informatique : le lien doit être fait sous forme d'un document contractuel appelé « cahier des charges », ou « spécifications détaillées ».

Dans l'**Approche Coopérative**, on ne détaille pas toute la Vision Métier du Modèle avant d'entamer la Modélisation Informatique. La définition des Fonctionnalités est progressivement affinée au fur et à mesure des **itérations**. Cette approche est préférable lorsque les spécifications sont incertaines et que l'on souhaite construire des Solutions Évolutives plutôt que des Solutions définitives, c'est-à-dire des Solutions Compétitives et non des Solutions de Commodité. Cette approche est plus rapide, elle gomme l'effet tunnel, elle ne contraint pas le Métier à définir tout son Modèle avant de passer la main à l'Informatique, elle fluidifie la relation Métier-IT, elle permet une vérification progressive, elle permet d'associer Métier et IT dans une équipe mixte seule responsable de la Solution, mais elle ne peut être choisie **que si le Modèle global de la Solution supporte les adjonctions ultérieures**.

L'Approche Contractuelle est aujourd'hui plus diffusée que l'Approche Coopérative. Elle résout des Problèmes **bien cernés** tels que automatiser les Back Offices, la Production, la Comptabilité, ... bref pour Construire des **Solutions de Commodité**.

Par contre, elle est mal adaptée aux nouveaux Problèmes **extensibles** tels que l'automatisation du Marketing, du suivi Client, des Solutions Internet offertes aux Clients, de la Gestion des campagnes commerciales ou de la Business Intelligence : la Modélisation Métier de ces domaines est progressive, chaque nouvelle version de la Solution aide à préciser les contours de la version suivante. Les Entreprises vont avoir un besoin croissant d'Approche Coopérative pour les **Solutions Compétitives**.

Mais comment Construire un Modèle Global de Solution qui supporte des ajouts progressifs ?

On déploie des **Versions** de Solution et pour Construire chaque Version de Solution on procède par **itérations**.

Le Modèle doit accueillir les Versions et Itérations successives de la Solution sans déstabiliser sa structure (http://alistair.cockburn.us/index.php/Incremental_versus_iterative_development).

Pour réussir, il faut respecter ce qui a été défini précédemment:

- dissocier le Cœur-Métier et l'**Organisation**,
- suivre la **séquence** « Entités, Fonctions, Processus »,
- réutiliser un maximum de **Composants**.

On peut rajouter :

- prévoir que tous les éléments du Modèle vont évoluer (tous les éléments du Modèle doivent intégrer une gestion de **Version**)
- demander aux **meilleurs** « Modeleurs » de se consacrer à la Construction des Modèles de Solutions ou Fondations ; ne pas les surcharger par des tâches de pure gestion, ils représentent une ressource rare
- faire **certifier** la qualité du Modèle Global par des experts.

Pour ceux qui souhaitent aller plus loin, voir les Livres Blancs du CEISAR sur Modélisation des Entités du Métier et Modélisation des Processus.

Modèle Unique

L'approche Coopérative est plus efficace si elle s'appuie sur un **Modèle unique** de la Solution : toute modification qu'elle provienne d'un changement Métier ou d'un changement IT met à jour le Modèle unique et modifie instantanément les vues offertes à chacun. On parle d'outils « Round Trip » (voir ci-dessus).

Comment arrêter une Version : limiter les spécifications par la date et non le périmètre fonctionnel.

Comme on ne définit pas tout ce qu'il faut faire avant de faire, il faut définir une règle pour arrêter une Version.

Le perfectionnisme est dangereux : de nombreux projets ont des difficultés parce que les MOA craignent d'oublier des exigences et ne savent quand arrêter d'en ajouter pour livrer une première Version. Ils se rappellent l'approche Contractuelle : « si vous oubliez des exigences, il sera coûteux et long de les prendre en compte dans les versions suivantes ».

Il faut les aider à appliquer la règle suivante : « comme il est aisé de passer à une nouvelle version incrémentale, livrons rapidement une première version pour offrir un premier niveau de service aux Acteurs ». Naturellement, s'il s'agit du remplacement d'une application ancienne, il faut au moins reprendre les Fonctions qui étaient offertes par l'ancienne application.

La bonne règle est de fixer une **date limite** réaliste pour la première version, ce qui a pour conséquence que les MOA doivent désigner un leader capable de **faire le tri** entre les exigences progressives pour tenir la bonne date. Ce principe de sobriété est la clé d'une approche coopérative réussie. La première version doit livrer :

- L'Architecture de la nouvelle Solution qui doit permettre des incréments rapides
- Les Fonctionnalités qui étaient déjà offertes par l'ancienne Solution
- Quelques Fonctionnalités supplémentaires séduisantes pourvu qu'elles soient compatibles avec l'échéance fixée.

Équipe Unique de Construction pour l'Approche Coopérative

Les Rôles Métier et IT sont réunis au sein de la même équipe qui est seule responsabilité de la Solution. Chaque Version de la Solution est un compromis entre souhaits fonctionnels et possibilités techniques. La sous-traitance des développements informatiques est possible. Mais elle procède par petits contrats successifs : un Contrat de sous-traitance peut être défini pour chaque Version après qu'un nombre suffisant d'itérations aient été effectuées pour sélectionner les exigences de la Version courante.

Peut-on fonctionner avec des équipes dispersées ?

Une équipe Projet fonctionne toujours mieux lorsque ses représentants sont installés dans les mêmes locaux. Mais ce n'est pas toujours faisable. On peut néanmoins appliquer une approche coopérative entre équipes dispersées s'ils partagent des outils de communications efficaces non seulement entre individus mais aussi entre Modèles. L'idéal est d'offrir à chaque équipe locale un référentiel (repository) local, et d'agréger au niveau Central les contributions de chaque équipe locale dans un référentiel global qui gère le versioning, la consistance et qui offre à chacun les vues qu'il souhaite.

Faut-il conserver des Transformateurs IT (concepteurs, développeurs informatique) en Entreprise ?

Le Modèle Métier doit être traduit en Modèle IT. Cette traduction est de plus en plus considérée comme une tâche externalisable sans véritable valeur ajoutée : les **Transformateurs IT** sont souvent **peu considérés** dans l'Entreprise. Cette tendance présente 2 dangers :

- D'une part, si le Modèle Métier est bien construit au départ, il est rarement mis à jour par la suite alors que le logiciel subit des modifications successives. Finalement seul le Modèle IT représente la réalité à jour du Modèle d'Entreprise : lorsqu'elle a externalisé son Modèle IT, il devient plus difficile pour l'Entreprise de savoir avec précision comment elle fonctionne.
- D'autre part, l'externalisation va être un frein à la mise en place d'une Approche Coopérative dans les Entreprises : difficile de jouer le jeu du compromis entre Souhaits fonctionnels et Possibilités IT si on externalise et contractualise la relation.

Il est souhaitable que toute Entreprise conserve un noyau de compétences IT reconnues, qui ont la main sur la conception globale du Modèle IT, faute de quoi elle se retrouvera un jour dépendant de fournisseurs qui auront fait évoluer les Solutions sans que l'Entreprise ne conserve la connaissance de son Modèle Métier.

On peut associer Approche Contractuelle et Approche Coopérative

Conserver de l'Approche Coopérative

- Versions successives

- Qualité du Modèle Global

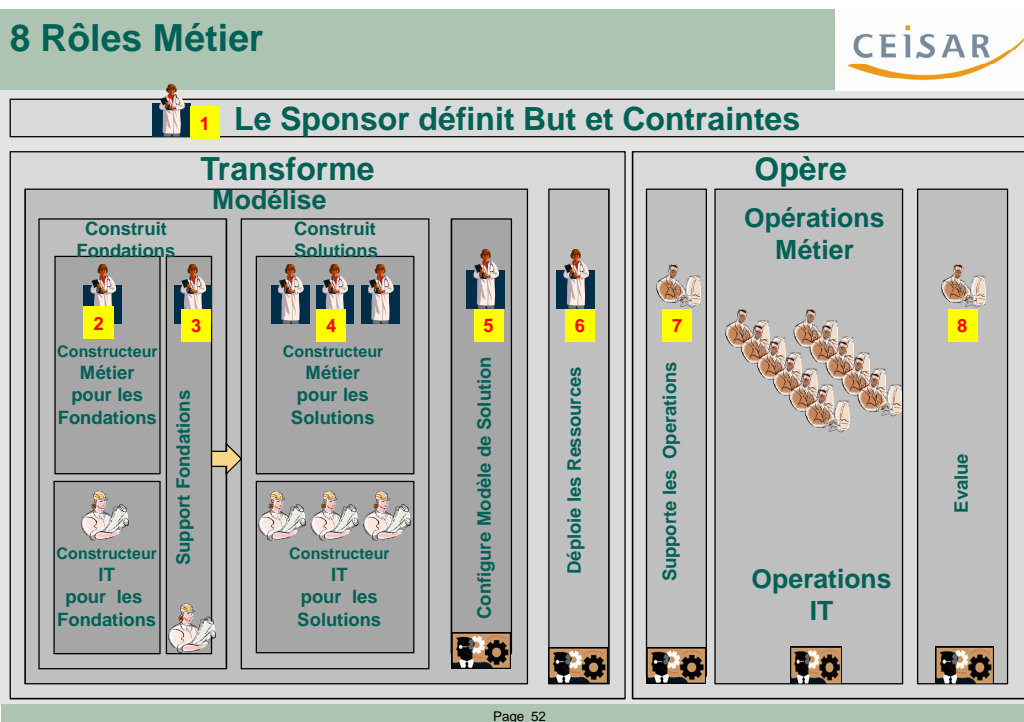
Conserver de l'Approche Contractuelle

- Séparation des Équipes Métier et IT
- Possibilité de contracter la Construction du Modèle IT

Il faut alors contracter sur des périmètres plus restreints que l'on définit pour chaque Version.

Comme l'a évalué un des Sponsors du CEISAR, il faut avoir consommé 40% de la charge de Modélisation pour obtenir une évaluation précise à 10% : plus on souhaite être précis dans l'estimation, plus il faut s'avancer dans le projet.

10 Les Rôles Métier de la Transformation



Comme la question posée par les Sponsors du CEISAR, pour ce livre blanc, concernait aussi la formation des Maitrises d'Ouvrage, nous nous sommes **concentrés sur les Rôles Métier** et non les Rôles IT. Il est à noter que Métier et IT sont intimement liés dans le Processus de Transformation et qu'un prolongement IT de cette liste devrait être défini dans un prochain livre blanc.

Il existe de nombreux Rôles Métier dans la Transformation tels que :

- Administrateur de Données
- Analyste
- Architecte Métier Fondations
 - Architecte Composants
 - Architecte Sécurité
- Architecte Métier Solution
- Chef de Projet Métier
- Concepteur Métier
- Configureur de Solution
- Ergonome
- Gestionnaire du Changement
- Maître d'Ouvrage
- Méthodologue
- Organisateur
- Responsable Qualité
- Responsable Sécurité
- Responsable Solution Métier
- Sponsor
- Urbaniste Métier

Plutôt que de lister un sur-ensemble des Rôles Métier qui existent chez nos Sponsors, nous avons cherché à simplifier cette liste en définissant des **Rôles globaux attachés aux équipes**; nous en avons défini 8 :

- En plus du Rôle de **Sponsor**, il existe
- 5 Rôles dans la Transformation
 - **Constructeur Métier pour les Fondations**

- **Support Métier** pour les Fondations auprès des Constructeurs Métier Solutions
- **Constructeur Métier** pour les **Solutions**
- **Configurateur** de Solution
- **Déploieur**
- 2 Rôles dans les Opérations
 - **Support** de la Solution auprès des Acteurs Opérationnels
 - **Évaluateur** de Solution Opérationnelle (ou Pilote de Processus)

Dans chacune des 8 catégories peuvent exister

- différents niveaux **hiérarchiques** selon la taille de l'équipe : par exemple pour le Rôle de « Constructeur Métier » (souvent appelé « Maitrise d'Ouvrage ») on peut définir le « Chef de projet Métier » et « l'Analyste Métier »
- différents spécialistes peuvent se **spécialiser** en tant qu'**experts** d'une partie du Modèle : ergonomes, organisateurs, experts sécurité.

Nous proposons de donner une définition du Rôle de chaque Équipe. A chaque Entreprise de décider si elle souhaite ou non décliner plusieurs Rôles d'Acteur au sein de chacune de ces Équipe.

La seule recommandation que l'on peut faire est de ne pas multiplier les Rôles d'Acteur pour ne pas multiplier les tâches de gestion et d'interaction : si on a le choix, il vaut mieux **peu d'Acteurs de haut niveau** et donc polyvalents, plutôt que de **nombreux Acteurs spécialistes**.

- Le **Sponsor** (ou Donneur d'Ordre) définit le Problème : But, Valeur et Contraintes. C'est le sponsor qui décide, qui paye et qui vérifie que la solution apporte bien la valeur définie dans le Problème.
- **Les Constructeurs Métier des Fondations** définissent le **plan d'ensemble des Solutions** (on parle d'Urbanistes), ils conçoivent les Composants Métier **commun** à toutes les Solutions : définition des Entités du Métier, Modèle de Processus (tels que Modèle de Processus de Souscription qui peut servir de pattern à tous les Processus de souscription), Fonctions Métier Communes (telles que tarification), Fonctions d'Organisation communes (comme contrôle d'habilitation ou normes d'ergonomie). Ce sont aussi ceux qui **définissent l'Approche**.
- **Les Supports Métier Fondations**: ils forment, assistent, contrôlent les Constructeurs Métier Solutions pour qu'ils réutilisent bien les Fondations Métier et appliquent bien l'Approche de Transformation.
- **Les Constructeurs Métier des Solutions** modélisent la Solution (Modèle global Métier, Modèle Métier d'Informations, Modèle Métier de Fonctions, Modèle Métier de Processus, Modèle d'Organisation), testent le Modèle de Solution IT qui s'en déduit et définissent le Modèle d'Interfaces. Le responsable de l'Équipe Projet Solution s'appelle « **Chef de Projet Solution** ».
- Les **Configurateurs Métier** utilisent paramétrage, moteur de règles et moteur de workflow pour configurer le Modèle (règles de tarification, de contrôle, de commissionnement...)
- Les « **Déploieurs** » conduisent le Changement : ils réorganisent les Opérations, prévoient les installations d'ordinateurs, forment les Acteurs Opérationnels, et aident à la Migration des Informations
- Les **Supports Métier aux Opérations** œuvrent à la hot line pour aider les Acteurs Opérationnels chaque fois qu'ils ont une difficulté.
- Les **Évaluateurs de Solution** (ou Pilote de Processus) observent le fonctionnement Opérationnel et en tirent des demandes d'amélioration pour les versions suivantes.

Focus sur le Rôle de « Constructeur Métier Fondations »

Nous insistons sur ce Rôle parce qu'il est rarement identifié dans les Entreprises, alors que les responsabilités que nous venons de décrire sont de plus en plus importantes dans les Organisations qui recherchent de la cohérence, plus de synergies et des économies d'échelle.

La raison en est qu'il n'existe pas d'Unité transversale aux Métiers : la Direction Générale est éloignée de ces préoccupations et les Directions Métier qui se consacrent à l'efficacité de leur propre Métier n'ont pas en charge le bien commun.

C'est pourquoi ces responsabilités sont généralement effectuées par la DSI : cette dernière n'a pas les compétences Métier, mais elle comprend mieux les enjeux de synergie, de mise en commun, de réutilisation parce qu'elle rassemble la Modélisation IT des Solutions pour **tous** les Métiers. Il faut impérativement expliquer à la Direction Générale l'importance de ces préoccupations pour que des ressources Métier de qualité les prennent en charge au sein de la DSI ou ailleurs.

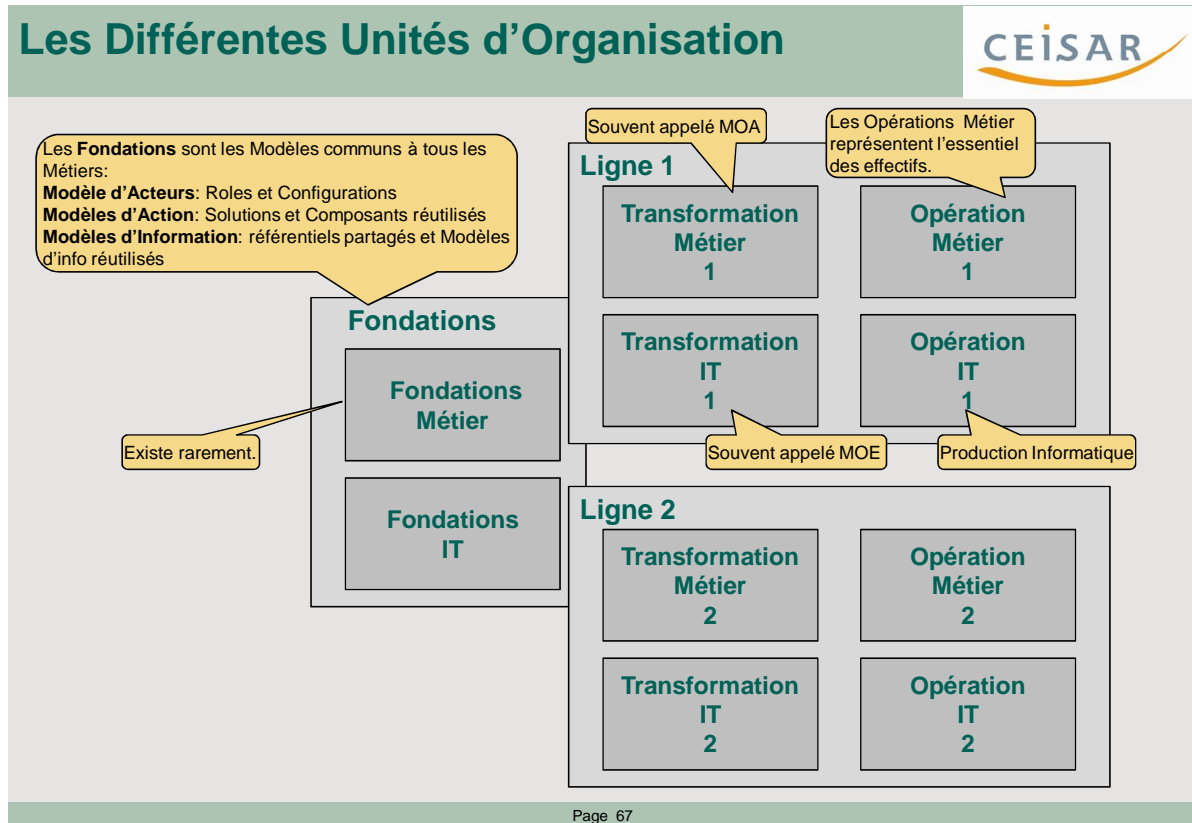
Son rôle essentiel est de définir la cartographie Métier des Solutions, d'identifier les Composants Métier **réutilisables** entre différents Modèles de Solutions et de contribuer à la définition des Processus de Transformation (ou « Approches » ou « Méthodologie ») que les Projet-Solution doivent respecter.

- Il définit la cartographie des Solutions dans ses aspects Métier (rôle de l'**Urbaniste**)
 - **Classification des Processus** et des **Fonctions**
 - Plan des Solutions et **flux** entre Solutions
 - Utilisation de **Référentiels d'Informations** partagés
- Il définit les **Entités** réutilisables dans différentes Solutions
 - **Entités Métier** telles que Produit, Acteurs, Contrat, Compte etc.
 - **Entités Organisation** telles qu'Unité, Rôle, Profil, Droit, Panier de tâches,....
 - **Pour chaque Entité, il précise** : Définition textuelle, Relations avec les autres entités, Patterns ou héritage d'Entités, identifiant, versioning, Attributs, Types
- Il définit les **Fonctions** réutilisables:
 - **Fonctions Cœur-Métier** telles que Tarifer, Contrôler l'éligibilité, Rechercher l'information sur un Client...
 - **Fonctions Organisation** telles que Habilitier, Affecter des Activités aux Acteurs, Gérer des tâches de chaque Acteurs, Journaliser des Actions, Gérer l'interface utilisateur
 - **Pour chaque Fonction il précise** : à quoi elle sert, quelles Informations lui communiquer pour qu'elle puisse s'exécuter,
Remarque : il est peu impliqué sur les Fonctions techniques réutilisables telles qu'interfaces à des périphériques (GED, impression, lecteur de cartes...) ou interfaces réseau
- Il définit les **Patterns de Processus métier** réutilisables tels que Modèle de Processus de souscription qui peut servir de pattern à tous les Processus de souscription
- Il définit les normes **d'interface utilisateur** : présentation, navigation, normes d'impression
- Il définit les **Processus de Transformation** que doivent respecter les Analystes Métier Solution.
- Il choisit les **outils** qui automatisent certaines Fonctions des Processus de Transformation.

11 Organisation

Une fois les Rôles définis, on peut choisir une Organisation.
Il faut distinguer plusieurs dimensions :

- Métier et IT
- Opération et Transformation
- Ligne Métier, Solution et Fondation



Si l'on souhaite favoriser l'agilité, il faut

- **Isoler les Fondations** : réunir des compétences Métier et IT au sein de la même équipe pour qu'ils produisent des Composants réutilisables et l'Approche adaptée. Il faut leur donner le pouvoir de faire respecter la réutilisation de composants.
- **Séparer Opérations et Transformations** parce que les préoccupations présentes sont toujours prioritaires vis-à-vis des préoccupations futures. Cette séparation peut avoir lieu au sein de chaque Ligne Métier ou globalement pour l'Entreprise.
- Chaque équipe de Transformation doit **rassembler les compétences Métier et IT** même s'ils adoptent une approche contractuelle, parce que les compromis sont alors résolus au niveau le plus proche des Projets : appliquer le principe de subsidiarité.
- L'Entreprise doit **maîtriser son Modèle Global** et ne sous-traiter que des Modèles détail : c'est la clé de son évolutivité.

Les « Contrats »

Selon la forme d'Organisation choisie il peut exister différentes formes de contrats entre Acteurs :

- Un contrat entre le Sponsor et le Chef de Projet global qui est la description du **Problème**
- Un contrat entre le Chef de Projet Métier et le Chef de Projet IT qui peut être le **modèle Métier Global ou Détail** selon que l'on fait appel ou non à des assistants à Maîtrise d'Ouvrage.
- Un contrat pour la sous-traitance IT qui peut être le **Modèle Métier** ou le **Modèle IT Global** selon que l'on sous-traite la conception globale IT ou non

Les qualités d'un Chef de Projet Solution : Modélisateur et Gestionnaire

- Un chef de projet qui n'est que **Gestionnaire** sans avoir de talents de **Modélisation** n'aura pas les réflexes de bon sens de celui qui sait ce qui est réaliste, ce qui s'implémente aisément, ce qui remet en cause le Modèle profondément. Pour compenser ce manque de jugement ou d'expérience, il aura tendance à se protéger en alourdissant les procédures de contrôle, de reporting, ou de réunions pour obtenir un consensus qui le protège.
- Un chef de projet qui n'est que **Modélisateur** sans avoir de talents de **Gestionnaire** sera confronté à d'autres problèmes bien identifiés : entre autres, il ne saura pas arrêter l'afflux d'exigences, il négligera les aspects reporting ou documentation.

S'il a des compétences Métier et IT, c'est un profil parfait, mais rare.

S'il ne les a pas, il faut qu'il s'entoure d'experts qui lui apportent cette compétence.

12 Formation des Transformateurs Métiers

Si l'on souhaite que les Constructeurs Métier s'approprient la Transformation et collaborent efficacement avec l'IT, il faut les former à la Transformation. Ils doivent avoir assimilé l'**Approche**, les **Outils** de Transformation, les **Fondations** et l'Architecture du **Modèle en place** qu'ils doivent Transformer.

Le préalable à toute formation est donc de définir l'Approche, les Outils et les Fondations.

S'il s'agit de l'Approche, des Outils et des Fondations **en place**, cette formation peut être assurée par les équipes internes qui les utilisent déjà.

Si l'Entreprise souhaite **améliorer** son Approche, ses Outils et ses Fondations, il faut alors faire intervenir des experts externes qui peuvent apporter des éléments nouveaux aux équipes en place : exemples concrets de Projets d'Entreprises qui ont réussi avec des Approches, Outils et Fondations différentes, conditions du succès.

Les Modules de Formation que doivent suivre les Transformateurs Métier sont les suivants :

- Comprendre ce qu'est l'**Architecture d'Entreprise** : alignement avec la Stratégie, décomposition entre Opérations et Transformations, décomposition entre Monde réel et Modèle, recherche de Synergies par réutilisation de Modèles et Partage de Ressources
- Comprendre ce qu'est un **Modèle d'Entreprise** : Modèle d'Informations, Modèle d'Actions (Processus et Fonctions) et Modèle d'Acteurs (Acteurs humains et ordinateurs)
- Savoir exécuter un Projet : la formation doit s'appuyer sur le Processus de Transformation en vigueur dans l'Entreprise. A l'occasion de la description de ce Processus il faut mettre en valeur la **décomposition en 8 Fonctions d'Ingénierie** (Comprendre le Contexte, Définir le Problème, Construire la Solution, Vérifier la Solution, Adapter la Solution, Configurer la Solution, Déployer la Solution, Opérer la Solution).
- Comprendre **Approche Coopérative** et **Approche Contractuelle**, et assimiler les Fonctions de Gestion associées :
 - le découpage en Phases
 - le Rôle des Acteurs de la Transformation et l'affectation des Acteurs aux Phases
 - les itérations autorisées
 - comment évaluer la charge d'une Transformation
 - le Processus de Décision et de Contrôle (la "Gouvernance")
 - le Processus de Planification et les Processus de suivi
 - la Gestion des Changements en cours de projet
 - la Documentation à générer
- **Savoir être** : préparer les Transformateurs Métier à séparer la structure du détail, et à ne pas compter que sur le respect « scolaire » du Processus de Transformation pour faire réussir leur projet.
- **Connaissance des Fondations** : il faut d'abord faire passer l'idée qu'il est possible de Construire des Solutions pour des besoins très **spécifiques** à l'aide de Fondations **communes**. Ce principe est à la base de la réutilisation, mais n'a rien d'évident. Une fois ce principe acquis il faut expliciter la variété des Composants que l'on peut réutiliser. Si les Fondations n'existent pas encore, il faut mettre en valeur les potentialités, si elles existent il faut aussi les présenter.
- Pour chaque Fonction d'Ingénierie il faut expliciter ce qui est à la charge de l'Équipe Métier et ce qui est à la charge de l'Équipe IT :
 - Définir le **Problème**,
 - Construire les **Entités-Métier** : créer, modifier ou réutiliser,
 - Construire les **Fonctions-Métier** :
 - Construire les **Processus-Métier**,
 - Construire les **Entités-Organisation**,
 - Construire les **Fonctions-Organisation**,
 - Construire les **Processus-Organisation**,
 - Construire les **Rôles**,
 - Construire les **Processus de Formation**,
 - Construire les **Processus de Migration** de données,
 - **Vérifier et Tester**

En guide de conclusion :

L'appétit en Solutions Compétitives va croître sensiblement, aiguisé par la crise financière.

Les Solutions Compétitives nécessitent une nouvelle Approche Coopérative qui représente un changement culturel important dont il ne faut pas sous-estimer la difficulté.

Les Entreprises ne doivent pas attendre pour se frotter à ce nouveau challenge : il faut informer les Directions Générales et lancer des projets pilotes pour faire la preuve de la qualité de cette démarche, avant de la généraliser.