

Éléments de gestion de projets

Alain Wafflard
<http://be.linkedin.com/in/alainwafflard/>

Éléments de gestion de projets

- 1.Définitions
- 2.Étude préliminaire
- 3.Initialisation
- 4.Réalisation
- 5.Pilotage
- 6.Acceptance
- 7.Déploiement et Migration
- 8.Communication et formation
- 9.Période de Garantie

Éléments de gestion de projets

1. Définitions

- 1) Qu'est-ce qu'un projet ?
- 2) Éléments d'un projet
- 3) Pourquoi la gestion de projets ?
- 4) Qu'est-ce que la gestion de projet ?

Qu'est-ce qu'un projet ?

- ISO 10006

« Processus unique, qui consiste en un ensemble d'activités coordonnées et maitrisées comportant des dates de début et de fin, entrepris dans le but d'atteindre un objectif conforme à des exigences spécifiques telles que les contraintes de délais, de coûts et de ressources »

- ISO 21500

Terminologie

- ISO : Organisation Internationale de Normalisation.

ISO

L'ISO établit et publie des normes internationales.

<http://www.iso.org/>

ISO 10006:2003

« Systèmes de management de la qualité -- Lignes directrices pour le management de la qualité dans les projets »

Cette norme donne des conseils sur l'application du management de la qualité aux projets. Elle est applicable à des projets de complexité variable, qu'ils soient petits ou grands, de courte ou longue durée, qui se situent dans des environnements différents, quel que soit le type de produit ou de processus de projet. Il peut être nécessaire d'adapter ces conseils à un projet précis.

L'ISO 10006:2003 ne constitue pas un guide pour le «management de projet» en lui-même, mais se contente de donner des conseils sur la qualité dans le cadre des processus de management de projet.

Qu'est-ce qu'un projet ?

- PRINCE2 ®

« A project is a temporary organisation that is needed to produce a unique and predefined outcome or result at a pre-specified time using predetermined resources. »

Terminologie

- PRINCE2 : PRojects IN Controlled Environments

PRINCE2®

PRINCE2 is a de facto standard developed and used extensively by the UK government. It embodies established and proven best practice in project management. There are six variables involved in any project, and therefore six aspects of project performance to be managed: Costs – Timescales – Quality – Scope – Risk – Benefits. PRINCE2 is an integrated framework of processes and themes that addresses the planning, delegation, monitoring and control of all these six aspects of project performance.

<http://www.prince-officialsite.com/>

ISO 21500:2012 « Lignes directrices sur le management de projet »

L'ISO 21500:2012 fournit des recommandations en matière de management de projet et peut être utilisée par tout type d'organisation, qu'elle soit publique, privée ou sous forme associative, ainsi que pour tout type de projet, quelle que soit sa complexité, sa taille ou sa durée.

L'ISO 21500:2012 donne une description de niveau macroscopique des concepts et processus qui sont considérés comme constituant de bonnes pratiques en matière de management de projet, mais ne donne pas de recommandations détaillées pour le management de programmes ou de portefeuilles de projets.

ISO 21500 est la première d'une série de normes relatives au management de projet. Elle est conçue dans le droit fil de normes internationales connexes telles que les normes ISO 10006:2003 « Systèmes de management de la qualité – Lignes directrices pour le management de la qualité dans les projets », ISO 10007:2003 « Systèmes de management de la qualité – Lignes directrices pour la gestion de la configuration », et ISO 31000:2009 « Management du risque – Principes et lignes directrices ».

Qu'est-ce qu'un projet ?

- PMBOK

«Entreprise temporaire décidée pour obtenir un produit ou un service unique »

- AFNOR

« Ensemble d'actions à réaliser pour satisfaire un objectif défini, dans le cadre d'une mission précise, et pour la réalisation desquelles on a identifié non seulement un début, mais aussi une fin. »

Terminologie

- ANSI : American National Standards Institute
- PMBoK : Project Management Body of Knowledge
- AFNOR : Association Française de Normalisation

ANSI : PMBoK

Ouvrage et marque édité par le Project Management Institute (PMI)

Le PMBOK documente les bonnes pratiques et les fondamentaux de la gestion de projet. Il a une portée générale et s'applique aux projets de nombreux secteurs, comme par exemple la construction, le logiciel, l'ingénierie et l'industrie, etc.

- Le PMBOK définit les 5 groupes de processus de management de projet suivants: Démarrage – Planification – Exécution - Surveillance et maîtrise – Clôture.
- Il aborde les dix domaines de compétences en management de projet : l'intégration, le périmètre, les délais, les coûts, la qualité, les ressources humaines, les communications, les risques, les approvisionnements, les stakeholders.
- 47 processus de management de projet sont présentés dans ces groupes de processus et domaines de connaissance, avec le détail des intrants, techniques, méthodes, outils et livrables (extrants).

version anglaise : ISBN 978-1933890517

version française : ISBN 978-1933890654

<http://www.pmi.org/>

AFNOR

L'AFNOR conçoit et déploie des solutions fondées sur les normes, organisé autour de quatre grands domaines de compétences : la normalisation, la certification, l'édition de solutions et services d'information technique et professionnelle et la formation.

<http://www.afnor.org/>

Qu'est-ce qu'un projet ?

- Moi

« Ensemble fini d'actions entreprises dans le but de livrer un résultat unique répondant à un besoin défini et compris collégialement, dans le respect des contraintes imposées, notamment en termes de qualité, de budget et de délai, mais aussi en termes de législation, de réactivité, de flexibilité, de garantie, de culture, etc. »

Exemples de projets

- Projets-ouvrage
 - Conception et réalisation
 - p.ex. un logiciel, un pont
- Projets-produit
 - Conception seule
 - p.ex. un nouveau modèle de voiture
- Projets internes
 - rationalisation des ressources
 - p.ex. la mise en phase du fonctionnement de deux entreprises après leur fusion

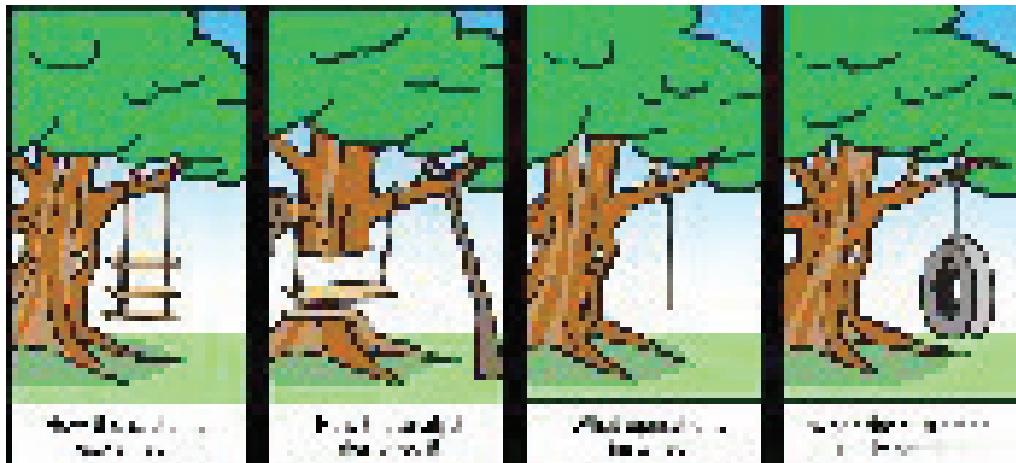
Eléments d'un projet

- Périmètre fonctionnel
 - réponse à un besoin défini
- Entreprise momentanée
 - ensemble d'actions coordonnées
- Delivrable fixé
 - unique et innovant
- Acteurs définis
 - Internes
 - Externes
 - Partenaires
 - Sous-traitants
 - Fournisseur

Eléments d'un projet

- Délai fixé
 - date de début
 - date de fin
- Budget fixé
- Contraintes techniques à respecter
- Contraintes légales à respecter

Pourquoi la gestion de projet ?



Diminuer au maximum les différences de compréhension et d'interprétation entre acteurs

Nombreuses contraintes :

- Temps
- Budget
- Ressources
- Connaissances disponibles
- Difficultés de découper le projet en tâches (à planifier, à prioriser et à assigner)
- Des risques à gérer
- Des clients à gérer et à cadrer
- Des collaborateurs à motiver
- Des conflits de personnes à régler
- Du stress à contrôler
- Des désaccords à solutionner
- Des challenges techniques à relever

Pourquoi la gestion de projet ?



Qu'est-ce que la gestion de projet ?

- « Organiser de bout en bout son déroulement »
- Anticiper
- Réaliser et/ou faire réaliser
- Monitorer (contrôler, mesurer)
 - Qualité
 - Délai
 - Coût
- Régler les conflits
- Maîtriser les contraintes
 - acteurs, délai, budget fixés
 - modifications fréquentes
- Réorganiser le projet

Qu'est-ce que la gestion de projet ?

■ Nous rêvons tous de ceci :



Qu'est-ce que la gestion de projet ?

Hélas, souvent ceci se passe :



Qu'est-ce que la gestion de projet ?

- Facteurs de réussite d'un projet
 - **Implication des utilisateurs 20%**
 - **Chef de projet expérimenté 15%**
 - **Objectifs clairs 15%**
 - **Soutien des responsables 15%**
 - Jalonnement du projet 10%
 - Autres 25%
 - Planification adéquate 5%
 - Appropriation 5%
 - Equipe compétente 5%
 - Besoins stables 5%
 - Autres 5%

Éléments de gestion de projets

2. Etude préliminaire

- 1) Pourquoi une étude préliminaire ?
- 2) Conception orientée client
- 3) Techniques de créativité
- 4) Business Model Canvas

Une étude de faisabilité doit permettre de décider si l'on prend le risque de se lancer dans le projet.
=> OK or KO ?

Pourquoi une étude préliminaire ?

- déterminer les objectifs du projet
- évaluer les investissements et humain et matériel nécessaires
- évaluer si l'organisation est apte à réaliser le projet
- estimer le retour sur investissement
- prendre la décision « do or don't »

Le but de l'étude préliminaire est de

- déterminer les objectifs du projet,
- évaluer les investissements et humain et matériel nécessaires
- évaluer si l'organisation est apte à réaliser le projet
- estimer le retour sur investissement
- prendre la décision « do or don't »

Les objectifs du projets doivent être précisés de façon claire, chiffrée et datée. Le résultat doit être conforme à des normes de **qualité** et de performances prédéfinies, pour le moindre **coût** et dans le meilleur **délai** possible.

D'une part, on estime si les bénéfices attendus seront en proportion des investissements engagés et du coût prévisionnel du projet. Pour de nombreux projets, on détermine ainsi le retour escompté sur investissement (ROI). Il faut toutefois noter que tous les projets ne visent pas forcément à atteindre un profit financier : on peut lancer un projet dans le but d'améliorer le service aux usagers d'une administration, ou pour améliorer le climat social d'une entreprise — dans ces cas, le retour sur investissement n'est pas nécessairement quantitatif.

D'autre part, l'étude de faisabilité détermine également si l'organisation est bien en mesure de mener le projet à son terme. On cherche en particulier à savoir si elle dispose des compétences, des ressources et des fonds nécessaires.

La décision de faire ou de ne pas faire arrive enfin :

- les risques de faire : quelles sont les difficultés auxquelles il faut s'attendre dans le déroulement du projet et les moyens de les prévenir,
- les risques de ne pas faire : quels sont les enjeux pour l'organisation si le projet n'était pas lancé et mené à terme.

Le projet ne sera véritablement lancé que si cette première phase est concluante.

Conception Orientée Client

- Motivation : Savoir
 - Ce que veut le client
 - Ce que cela signifie en termes de conception
 - Comment maximaliser la valeur du produit pour le client
- Méthodes
 - Analyse fonctionnelle
 - Value Analysis
 - Quality Function Deployment
 - Conjoint Analysis

Analyse Fonctionnelle

L'Analyse Fonctionnelle est une méthode qui permet d'avoir une image claire et complète des fonctions qu'un client attend d'un concept, afin de trouver une solution de principe au problème de la conception.

Value Analysis (VA)

VA combine une procédure structurée avec les techniques de créativité. Elle consiste à remanier un produit afin qu'il soit optimal du point de vue des coûts, des fonctions et de la qualité.

Quality Function Deployment (QFD)

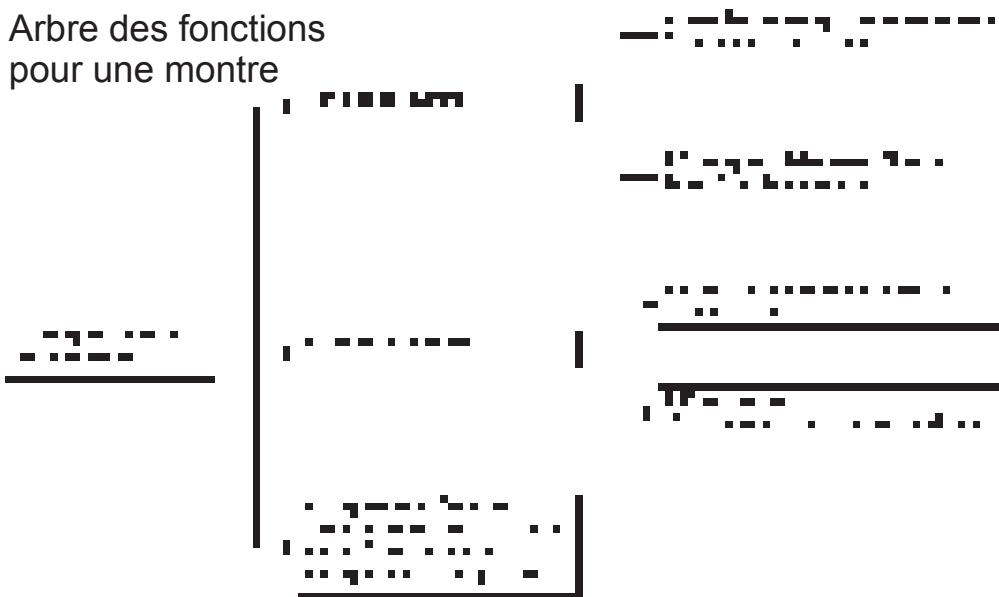
QFD est une méthode systématique pour traduire les souhaits des clients et du marché en exigences techniques.

Conjoint Analysis (CA)

CA est une technique de marketing visant à déterminer les caractéristiques des produits pour un segment du marché.

Analyse fonctionnelle

Arbre des fonctions pour une montre



L'Analyse Fonctionnelle est une méthode qui permet d'avoir une image claire et complète des fonctions qu'un client attend d'un concept, afin de trouver une solution de principe au problème de la conception. L'Analyse Fonctionnelle fait abstraction du produit physique.

1. Un organigramme forme le point de départ pour déterminer les limites du système et pour faire entrer les interactions par rapport à l'environnement en ligne de compte.
2. La deuxième phase est un brainstorming pour une série de fonctions utiles : les fonctions de base, qui sont la raison d'être de l'élément et les fonctions de support, qui améliorent l'élément. Les fonctions du produit sont décrites selon la forme standard : "substantif (+critère) + verbe".
3. Ces fonctions sont placées dans un arbre des fonctions et élaborées vers des niveaux plus détaillés en posant toujours la même question : "Comment (avec quelles sous-fonctions) vais-je réaliser cette fonction ?"
4. Ce procédé est répété jusqu'à ce qu'on obtienne des **fonctions concrètes et mesurables**. La description de la fonction se présente alors sous la forme : "substantif + critère + verbe"
5. Contrôlez la structure de l'arbre en commençant par les fonctions mesurables et en remontant vers le niveau le plus haut en posant la question : "A quoi sert cette fonction ?"

La structure des fonctions ainsi obtenue est un modèle abstrait du produit à concevoir. Elle décrit l'essence du problème de la conception. La force de l'Analyse Fonctionnelle réside dans le fait quelle permet de concevoir à un niveau abstrait des solutions alternatives à une problématique.

L'étape suivante de la conception, quand la structure de l'arbre des fonctions est pratiquement déterminée, consiste à concevoir des alternatives pour réaliser des fonctions concrètes : le function listing.

Les critères des fonctions mesurables peuvent servir de base pour la spécification du produit.

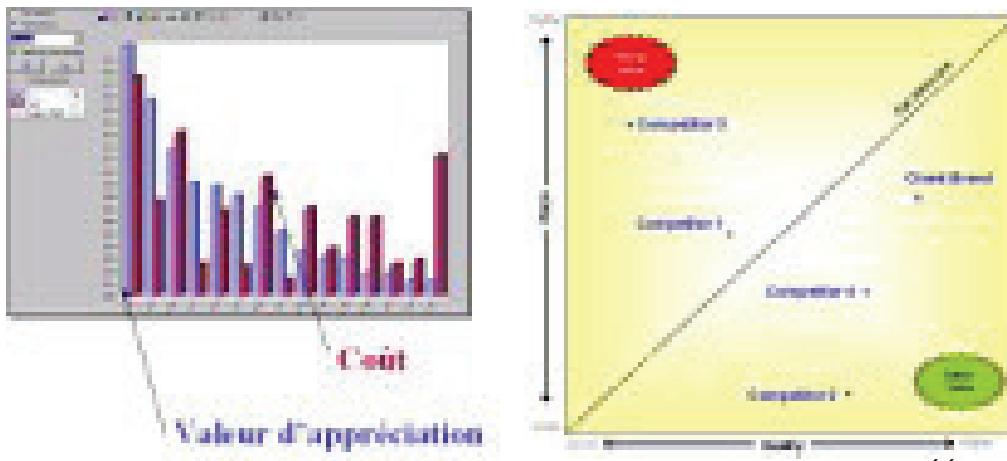
Value Analysis



- Valeur = fonction / coût
 - Fonction = tout ce qui incite l'utilisateur à désirer le produit ou le service
 - Coût = conception + développement + production + livraison + communication + overhead
- Exemple de la télécommande :
 - Valeur = Utilité de chaque bouton pour le client
 - Coût = développement et communication au client de chaque bouton

L'Analyse de la Valeur évalue la relation entre les coûts et la réalisation des fonctions de produits ou de services concrets, optimise le produit du point de vue des coûts de la réalisation des fonctions et de la qualité et élimine les coûts sans plus-value au produit.

Value Analysis



Conjoint Analysis

- Exemple:
 - Une entreprise veut vendre des gobelets.
 - Mais quelles caractéristiques ?
 - contenance: petite 150ml ou grande 300ml ?
 - matériau: plastique ou verre ?
 - prix de vente: 0.15€ ou 0.30€ ?
 - Déterminer, par analyse du marché, les variantes attrayantes.
 - $2 \times 2 \times 2 = 8$ variantes
 - On demande à plusieurs clients d'attribuer une "valeur d'utilité" aux 8 différents gobelets.
 - Sur la base de ces "valeurs d'utilité" et au moyen d'une méthode d'estimation, on calcule la probabilité de succès pour chaque choix de produit.

Prévoir quelle combinaison de caractéristiques un produit doit posséder pour répondre au mieux aux souhaits d'un segment du marché.

Etapes :

- détermination des caractéristiques les plus importantes du produit
- détermination du nombre de niveaux pour chaque caractéristique et leur valeur
- détermination d'une méthode pour la compilation de données
- l'analyse des résultats

Conjoint Analysis

No de produit.	Contenance	Matériaux	Prix	Utilité
1	150 ml	Plastique	0.15 €	8
2	300 ml	Plastique	0.15 €	9
3	150 ml	Verre	0.15 €	8
4	300 ml	Verre	0.15 €	11
5	150 ml	Plastique	0.30 €	6
6	300 ml	Plastique	0.30 €	7
7	150 ml	Verre	0.30 €	2
8	300 ml	Verre	0.30 €	4

Dans la réalité, le nombre de paramètres et le nombre de niveaux sont plus importants. Il est vraisemblable que le nombre des produits à évaluer devient trop important pour les soumettre aux clients. Dans ce cas, le Conjoint Analysis utilise des méthodes de compilation de données limitant le nombre de variantes à évaluer. Par exemple, on écartera les produits raisonnablement peu intéressants pour le client.

Techniques de créativité

	Méthode de travail intuitive	Méthode de travail systématique
Méthode de réflexion par association	Brainstorming Mindmapping Reverse brainstorming	TRIZ
Méthode de réflexion par confrontation	Synectics Abstraction progressive Wishful thinking	Analyse morphologique Function listing

La génération d'idées est une phase cruciale lors du développement de produits, pour générer des idées soit de produit, soit de solution. Le développement de concepts est une phase qui doit être traitée avec la plus grande attention, mais qui est souvent négligée par beaucoup de concepteurs. Au moment de la génération des idées et de leur développement, la structure finale du produit ou du processus est déterminée. L'entreprise qui aura développé la meilleure idée aura également le plus de chances de succès. Il est donc important de générer l'idée gagnante dans la phase de la génération d'idées. Mais il ne suffit pas qu'une entreprise dispose de collaborateurs créatifs, il faut également que leur créativité soit stimulée.

Les techniques de créativité existantes peuvent être réparties en 4 groupes, selon d'une part la méthode de travail et d'autre part la méthode de réflexion.

Beaucoup de méthodes sont appliquées en groupe. Ceci a l'avantage que les participants franchissent les barrières de réflexion de leurs collègues et que de cette façon le groupe obtient de meilleurs résultats que si les idées étaient générées individuellement.

Brainstorming



Le brainstorming est une technique de génération d'idées qui stimule la réflexion créative lors de la recherche de solutions pour un problème donné. Le brainstorming classique associe, connecte et combine différentes idées isolées pour en générer de nouvelles. Une session de brainstorming se tient généralement en groupe avec un animateur. Le résultat d'une session de brainstorming est un grand nombre d'idées non-vérifiées et très diverses dont l'intérêt pour résoudre le problème donné doit encore être évalué.

Déroulement d'une session classique de brainstorming

1. Après une introduction circonstanciée de la problématique, une équipe de personnes de différents horizons émettent oralement leurs idées concernant le problème.
2. Formulation et discussion du problème au sein de l'équipe.
3. Reformuler le problème en faisant "un pas en arrière" et en essayant d'identifier un maximum de facettes du problème. Chaque facette est formulée : "Comment allons-nous faire ceci ou cela ?".
4. Choix d'une formulation d'introduction de la session. Ce choix peut être autocratique (prise par l'animateur) ou démocratique (prise par le groupe).
5. L'échauffement : cette phase a pour but de mettre les participants en "roue libre". Pendant cinq minutes tous les participants émettent des idées diffuses. La **pollinisation croisée des idées** est très importante. Rien n'est mis sur papier.
6. Le brainstorming effectif : toutes les idées sont notées.
7. Evaluation et développement des idées : parmi les centaines d'idées qui auront été émises, il faut ensuite sélectionner, évaluer et développer celles qui sont les plus intéressantes. Cette phase du processus créatif est la plus difficile, et également celle qui prend le plus de temps, mais elle ne fait pas partie du brainstorming lui-même.

Ne pas tomber dans les pièges suivants

- Manque de confiance et peur de la critique, de sorte que peu d'idées sont générées.
- Critique, concurrence, attitude défensive.
- Interruptions, questions et explications : cassent le rythme du brainstorming
- Inépties : il faut travailler dans le but d'accomplir une tâche, il ne s'agit pas d'émettre des idées dénuées d'intérêt.
- Une définition trop large du problème, de sorte que les idées formulées ne sont pas suffisamment concrètes.

Mindmapping

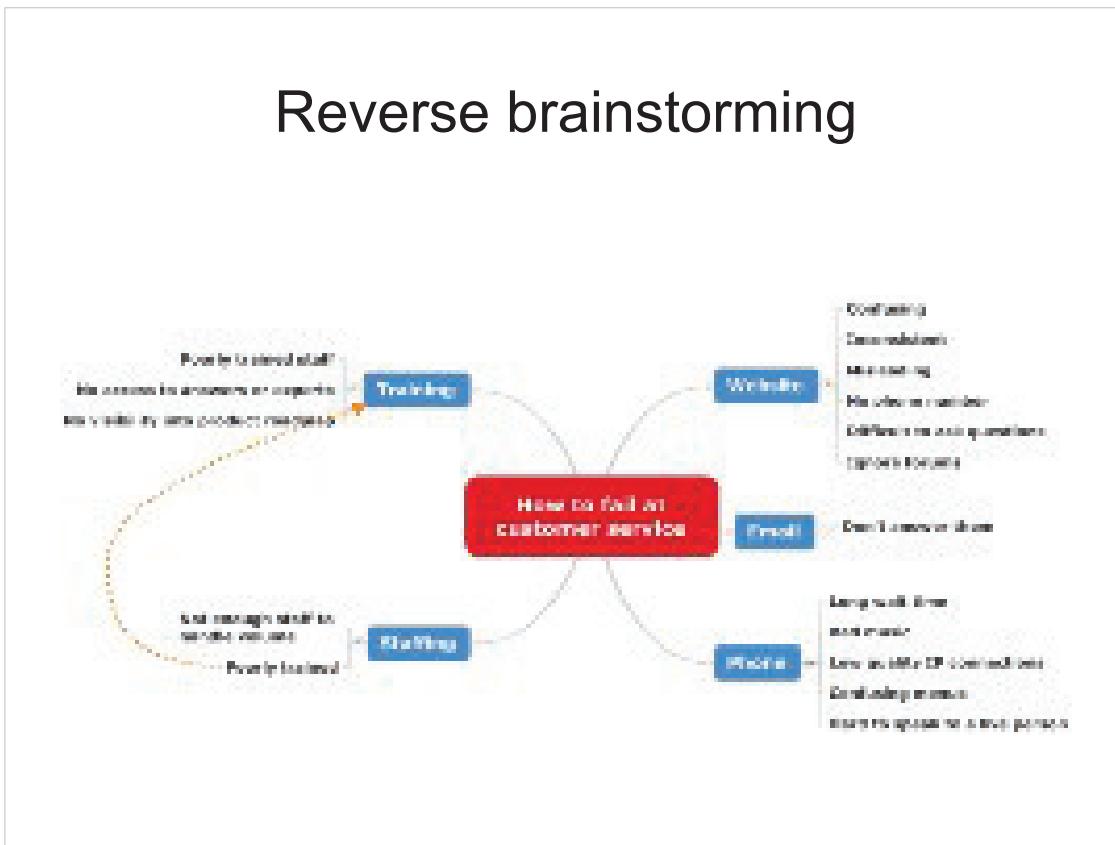


Une *mindmap* est une représentation graphique d'un concept entourée des idées qui sont en rapport avec celui-ci. Chacune de ces idées peut à son tour être développée de la même façon. Des branches, flèches et nœuds donnent une structure dans cette représentation du concept. Une mindmap peut être utilisée pour représenter des idées, donner une représentation claire (du contexte) d'un problème, générer des idées,

Le grand intérêt de *Mindmapping* est qu'il permet de représenter clairement beaucoup d'informations sur un espace limité. Le fait que les hommes ont une mémoire visuelle fortement développée fait de la technique du Mindmapping un important support pour transmettre des idées complexes.

<http://www.mind-mapping.org/>

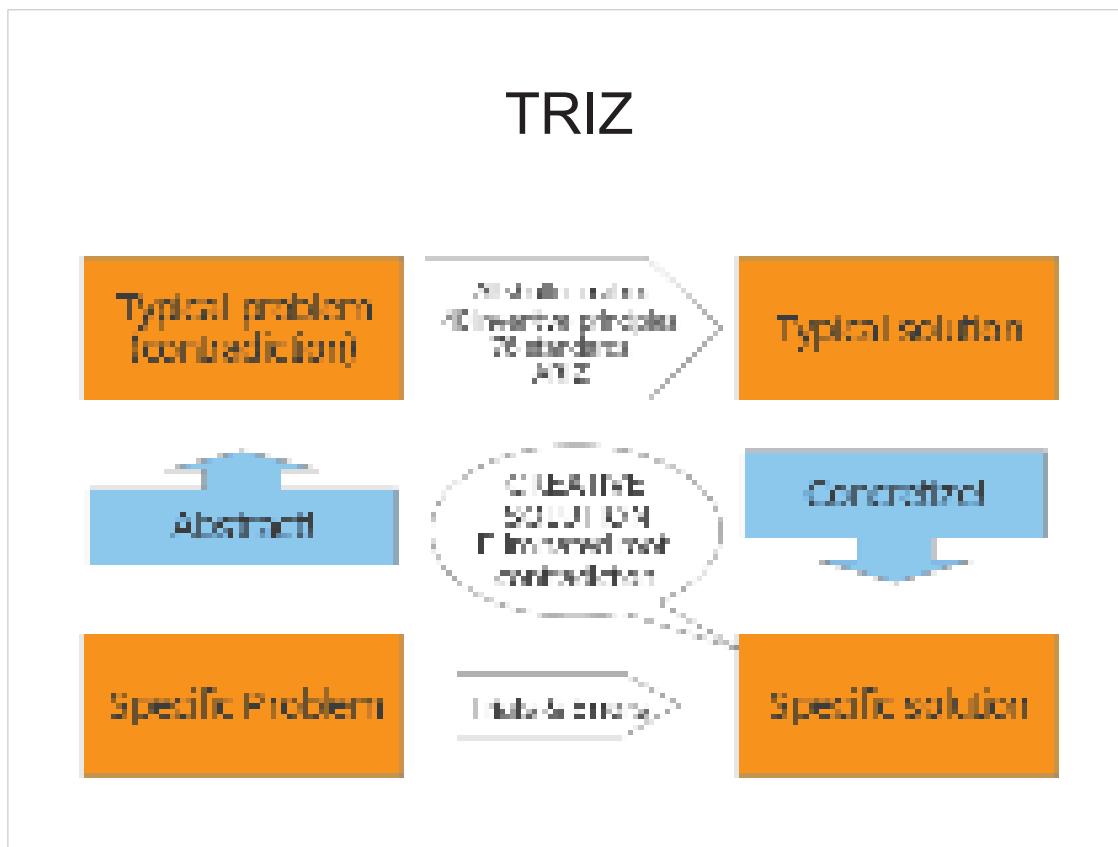
Reverse brainstorming



Le reverse brainstorming est une technique de créativité et d'évaluation d' idées qui consiste à énumérer tous les points faibles (erreurs éventuelles, défauts, ...) d'une proposition. Cette procédure se déroule de la même façon qu'une session de brainstorming classique.

Dans une phase suivante on peut se concentrer sur ces faiblesses et leur résolution. Un brainstorming classique peut être utilisé pour chercher des solutions pour les exigences fonctionnelles complémentaires.

En industrie, l'AMDEC (ou FMEA) est la méthode par excellence pour traiter les possibilités d'erreur. Dans l'IT, le reverse brainstorming peut être utilisé pour découvrir préventivement des bugs.



Triz -Theory of inventive problem solving- est une méthode pour solutionner des problèmes en innovant, qui a été développée par le Russe Genrich Altshuller. Il analysa un très grand nombre de brevets et chercha des modèles résurgents dans le processus d'innovation.

Il fit les constatations suivantes :

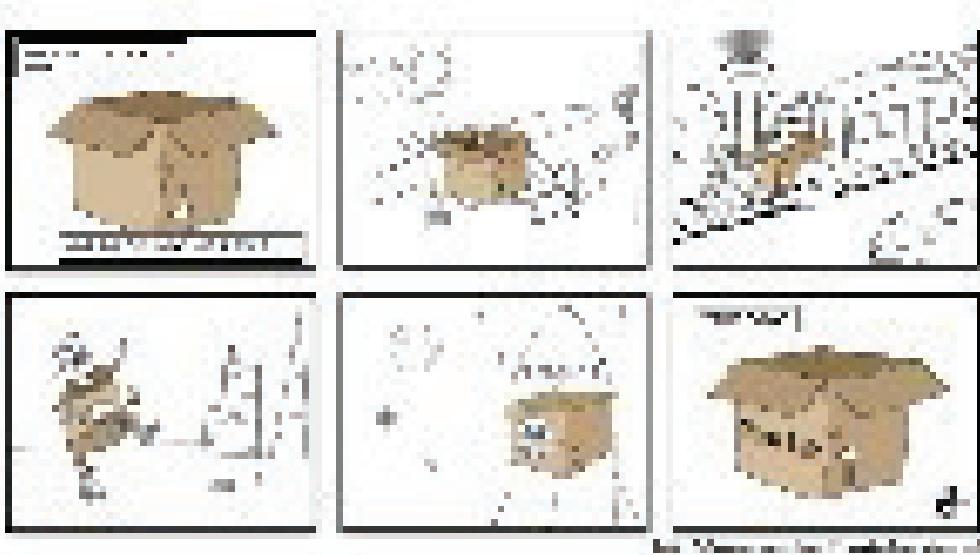
- Les innovations reposent souvent sur les mêmes solutions fondamentales, même si elles sont appliquées dans différents secteurs industriels et scientifiques.
- Il arrive souvent que la solution la plus évidente pour un problème donné ne soit pas applicable, étant donné qu'il faut répondre à des exigences opposées. Une solution n'est vraiment innovatrice qu'à partir du moment où elle élimine la contradiction.
- Les différentes inventions sont basées sur 40 principes généraux de solution (action périodique, segmentation, combinaison, ...).

TRIZ est une méthode qui indique la voie dans laquelle on a le plus de chances de trouver la solution à un problème innovatif concret.

La théorie, en bref : Pour décrire un système, Altshuller distingue 39 aspects généraux (poids, taille, solidité, ...). Pour appliquer TRIZ on formule un problème concret inventif en termes de ces aspects généraux, éventuellement contradictoires. Alsthuller a établi une 'une matrice de contradictions techniques' comportant des aspects apparemment contradictoires . On peut vérifier pour chaque problème concret quelles contradictions de la matrice doivent être résolues. Lors de l'analyse des brevets, il effectua systématiquement cette traduction avec l'aide de ses étudiants, et il rechercha de quelle manière ces contradictions avaient été levées. Pour chaque contradiction il répertoria alors les principes inventifs les plus fréquents. Il recommande finalement d'appliquer ces principes pour trouver une solution au problème posé.

Par exemple, l'antenne d'une radio portative doit être courte pour être transportable mais longue pour assurer une bonne réception. La solution à ce problème est l'antenne télescopique.

Synectics



Tout comme Brainstorming, Synectics est une technique de créativité de groupe. Le groupe est restreint et ne compte généralement pas plus de 7 membres : un animateur, un client et cinq participants. L'animateur organise le processus global et ne participe pas à la partie créative.

Le client présente clairement le problème au groupe et formule le résultat souhaité.

Les participants reformulent alors le problème en collaboration avec le client en posant des questions qui commencent par « Comment ... ? » ou « Je voudrais ... ». Le but de cette phase est de proposer le plus possible de reformulations intéressantes et de les éclairer brièvement. Il n'est pas permis d'émettre de critiques. Le client choisit une ou deux reformulations et argumente son choix.

On demande alors aux participants de générer des idées en rapport avec les formulations sélectionnées. Les autres participants peuvent les développer brièvement. Afin d'éviter tout malentendu le client reformulera l'idée en ses propres termes. Au moyen de trois réactions ponctuelles il indique alors les points positifs de l'idée.

Ensuite il indique, au moyen d'une formulation commençant par "comment", la direction dans laquelle il voudrait que cette idée soit développée.

L'étape précédente est répétée plusieurs fois jusqu'à ce qu'on obtienne plusieurs solutions envisageables. Le client évalue les solutions possibles qui ont été trouvées et indique ce qu'il va entreprendre pour les réaliser.

L'avantage de cette méthode par rapport au brainstorming est qu'on obtient un résultat beaucoup plus rapidement. Les phases divergentes et analytiques se suivent rapidement. On demande régulièrement au client de déterminer la direction du développement de l'idée. Un des désavantages est qu'on risque plus facilement négliger des courants d'idées intéressantes.

<http://synecticsworld.com/>

Abstraction progressive

Niv.abstr. 1				relier			
Niv.abstr. 2		enfiler			serrage		
Alternative	agrafe		classeur		bombe		classeur avec syst. de serrage

L'abstraction progressive est une technique de créativité qui aide le concepteur à prendre ses distances par rapport à la solution actuelle (ou la solution la plus évidente) à un problème de conception. Avec l'abstraction progressive le concepteur abordera les fonctions du concept à un niveau plus élevé, plus général et plus abstrait. A partir de ces niveaux supérieurs le concepteur a une vue plus globale du problème et il trouvera plus facilement des alternatives pour remplir les fonctions requises.

La stratégie de l'entreprise décide jusqu'à quel niveau on peut (doit) abstraire : un niveau plus élevé donnera un produit ayant un degré d'innovation plus élevé, mais qui demandera plus de temps et qui sera plus coûteux.

Cette méthode est d'application non seulement pour le développement de produit, mais aussi pour tout problème lié à la fonctionnalité des produits.

Exemple

Un concepteur cherche un nouveau produit qui pourrait remplacer les agrafes. Comme point de départ il a le produit actuel et sa fonction principale : "enfiler".

Un concept alternatif partant du même niveau est un classeur. Les deux se basent sur des principes analogues.

Un niveau d'abstraction supérieur d'enfiler est "relier". Etant donné qu'à part enfiler, il y a encore d'autres manières de relier, p.ex. "coller" ou "serrer", la vision du concepteur sera élargie s'il part de la fonction "relier".

Wishful thinking

- « prendre ses rêves pour la réalité »
- **Exemple de problème**
 - Trouver une technique pour séparer deux produits synthétiques.
- **Solution idéale**
 - Que les deux produits ne doivent plus être séparés.
- **Nouvelle problématique**
 - Développer un nouveau produit synthétique utilisant le mélange des deux produits à recycler comme base.
OU BIEN
 - Développer des produits utilisant des produits synthétiques compatibles.

Wishful thinking est une technique qui est en premier lieu utilisée pour définir les problèmes, mais qui peut également être considérée comme étant une technique de créativité. L'aspect créatif consiste à approcher le problème sous un angle différent.

Avec Wishful thinking, le concepteur se pose la question de savoir quelle serait la meilleure solution à un problème. On peut ainsi obtenir une nouvelle définition du problème dont les solutions approchent le résultat idéal.

Exemple

Problème de recyclage :

trouvez une technique pour séparer deux produits synthétiques.

Solution idéale :

les deux produits ne doivent pas être séparés.

Nouvelle problématique :

développez un nouveau produit synthétique utilisant le mélange des deux produits à recycler comme base,

ou

recyclez (et développez) des produits utilisant des produits synthétiques compatibles (cette solution sera surtout intéressante pour les entreprises qui sont obligées prendre en charge le recyclage de leurs propres produits).

Les solutions pour cette nouvelle définition du problème seront générées au moyen des autres techniques de génération d'idées.

Analyse morphologique

Température	Consistance - Forme	Destination
chaud	liquide	accompagnement
tempérée	conservé	apéritif
tiède	gelé	dîner
froid	gelé	déjeuner
	poudre	dessert
	solide	plat
		entrée
		plat principal

C'est à l'astronome suisse Fritz Zwicky que l'on doit l'analyse morphologique. Conçu dans les années 1940 et diffusée fin des années 1960, cet outil lui a permis de démontrer l'existence (théorique) d'étoiles non encore répertoriées. Il l'a également utilisée plus tard pour la mise au point de fusées.

Principales étapes

- Choisir le sujet d'analyse et les axes qui vont retenir l'attention ;
- Pour chacun, décliner les ou des possibilités ou alternatives ;
- Consigner le tout dans un tableau ou une matrice ;
- Evaluer les combinaisons.

Cas pratique

Un producteur de produits de bouche (dont foie gras) espère élargir son offre avec une voire plusieurs formules novatrices. Il retient trois axes - consistance, température et moment ou destination de la consommation - qu'il décline pour former ce tableau (voir slide).

Le plan de bataille est prêt, reste à partir au combat c'est-à-dire évaluer les combinaisons :

- chaud / liquide / accompagnement
- chaud / liquide / apéritif
- ... / ... / ...
- tiède / solide / entrée
- tiède / solide / plat principal

Différentes mises en forme

L'analyse morphologique peut se mettre en scène sous différentes formes. Il semble que Zwicky donnait la préférence à des matrices carrées en deux ou en trois dimensions. Dans ce cas, on obtient des croisements plus linéaires ou systématiques. L'alternative - une matrice rectangle (cf. exemple) - permet quant à elle de former des combinaisons plus complexes. Idéalement, il y a lieu d'évaluer toutes les combinaisons livrées par la matrice ou le tableau. Pratiquement, ce n'est pas toujours envisageable ; cet exemple nous donne déjà 1440 combinaisons différentes !

Function Listing

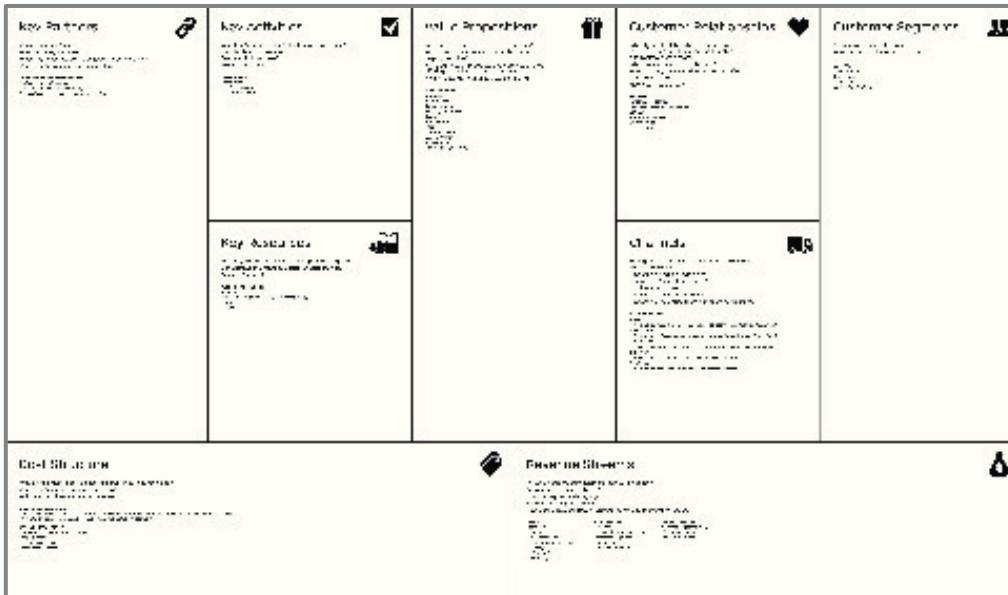
1.1 Montrer l'heure et la date sur l'écran	Montrer l'heure (à l'écran)
1.2 Intégrer l'heure et la date dans un calendrier de l'heure	Calendrier
2.1 Mettre en place le fonctionnement de la montre	Horloge
2.2 Utiliser une interface graphique pour la montre	Interface graphique
3.1 Fonctionnalité	Fonctionnalité

Le « function listing » consiste à générer plusieurs solutions alternatives pour les fonctions requises pour le concept. Cette liste des fonctions requises, ou « arbre des fonctions » sera généralement le résultat d'une analyse fonctionnelle.

Nous reprenons l'exemple de la montre vu pour l'analyse fonctionnelle, et l'arbre des fonctions qui en est le fruit. Dans le tableau (cf slide), nous recherchons des solutions pour ces fonctions requises.

L'étape suivante dans la conception consistera à sélectionner, combiner et éventuellement développer ces solutions partielles.

Business Model Canvas



Le Business Model Canvas : Un outil stratégique pour l'entreprise

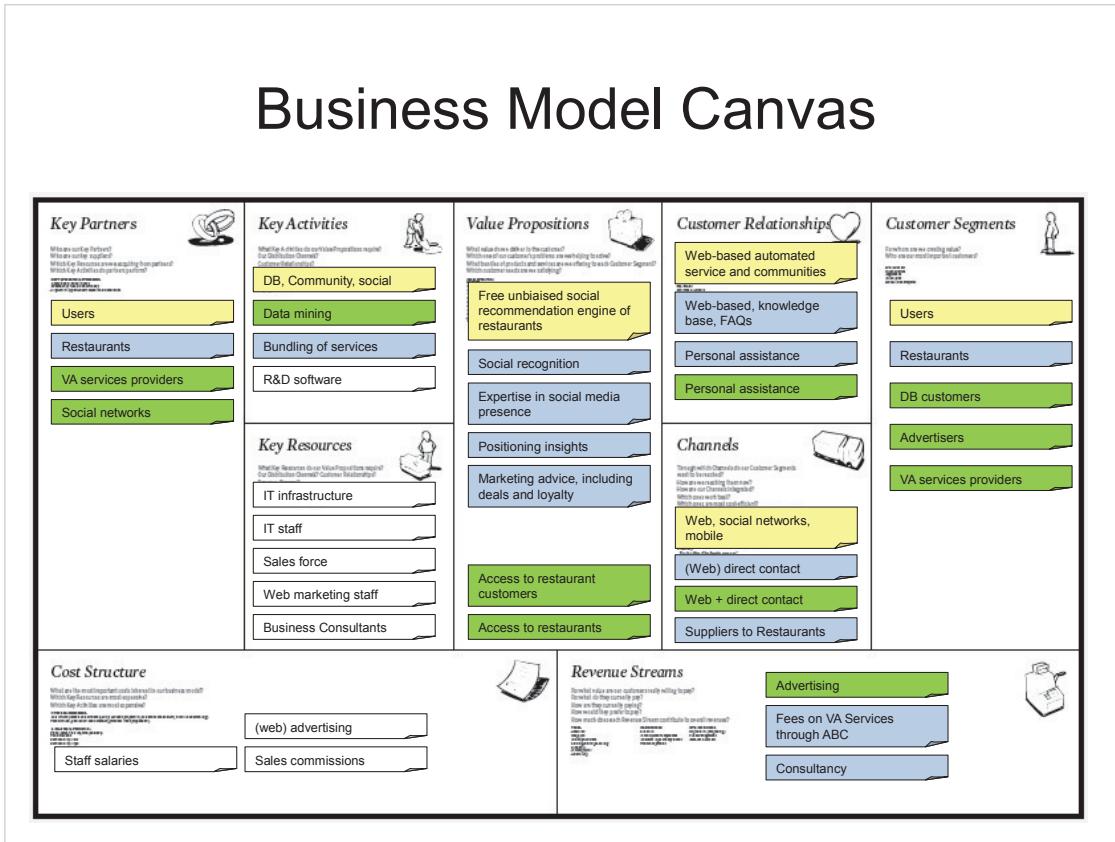
Tout entrepreneur établit un jour son plan d'affaires, c'est à dire met en lumière la faisabilité commerciale, technique et financière de leur projet et ainsi déterminer sa rentabilité. La réalisation de cette étude peut être simplifiée si le chef d'entreprise a une vision claire de la création de valeur qu'il souhaite mettre en œuvre, qui est la rencontre entre les attentes réelles de la clientèle, des produits ou services proposés, les canaux de distribution, le processus de production, les partenaires etc.

Outil de modélisation et de réflexion stratégique, le Business Model Canvas d'Alexander Osterwalder valide la solidité d'une action, d'un plan de développement ou la mise en œuvre d'un projet, d'une idée grâce à la visualisation et à l'interaction de neuf blocs bien identifiés et éléments indispensables à la bonne réussite de tout projet. Il s'articule autour de 9 blocs interactifs :

- le produit
- la clientèle cible
- les relations "clientèle"
- les canaux de distribution
- les activités de l'entreprise
- les ressources clés
- les partenaires
- les coûts occasionnés
- les rentrées engendrées

Chacun d'entre eux représente un élément des processus permettant la création de valeur. Quelque soit le bloc choisi pour démarrer l'analyse, il aura toujours un impact sur l'ensemble du modèle. Ainsi, cet outil garantit une analyse complète sans omission d'éléments ou de blocs.

<http://www.ucm.be/Entreprendre/Le-Business-Model-Canvas-Un-outil-strategique-pour-l-entreprise>



Cas d'application

Le cas représenté ci-dessus est réel. Il décrit le business model d'une entreprise désireuse gérant un moteur de recherche web 2.0 spécialisé dans la restauration, avec les fonctionnalités telles que recherche par lieu géographique, cotation et commentaire par les utilisateurs, budget approximatif, etc.

Chaque assemblage (ensemble de post-it de même couleur) détermine une stratégie et une offre de produits ou services ayant une véritable valeur ajoutée. Lorsque le business model sera complété dans ce canvas, càd lorsque le chef d'entreprise aura choisi sa stratégie et le modèle de fonctionnement des processus, il pourra s'atteler à réaliser son plan d'affaires, en s'inspirant du Business Model Canvas.

Méthode

- On complète le BMC à partir du template vierge, généralement (mais non obligatoirement) par la case « customer » et on indique dans des « post it » les différents segments clients du business model, généralement (mais non obligatoirement) en appliquant une couleur par segment.
- On définit ensuite la « value proposition » en ajoutant des post-it, en gardant la cohérence des couleurs de post-it choisies dans le premier bloc.
- On doit arriver obligatoirement à remplir tous les blocs, pour tous les customer segments, pour terminer le travail.

On peut travailler sur papier avec des post-it ou sur ordinateur avec un logiciel de présentation tel que MS Powerpoint, OpenOffice Impress, ou avec un mindmapper tel que XMind (qui contient ce template).

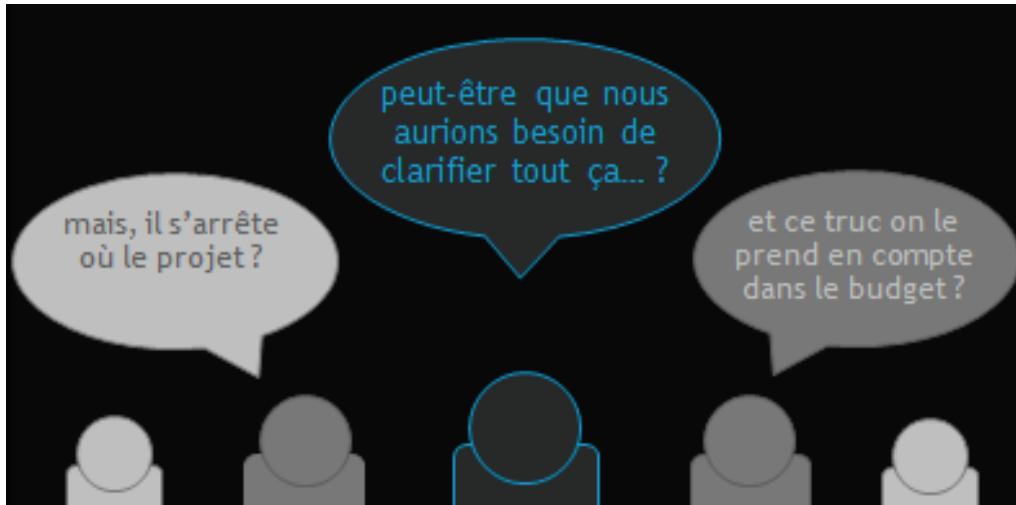
Éléments de gestion de projets

3. Initialisation

- 1) Spécifications détaillées**
- 2) Ressources humaines
- 3) Découpage
- 4) Planification du travail
- 5) Gestion des risques
- 6) Planification de l'acceptance
- 7) Budgétisation
- 8) Communication interne

Avant le démarrage du projet, il y a la phase – cruciale – de sa préparation, constituée de la budgétisation, la détermination suffisamment détaillée des tâches, la planification de celles-ci, la détermination des ressources nécessaires (humaines et matérielles).

Périmètre



Le **périmètre** du projet correspond à la délimitation précise du projet.

Pour un projet IT (mise en place d'un nouvel ERP, évolution d'un système IT en fonction d'une nouvelle organisation ...), le périmètre total est l'identification et le recensement des applications et modules impactés par le projet.

Le projet peut être ensuite sub-divisé en « sous-projets » (ou « workpackages ») possédant chacun son propre périmètre.

Cas d'entreprise

Deux sous-projets ont été mis en place dans une grande entreprise pour répondre au projet de bascule à l'euro de l'ensemble de son Système d'Information comptable. Le projet est nommé BINGO.

Le premier sous-projet, intitulé BINGO 1 a eu pour périmètre les modules du SI liés à la comptabilité auxiliaire client, la comptabilité auxiliaire fournisseur, la comptabilité auxiliaire des immobilisations et la comptabilité auxiliaire de caisse. Ce projet a été mené à son terme avant le commencement du second.

Le second sous-projet, intitulé BINGO 2 a eu pour périmètre les modules du SI liés à la comptabilité auxiliaire des opérations financières, le module de bouclage. Le lotissement a donc consisté à effectuer la bascule à l'euro en deux temps pour limiter les risques avec une phase transitoire de co-existence d'un système en franc et d'un système en euro.

Les trois acteurs principaux

- maître d'ouvrage (entreprise)
- maître d'œuvre (entreprise)
- chef de projet (personne)

Maître d'ouvrage (MOA)

Le MOA est à l'origine de l'expression d'un besoin qui est l'objectif du projet à atteindre. Il doit décrire le besoin dans un document, souvent nommé « cahier des charges fonctionnel » ou spécifications fonctionnelles. Dans le cas d'un projet IT, le MOA est aussi chargée de préparer des cas de tests fonctionnels pour vérifier que les développements ou paramétrages effectués par la MOE fonctionnent.

Exemple de MOA : Dans le cas d'un projet d'implémentation du module « contrôle de gestion » du progiciel SAP, le MOA est constituée des contrôleurs de gestion impliqués sur le projet.

Maître d'œuvre (MOE)

Le MOE prend connaissance du besoin exprimé et tâche d'y répondre par des installations, configurations et développements informatiques. Pour ce faire, il rédige un dossier de réponse aux besoins, nommé « cahier des charges technique » ou dossier de paramétrage ou dossier de conception général, ou encore étude technique.

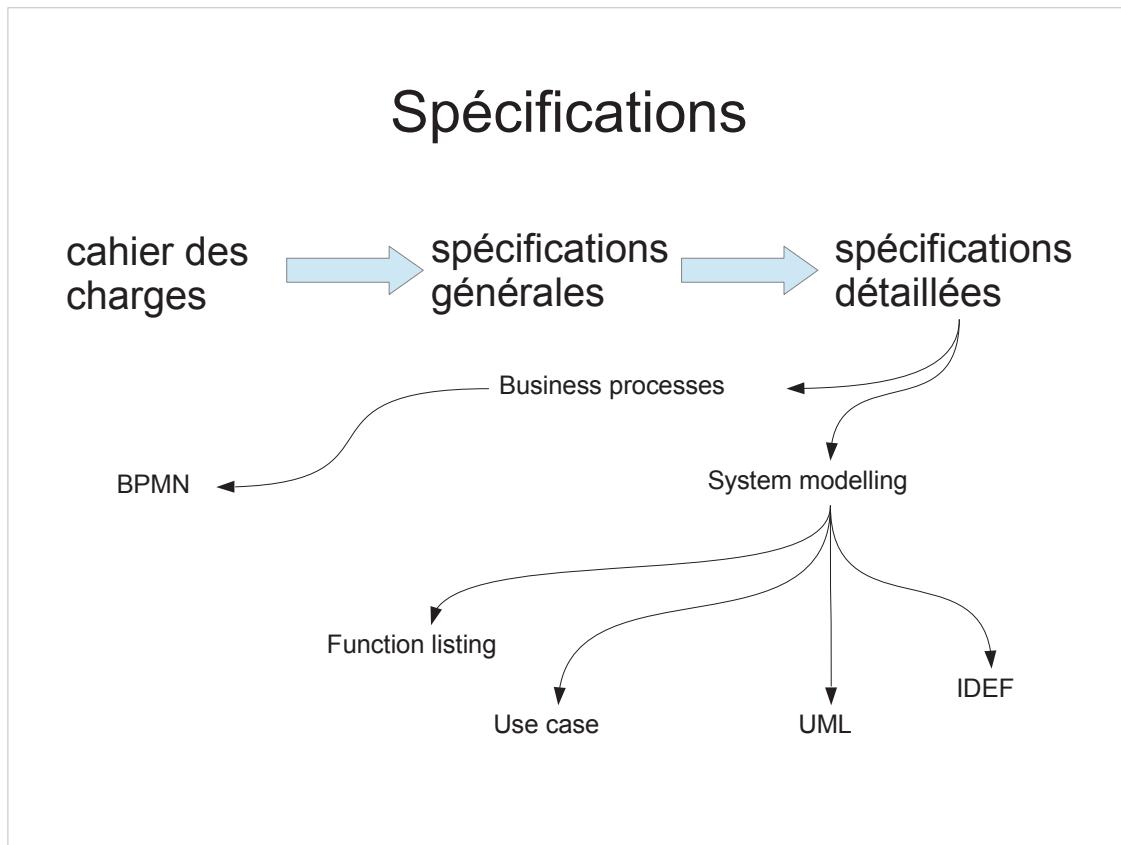
Exemple de MOE : Dans le cas d'un projet d'implémentation du module « contrôle de gestion » du progiciel SAP, le MOE est le service informatique qui devra répondre de cette implantation.

Chef de projet

Cette personne est chargée de mener le projet à bien. Caractéristique indispensable : être tourné vers les autres. Le rôle du chef de projet est complexe :

- Interface entre le client et l'équipe
- Pilotage et maîtrise des composants du projet : planning, budget, ressources, qualité des délivrables, sous-traitants, ...

Sa devise est « Get thing done on time and on budget ».



Cette phase a pour but de spécifier ce qui doit être réalisé ou fabriqué pour atteindre l'objectif, ce qui constituera le cahier des charges.

Il y a plusieurs niveaux de spécifications.

- Le « **cahier des charges** » ou « **Requirements Specifications** » est écrit par le MOA et définit en des termes informels (langage naturel) les objectifs à atteindre, les délais imposés, les coûts maximum, les contraintes, la sécurité, la fiabilité, la performance (temps de réponse), les exigences sur l'interface utilisateur, et d'une manière générale toutes les exigences nécessaires.
- Les « **spécifications générales** » découlent du cahier des charges et formalise (càd exprimer techniquement) toutes les fonctionnalités de l'objectif exprimé par les utilisateurs. C'est la première tâche d'initialisation d'un projet.
- Les « **spécifications détaillées** » ou « **Functional Specifications** » détaillent le fonctionnement interne du logiciel. Il s'agit bien entendu de documents beaucoup plus techniques. On peut y parler de fichiers, de tables, de serveurs, etc. Elles sont écrites pendant la réalisation du projet. Nous y reviendrons plus tard.

Les spécifications détaillées peuvent tirer profit les **méthodes de modélisation** de systèmes IT :

- **systèmes IT :**
 - Unified Modeling Language : UML
 - Integration DEFinition : IDEF
- **processus business :**
 - Business Process Model and Notation (BPMN)
 - Role Activity Diagram (RAD)

Ces méthodes conviennent aussi bien pour décrire l'existant (« as-is ») que le projeté (« to-be »), aussi bien pour expliciter les fonctionnalités que les recherches de solution.

Éléments de gestion de projets

3. Initialisation

- 1) Spécifications détaillées
 - IDEF
 - BPMN
- 2) Ressources humaines
- 3) Découpage
- 4) Planification du travail
- 5) Gestion des risques
- 6) Planification de l'acceptance
- 7) Budgétisation
- 8) Communication interne

IDEF

- IDEF
 - « Integration DEFinition »
 - langages de modélisation de systèmes
 - Domaines : manufacture et génie logiciel
- A maturité
 - IDEF0 : modélisation fonctionnelle
 - IDEF1X : modèles d'information et conception de base de données

IDEF, qui signifie « Integration DEFinition » est une famille de langages de modélisation appropriés aux systèmes (IT ou autre) et au génie logiciel. Ils couvrent en théorie un large éventail d'utilisations tels que la modélisation fonctionnelle de données, la simulation, l'analyse orientée objet, l'acquisition de connaissances, quoique seulement certains d'entre eux sont arrivés à maturité.

Les composants les plus mûrs et les plus utilisés de la famille IDEF sont

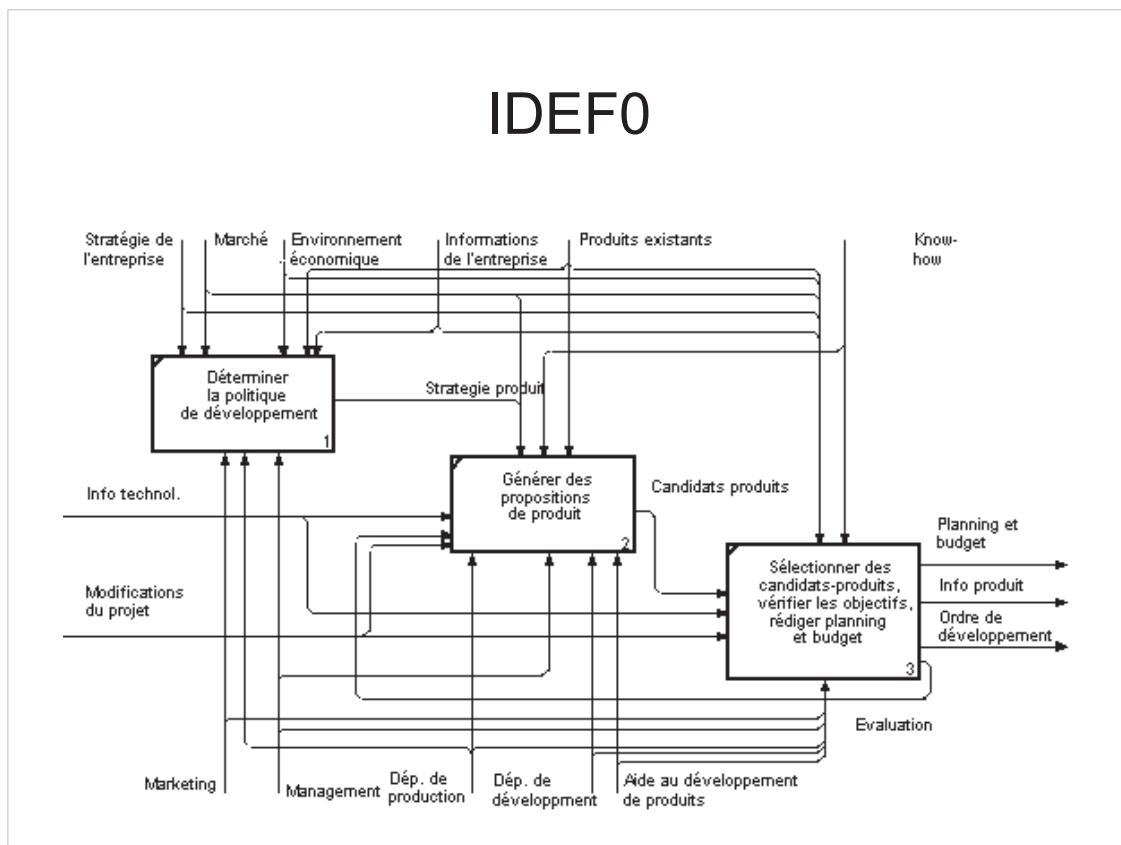
- IDEF0 : modélisation fonctionnelle
- IDEF1X : modèles d'information et conception de base de données

Pour information, voici les autres composants IDEF, incomplètement définis (et non utilisés) :

- IDEF2 : simulation (de systèmes manufacturiers)
- IDEF3 : modélisation de processus
- IDEF4 : conception orientée objet
- IDEF5 : description des ontologies
- IDEF6 : capture du raisonnement rationnel
- IDEF8 : modélisation des interfaces utilisateur
- IDEF9 : analyse des contraintes d'entreprise
- IDEF14 : conception de réseaux

Terminologie

- Ontologie : Ensemble structuré des concepts représentant un domaine de connaissance



IDEFO est le plus souvent utilisé pour la modélisation des processus de production et de service, ainsi qu'en tant que support pour le « business process reengineering » (BPR). Cette technique convient parfaitement pour la modélisation des activités et du flux des informations entre les activités. IDEF0 a une structure hiérarchique.

Avantages

- Les modèles les plus complexes restent clairs, étant donné que les détails sont représentés à différents niveaux.
- IDEF0 peut traiter des modèles volumineux.

Inconvénients

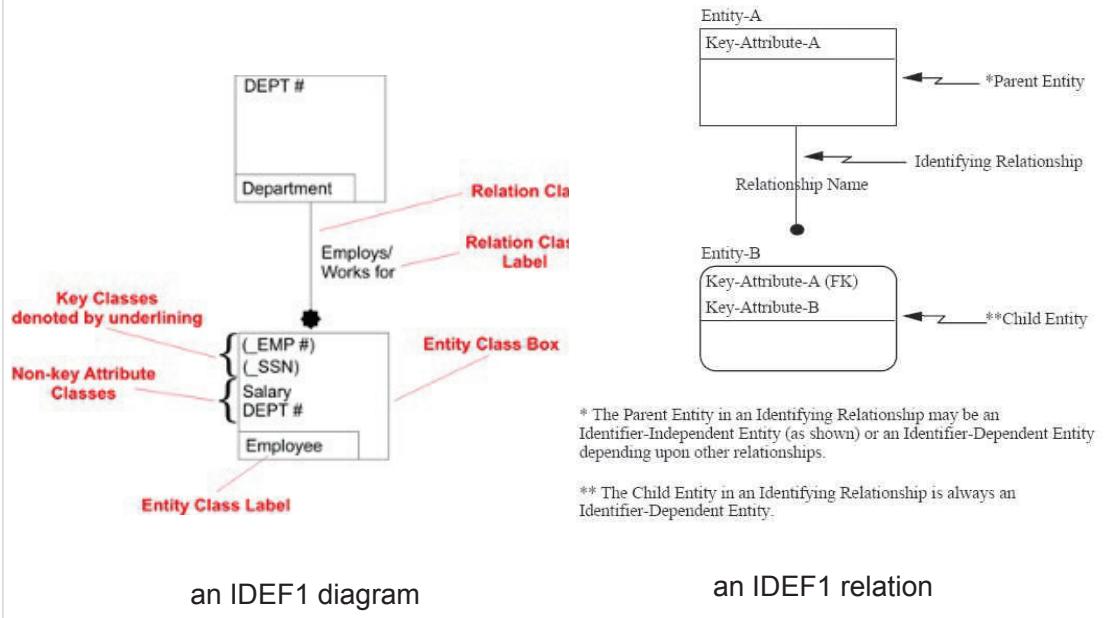
- Non gestion des rôles et des fonctions des personnes impliquées
- Pas de modélisation du temps
- Pas de représentation des relations logiques (relations ET et OU).
- L'utilisateur n'a aucune vue générale sur le déroulement du système.
- Technique inappropriée pour optimiser un processus

Légende

- Les rectangles représentent les activités.
- Les flèches venant de gauche sont les entrées (de matière, d'information ...) destinées à être transformées.
- Les flèches partant vers la droite sont les sorties (de matière, d'information ...) produites par le système.
- Les flèches venant du haut sont des contrôles.
- Celles venant du bas sont des mécanismes.

Synonyme : Structured Analysis and Design Technique (SADT)

IDEF1X



IDEF1X est une méthode de conception de bases de données relationnelles dont la syntaxe permet de développer un schéma conceptuel. Ce schéma conceptuel représente les données d'entreprises indépendamment de leur localisation et de leur méthode d'accès. IDEF1X est d'application après que les données ont été clairement définies et que la décision d'implanter une base de données a été prise.

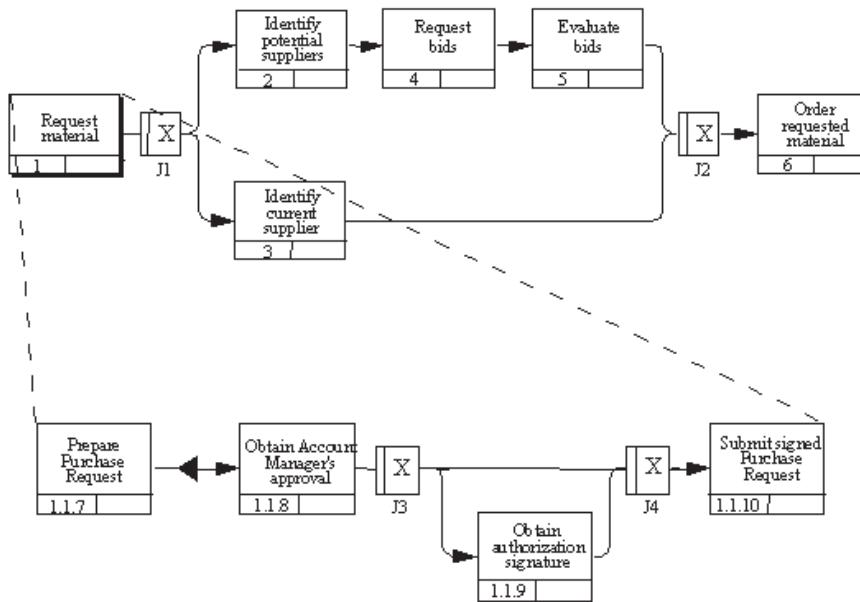
Si le système d'information sélectionné n'est pas une base de données relationnelles, mais par ex. un système orienté objet, alors IDEF1X n'est pas la méthode idéale.

Une faiblesse de presque toutes les méthodes de modélisation, y compris IDEF1X, est qu'elle nécessite une initiation et qu'elle n'est pas intuitive. Or, un mauvais démarrage dans une modélisation entraînera a posteriori son rejet.

Un exemple d'un concept souvent mal utilisé est celui de la relation visant à catégoriser. Il y a des cas où la catégorisation est nécessaire, mais il y en d'autres où il vaut mieux utiliser une taxonomie de données.

Enfin, IDEF1X n'est pas un langage de description de concept ou de terminologie. Pas question d'y exprimer des faits tels que « Une déclaration de travaux est un document et un contrat légal » ou « Un carré est un polygone à quatre égalité côtés ».

IDEF3 (processus)



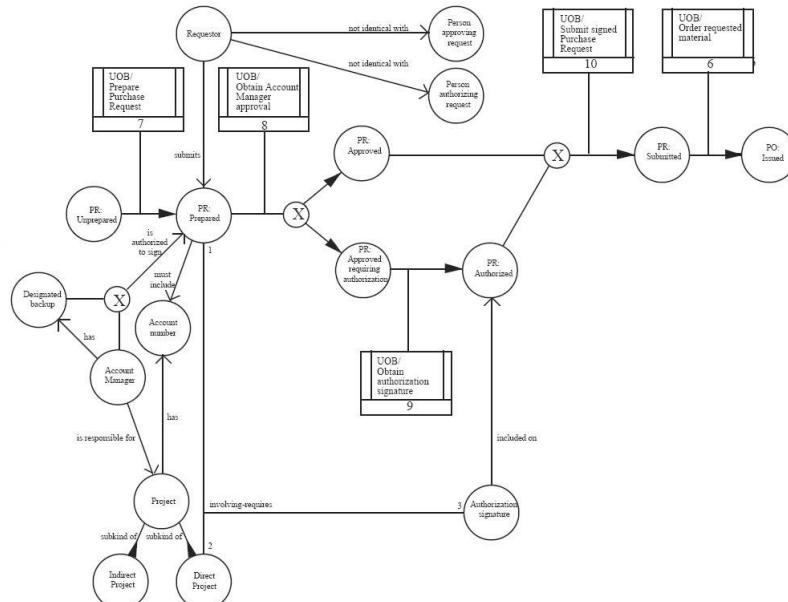
IDEF3, également connu sous le nom de « workflow diagramming », décrit les relations de succession et les relations causales entre les processus. Les diagrammes IDEF3 s'utilisent e.a. comme support pour le business process reengineering. L'objectif principal de IDEF3 est de fournir une méthode structurée par laquelle un expert du domaine peut exprimer des connaissances sur le fonctionnement d'un système ou d'une organisation.

Cette méthode de modélisation est basée partiellement sur IDEF0 et elle comporte également un aspect 'temps'. Le modèle obtenu peut servir à des fins de simulation. Contrairement à IDEF0, IDEF3 permet également d'établir des relations logiques. Le "X" dans la figure suivante signifie p.ex. qu'une seule des boucles est parcourue. La construction de IDEF3 est hiérarchique de sorte qu'on garde une vue d'ensemble même pour les grands projets.

Contrairement à IDEF0, IDEF3 ne permet pas d'utiliser ni des flèches de contrôle ni des flèches de mécanisme. IDEF3 offre par contre la possibilité d'utiliser trois sortes de flèches : des flèches de relation, d'object flow et de succession.

La méthode BPMN est le digne successeur de IDEF3. Plus moderne, elle intègre la notion de rôle et de fonctions dans la modélisation des processus.

IDEF3 (objet)

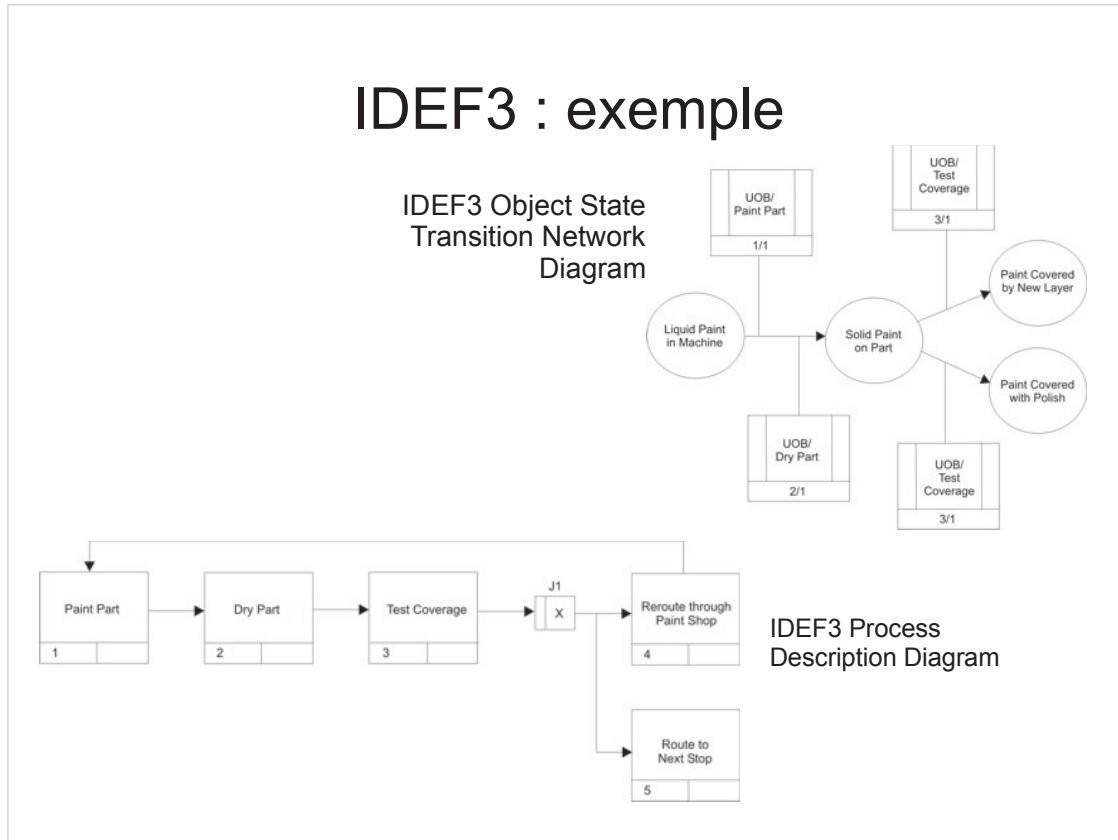


Modélisation orientée sur les processus

La schématique « processus » d'IDEF3 est le moyen de capturer, gérer et visualiser la connaissance sur les processus. Ces schémas fournissent un support graphique aidant les « business analysts » de différents domaines d'application à échanger leur connaissance sur les processus de l'entreprise. Ça inclut la connaissance sur les événements (events) et les activités, les objets qui participent à ces événements, et les contraintes qui régissent le comportement d'un événement.

Modélisation orientée sur les objets

La schématique « objet » d'IDEF3 est le moyen de capturer, gérer et visualiser les descriptions des processus centrées sur les objets, c'est à dire l'information qui spécifie comment des objets d'un certain genre sont transformés en d'autres genres ou changent d'état lorsqu'ils sont manipulés dans un processus.



The following example illustrates how the building blocks of the IDEF3 method can describe a scenario typically found in a manufacturing environment. The situation to be described is a painting and inspection process associated with applying primer paint to a part that will become an element of a subassembly for a piece of heavy construction equipment. The example IDEF3 description shown in Figure 1 is the graphical representation of the scenario (story) told by a paint shop supervisor when asked to describe: "What goes on in the primer shop?"

The story the example describes follows:

« Parts enter the shop ready for the primer coat to be applied. We apply one very heavy coat of primer paint at a very high temperature. The paint is allowed to dry in a bake oven after which a paint coverage test is performed on the part. If the test reveals that not enough primer paint has been sprayed on the surface of the part, the part is re-routed through the paint shop again. If the part passes the inspection, it is routed to the next stop in the process. Note the activities described in the scenario are clearly identified and appear as labeled boxes in Figure 1 and that the labeled boxes can describe activities, processes, events, etc. The IDEF3 term for elements represented by boxes is a Unit Of Behavior (UOB). The arrows (links) tie the boxes (activities) together and define the logical flows. The smaller boxes define junctions that provide a mechanism for introducing logic to the flows. »

Éléments de gestion de projets

3. Initialisation

- 1) Spécifications détaillées
 - IDEF
 - BPMN
- 2) Ressources humaines
- 3) Découpage
- 4) Planification du travail
- 5) Gestion des risques
- 6) Planification de l'acceptance
- 7) Budgétisation
- 8) Communication interne

BPMN

- Emanation de l'OMG
- Notation facilement compréhensible par tous les acteurs de l'entreprise, e.a.
 - Business analyst
 - Développeur
 - Manager
- Compatibilité avec les langages XML-based conçus pour l'exécution des business processses
 - p.ex. WSBPEL

Terminologie

- OMG : Object Management Group
- BP : Business Process
- BPMN : Business Process Model and Notation

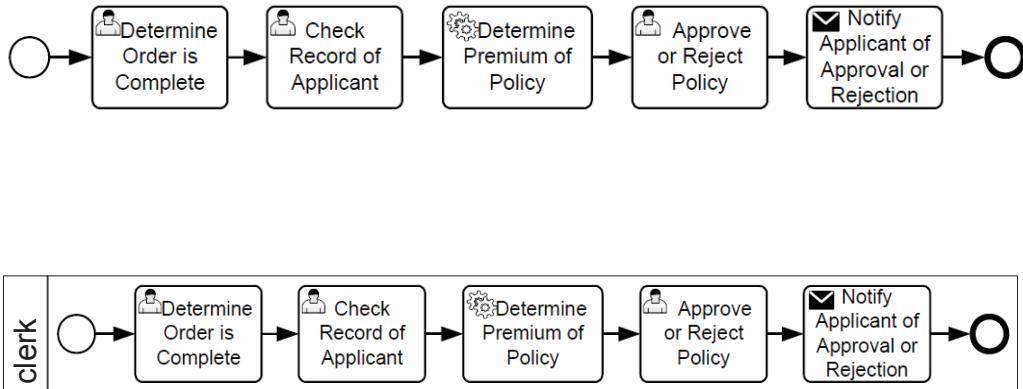
L'OMG est un consortium de normes technologiques sans but lucratif, international et d'affiliation libre. Fondée en 1989, ses membres sont des entreprises, des utilisateurs finaux, des établissements académiques et des organismes gouvernementaux. Les normes de l'OMG ont pour but d'intégrer les technologies issues de diverses entreprises par la standardisation. Parmi elles, citons le Unified Modeling Language (UML), le Model Driven Architecture (MDA), le Standards Customer Council (CSCC) et le groupe de standardisation de la qualité du logiciel (CISQ).

BPMN est également une émanation de l'OMG. BPMN est l'amalgame de plusieurs « best practices » issues de la modélisation de l'entreprise.

L'objectif principal de BPMN est de fournir une notation facilement compréhensible par tous les acteurs – internes et externes - de l'entreprise, en particulier pour les « business analysts » (qui créent les ébauches du processus), les développeurs (chargés de l'application qui va exécuter ces processus), et pour les managers (qui gèrent et contrôlent ces processus). BPMN peut être vu comme un pont standardisé entre la définition des businesses process et leur mise en œuvre.

Un autre objectif, tout aussi important, est de s'assurer que les langages XML conçus pour l'exécution des business processes (p.ex. WSBPEL, Web Services Business Process Execution Language), peuvent être visualisées grâce à cette notation.

BP privé (ou interne)



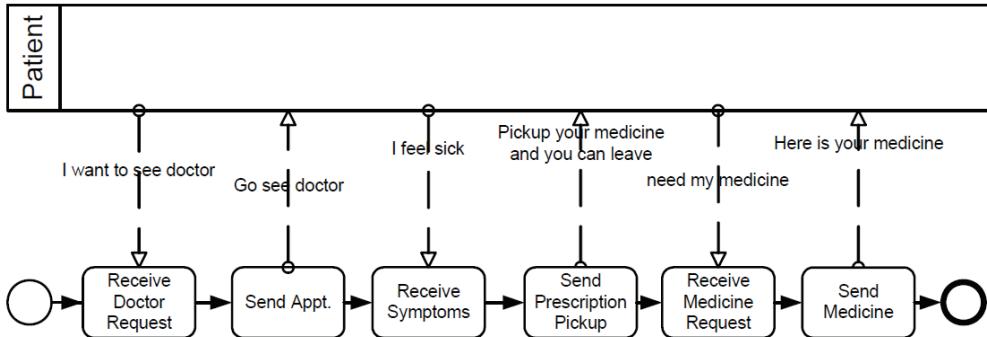
Les BP privés sont internes à une entreprise. Ils sont aussi appelés « workflow », « BPM process », ou « Orchestration of services ».

On en distingue deux types :

- « executable » : prêt à être exécuté (nombreux détails)
- « non-executable » : au niveau conceptuel, à des fins de documentation (peu de détails)

Lorsqu'on utilise la notation « pool & swimlane », alors on placera ce BP privé dans un seul pool. Autrement dit, le Process Flow doit être contenu exclusivement dans ce pool et ne peut pas sortir de ses limites. Par contre, les Message Flow peuvent sortir des limites du pool pour montrer les interactions existantes entre différents BP privés.

BP public

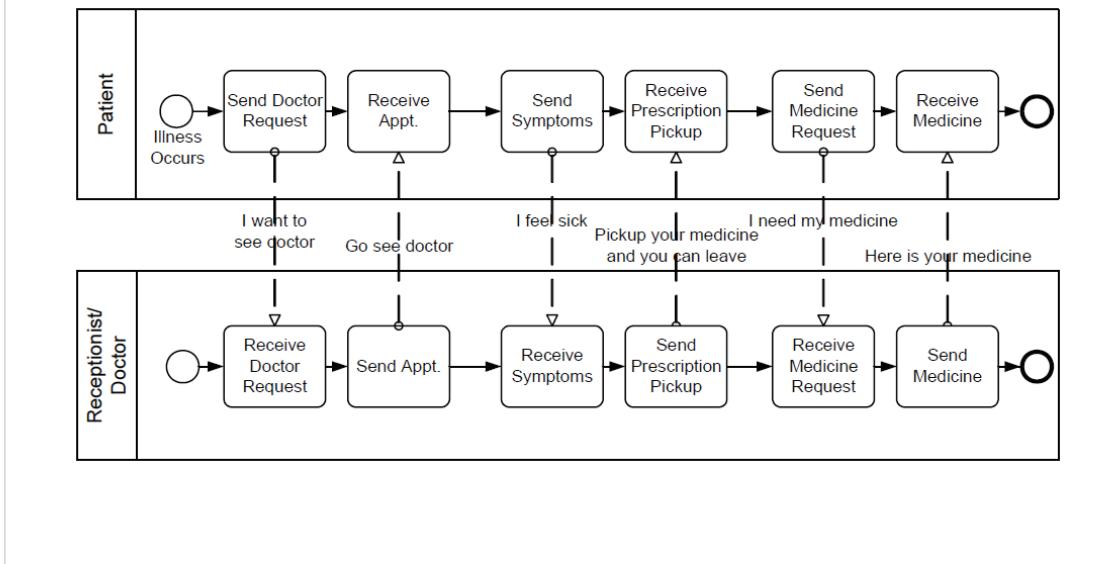


Terminologie

- OP : other participant (autre participant)
- MF : message flow (flux de message)

Un BP public représente une interaction entre un BP privé et un autre BP ou un OP. Ce sont uniquement les activités utilisées pour communiquer avec l'OP qui sont incluses dans le BP public. Les activités internes (donc, du BP privé) ne sont pas montrées dans le BP public. En conséquence, le BP public montre essentiellement les MF ordonnés.

BP collaboratif

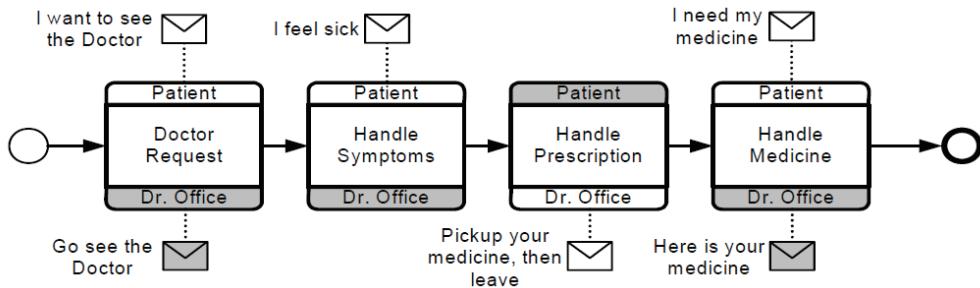


Une « collaboration » décrit les interactions entre plusieurs entités business, représentées chacune par un « pool ». Les MF connectent ainsi les pools. Les messages proprement dits peuvent aussi être montrés.

Les BP internes (exécutables) correspondants auront en général (beaucoup) plus d'activités et de détails que les BP publics.

Un pool peut rester vide, comme une boîte noire. Les chorégraphies peuvent être montrées entre les pools car elles coupent le flux de messages. Toutes les combinaisons de pools, BP et une chorégraphie sont autorisés dans une collaboration.

Chorégraphie



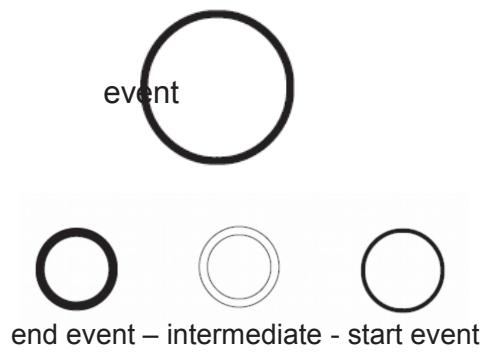
La « chorégraphie » est la définition d'un comportement attendu de participants agissant entre eux. Alors qu'un BP est défini au sein d'un pool, une chorégraphie l'est entre pools ou participants.

La chorégraphie ressemble visuellement à un BP privé, dans le sens où elle est composé d'un réseau d'activités, d'événements et de gateways. Cependant, elle en diffère en ce sens que les activités sont des interactions représentant un ou plusieurs échanges de messages impliquant plusieurs participants.

Éléments BPMN de base

- Objet de flux
 - Event
 - Activity
 - Gateway
- Données
 - Data Objects
 - Data Inputs
 - Data Outputs
 - Data Stores
- Connecteur
 - Sequence Flows
 - Message Flows
 - Associations
 - Data Associations
- Couloir
 - Pools
 - Lanes
- Artefact
 - Group
 - Text Annotation

Objets de flux



	"Catching"	"Throwing"	Non-interrupting
Message	✉️	✉️	✉️
Timer	⌚	⌚	⌚
Error	⚠️	⚠️	⚠️
Escalation	⚠️	⚠️	⚠️
Cancel	☒	☒	☒
Compensation	⤳	⤳	⤳
Conditional	☰	☰	☰
Link	🔗	🔗	🔗
Signal	⚠️	⚠️	⚠️
Terminate	ⓧ	ⓧ	ⓧ
Multiple	pentagon	pentagon	pentagon
Parallel Multiple	⊕	⊕	⊕

An **Event** is something that “happens” during the course of a Process or a Choreography. These Events affect the flow of the model and usually have a cause (trigger) or an impact (result). Events are circles with open centers to allow internal markers to differentiate different triggers or results. There are three types of Events, based on when they affect the flow: Start, Intermediate, and End.

- **Start Event** indicates where a particular Process or Choreography will start.
- **Intermediate Events** occur between a Start Event and an End Event. They will affect the flow of the Process or Choreography, but will not start or (directly) terminate the Process.
- **End Event** indicates where a Process or Choreography will end.

The **Start** and some **Intermediate Events** have “triggers” that define the cause for the Event. There are multiple ways that these events can be triggered.

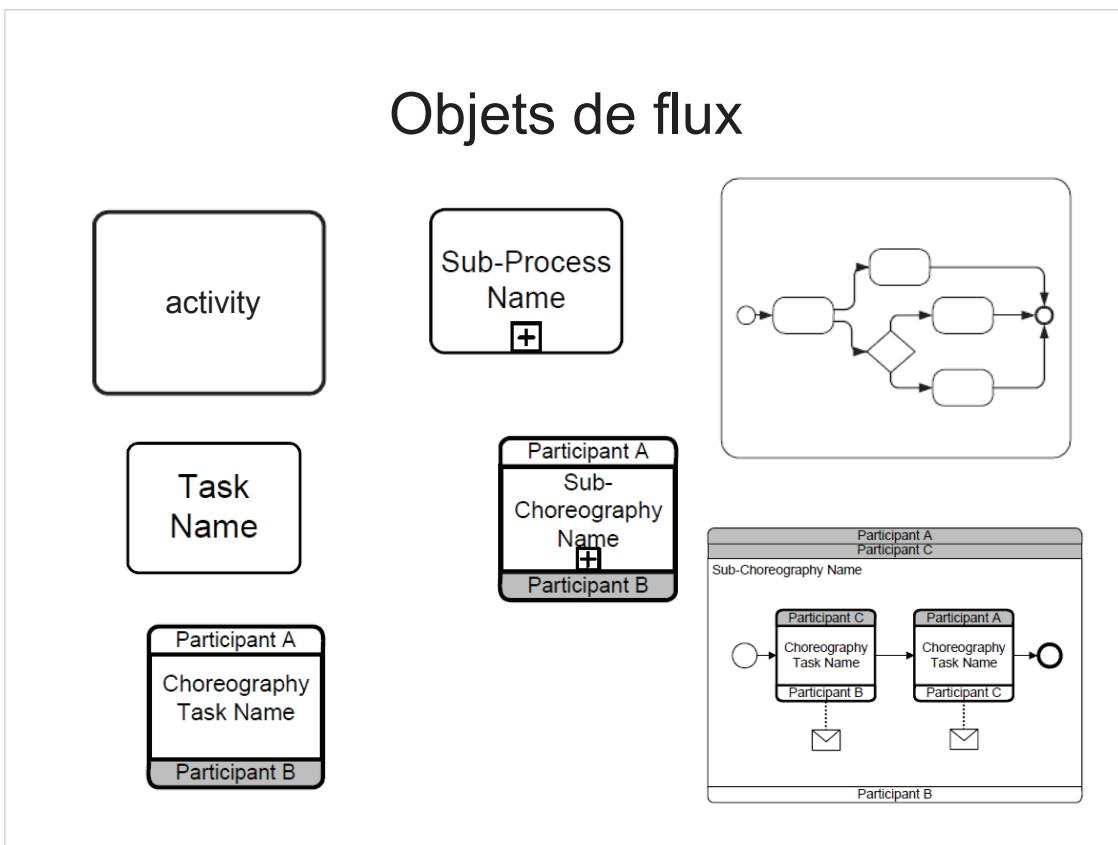
End Events MAY define a “result” that is a consequence of a Sequence Flow path ending.

Start Events can only react to (“catch”) a trigger.

End Events can only create (“throw”) a result. Intermediate Events can catch or throw triggers.

For the Events, triggers that catch, the markers are unfilled, and for triggers and results that throw, the markers are filled.

Objets de flux



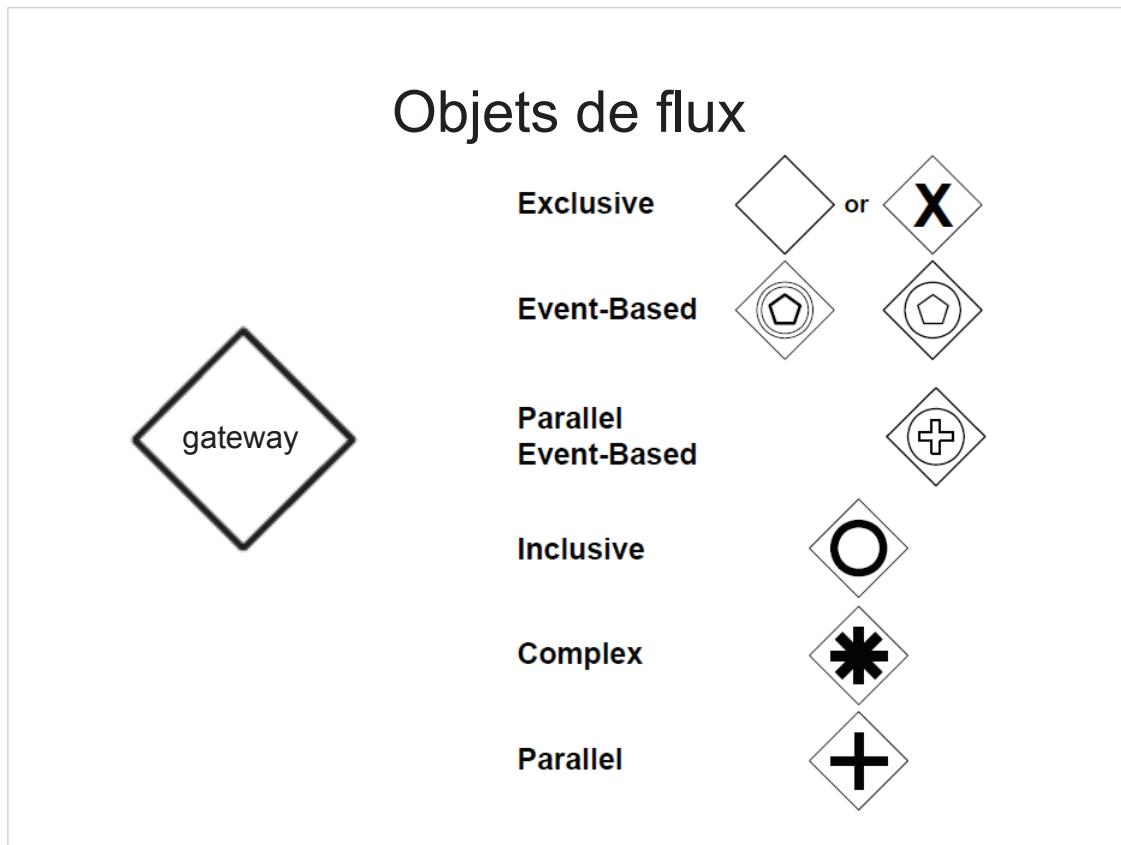
An **Activity** is a generic term for work that company performs in a Process. An Activity can be atomic or non-atomic (compound). The types of Activities that are a part of a Process Model are Sub-Process and Task, which are rounded rectangles. Activities are used in both standard Processes and in Choreographies.

A **Task** is an atomic Activity that is included within a Process. A Task is used when the work in the Process is not broken down to a finer level of Process detail.

A **Choreography Task** is an atomic Activity in a Choreography. It represents a set of one or more Message exchanges. Each Choreography Task involves two Participants. The name of the Choreography Task and each of the Participants are all displayed in the different bands that make up the shape's graphical notation. There are two or more Participant Bands and one Task Name Band.

A **Sub-Process** is a compound Activity that is included within a Process or Choreography. It is compound in that it can be broken down into a finer level of detail through a set of sub-Activities.

- **Collapsed Sub-Process** : The details are not visible in the Diagram. A “plus” sign in the lower-center of the shape indicates that the Activity is a Sub-Process and has a lower level of detail.
- **Expanded Sub-Process** : The boundary is expanded and the details are visible within its boundary. Note that Sequence Flows cannot cross the boundary of a Sub-Process.
- **Collapsed Sub-Choreography** : The details are not visible in the Diagram. A “plus” sign in the lower-center of the Task Name Band of the shape indicates that the Activity is a Sub-Process and has a lower level of detail.
- **Expanded Sub-Choreography** : The boundary is expanded and the details are visible within its boundary. Note that Sequence Flows cannot cross the boundary of a Sub-Choreography.



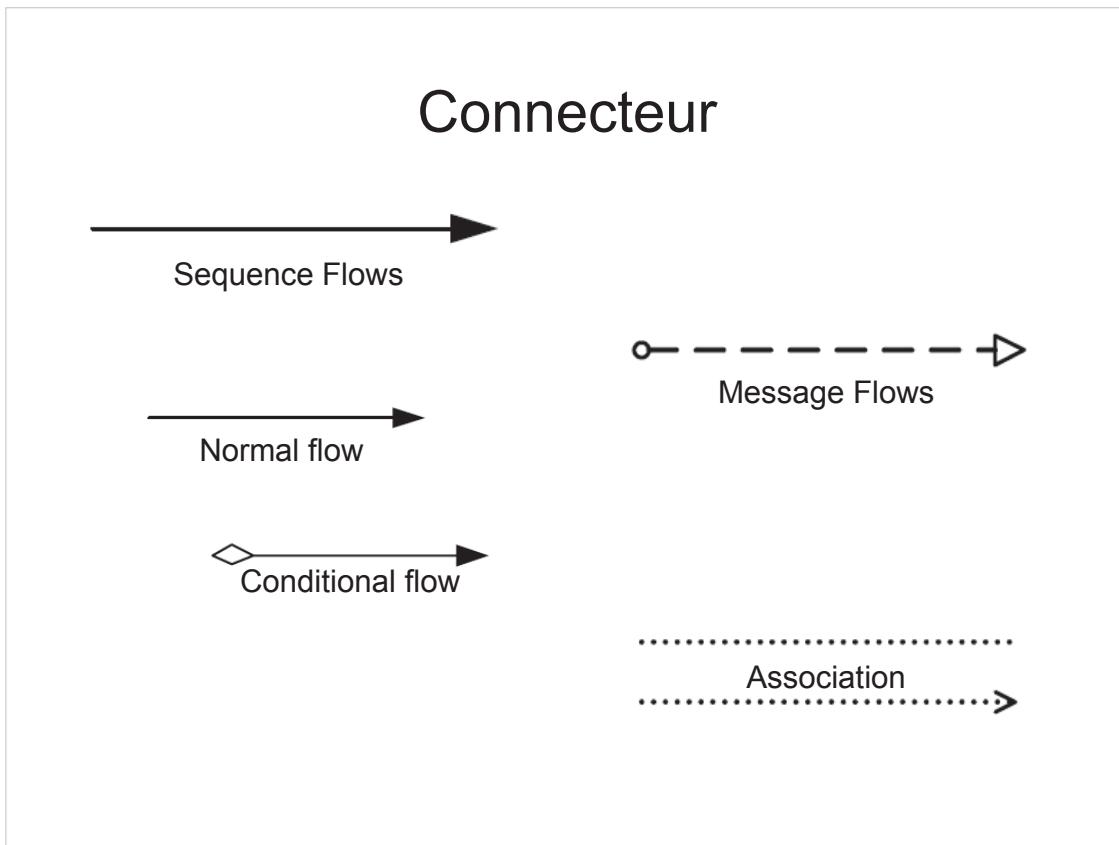
A **Gateway** is used to control the divergence and convergence of Sequence Flows in a Process and in a Choreography. Thus, it will determine branching, forking, merging, and joining of paths. Internal markers will indicate the type of behavior control.

Gateway Control Types

Icons within the diamond shape of the Gateway will indicate the type of flow control behavior. The types of control include :

- **Exclusive** decision and merging. Both **Exclusive** and **Event- Based** perform exclusive decisions and merging. Exclusive can be shown with or without the “X” marker.
- **Event-Based** and **Parallel Event-based** gateways can start a new instance of the Process.
- **Inclusive** Gateway decision and merging.
- **Complex** Gateway -- complex conditions and situations (e.g., 3 out of 5).
- **Parallel** Gateway forking and joining.

Each type of control affects both the incoming and outgoing flow.

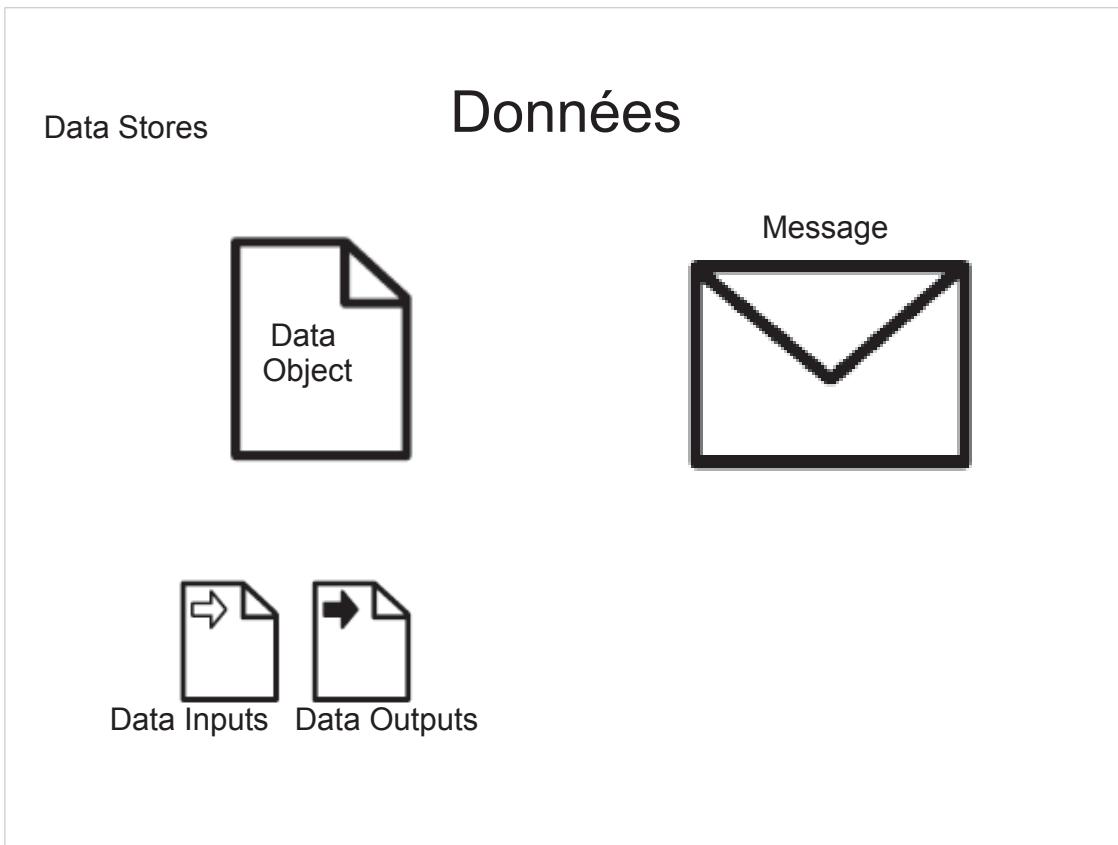


A **Sequence Flow** is used to show the order that Activities will be performed in a Process and in a Choreography.

- **Normal Flow** : Normal flow refers to paths of Sequence Flow that do not start from an Intermediate Event attached to the boundary of an Activity.
- **Conditional flow** : A Sequence Flow can have a condition Expression that are evaluated at runtime to determine whether or not the Sequence Flow will be used (i.e., will a token travel down the Sequence Flow). If the conditional flow is outgoing from an Activity, then the Sequence Flow will have a minidiamond at the beginning of the connector. If the conditional flow is outgoing from a Gateway, then the line will not have a mini-diamond (see figure in the row above).

A **Message Flow** is used to show the flow of Messages between two Participants that are prepared to send and receive them. In BPMN, two separate Pools in a Collaboration Diagram will represent the two Participants (e.g., PartnerEntities and/or PartnerRoles).

An **Association** is used to link information and Artifacts with BPMN graphical elements. Text Annotations and other Artifacts can be Associated with the graphical elements. An arrowhead on the Association indicates a direction of flow (e.g., data), when appropriate.

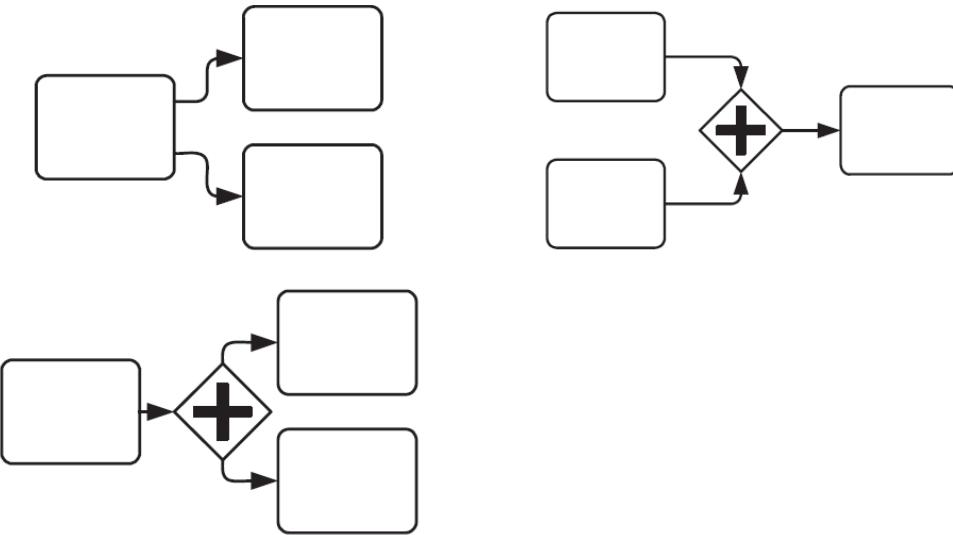


Data Objects provide information about what Activities require to be performed and/or what they Produce. Data Objects can represent a singular object or a collection of objects.

Data Input and **Data Output** provide the same information for Processes.

A **Message** is used to depict the contents of a communication between two Participants (as defined by a business PartnerRole or a business PartnerEntity).

Fork & Join

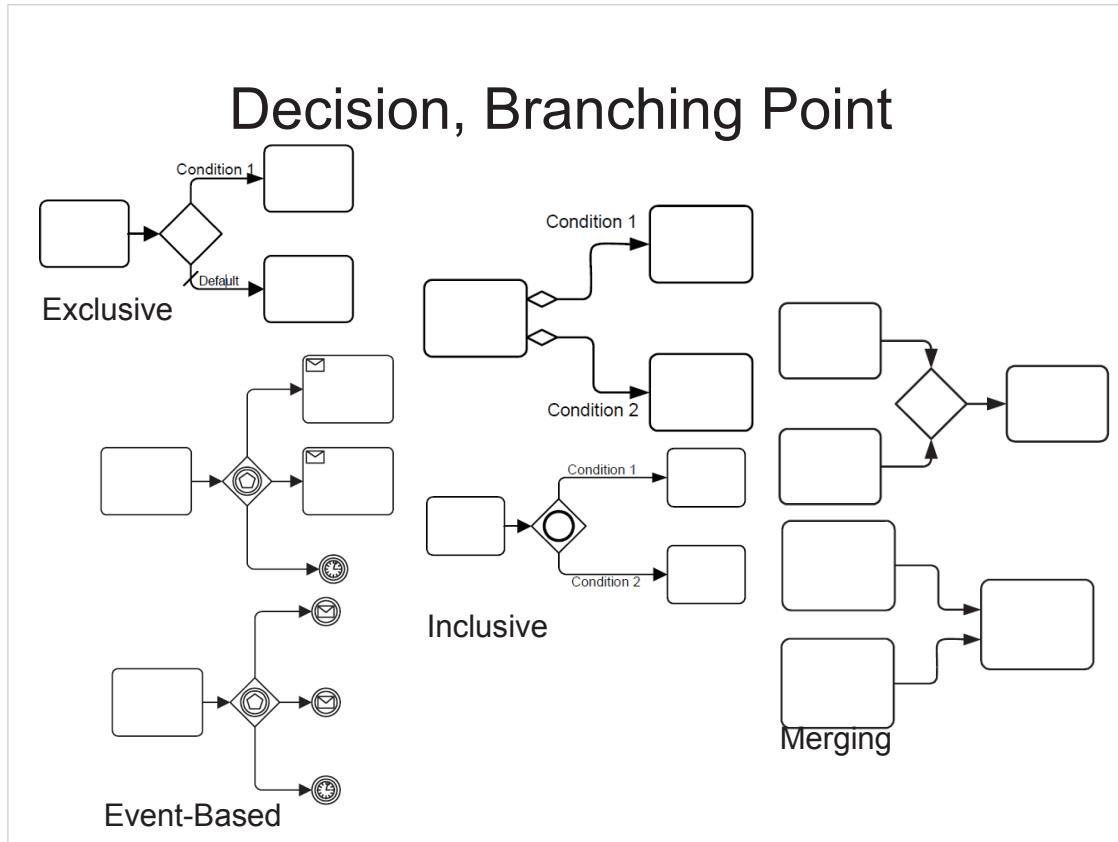


BPMN uses the term “**fork**” to refer to the dividing of a path into two or more parallel paths (also known as an AND-Split). It is a place in the Process where activities can be performed concurrently, rather than sequentially.

There are two options:

- Multiple Outgoing Sequence Flows can be used (see figure top-left). This represents “uncontrolled” flow is the preferred method for most situations.
- A Parallel Gateway can be used (see figure top-right). This will be used rarely, usually in combination with other Gateways.

BPMN uses the term “**join**” to refer to the combining of two or more parallel paths into one path (also known as an AND-Join or synchronization). A Parallel Gateway is used to show the joining of multiple Sequence Flows.



Decisions are Gateways within a Process or a Choreography where the flow of control can take one or more alternative paths.

Exclusive : This Decision represents a branching point where Alternatives are based on conditional Expressions contained within the outgoing Sequence Flows. Only one of the Alternatives will be chosen.

Event-Based : This Decision represents a branching point where Alternatives are based on an Event that occurs at that point in the Process or Choreography. The specific Event, usually the receipt of a Message, determines which of the paths will be taken. Only one of the Alternatives will be chosen. There are two options for receiving Messages:

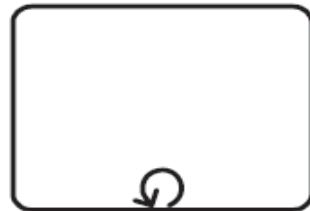
- Tasks of Type Receive can be used.
- Intermediate Events of Type Message can be used.

Inclusive : This Decision represents a branching point where Alternatives are based on conditional Expressions contained within the outgoing Sequence Flows. In some sense it is a grouping of related independent Binary (Yes/No) Decisions. Since each path is independent, all combinations of the paths MAY be taken, from zero to all. However, it should be designed so that at least one path is taken. A Default Condition could be used to ensure that at least one path is taken. There are two versions of this type of Decision:

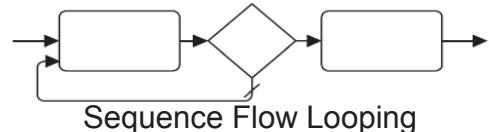
- The first uses a collection of conditional Sequence Flows, marked with minidiamonds (see top-right figure).
- The second uses an Inclusive Gateway (see bottom-right picture).

Merging : BPMN uses the term “merge” to refer to the exclusive combining of two or more paths into one path (also known as an OR-Join). A Merging Exclusive Gateway is used to show the merging of multiple Sequence Flows (see upper figure to the right). If all the incoming flow is alternative, then a Gateway is not needed. That is, uncontrolled flow provides the same behavior (see lower figure to the right).

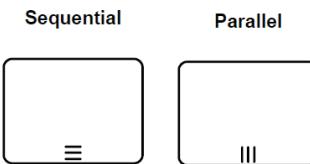
Looping & Multiple Instances



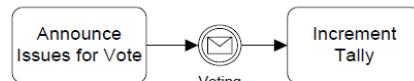
Activity Looping



Sequence Flow Looping



Multiple Instances



Process Break

Mechanisms for looping within a Process

Activity Looping : The attributes of Tasks and Sub-Processes will determine if they are repeated or performed once. There are two types of loops: Standard and Multi-Instance. A small looping indicator will be displayed at the bottom-center of the activity.

Sequence Flow Looping : Loops can be created by connecting a Sequence Flow to an “upstream” object. An object is considered to be upstream if that object has an outgoing Sequence Flow that leads to a series of other Sequence Flows, the last of which is an incoming Sequence Flow for the original object.

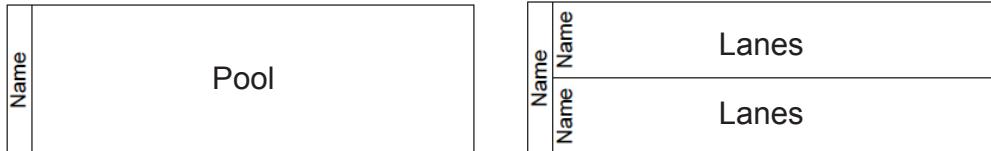
Multiple Instances

The attributes of Tasks and Sub-Processes will determine if they are repeated or performed once. A set of three horizontal lines will be displayed at the bottom-center of the activity for sequential Multi-Instances. A set of three vertical lines will be displayed at the bottom-center of the activity for sequential Multi-Instances.

Process Break

Something out of the control of the process makes the process pause. A Process Break is a location in the Process that shows where an expected delay will occur within a Process. An Intermediate Event is used to show the actual behavior. In addition, a Process Break Artifact, as designed by a modeler or modeling tool, can be associated with the Event to highlight the location of the delay within the flow.

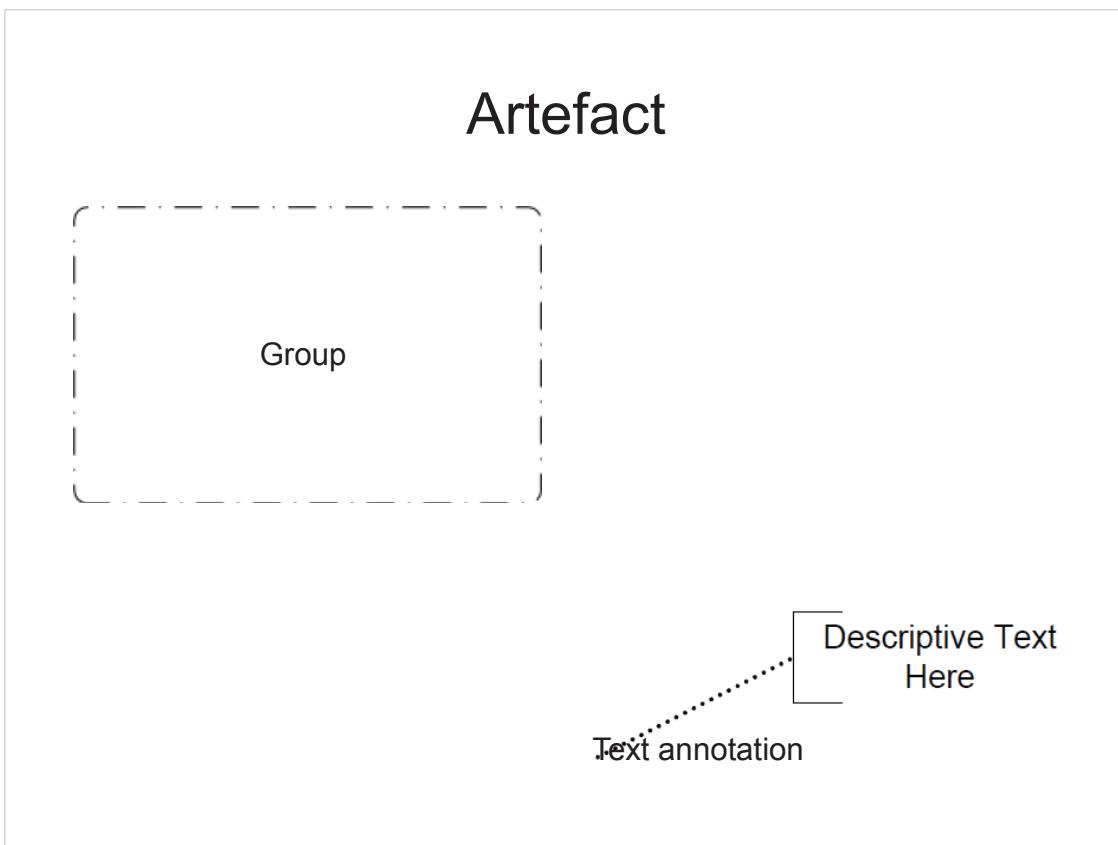
Couloir



A **Pool** is the graphical representation of a Participant in a Collaboration. It also acts as a “swimlane” and a graphical container for partitioning a set of Activities from other Pools, usually in the context of B2B situations. A Pool MAY have internal details, in the form of the Process that will be executed. Or a Pool MAY have no internal details, i.e., it can be a "black box."

A **Lane** is a sub-partition within a Process, sometimes within a Pool, and will extend the entire length of the Process, either vertically or horizontally. Lanes are used to organize and categorize Activities.

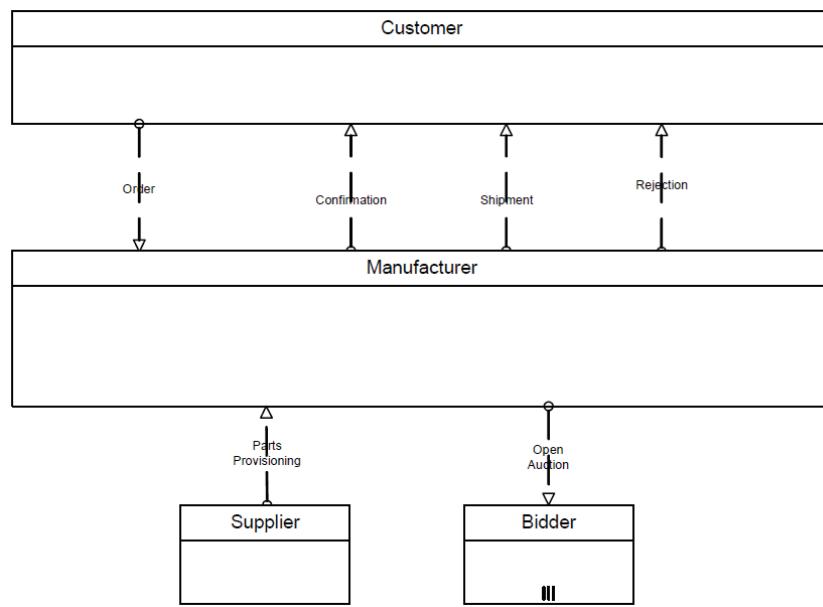
Artifact



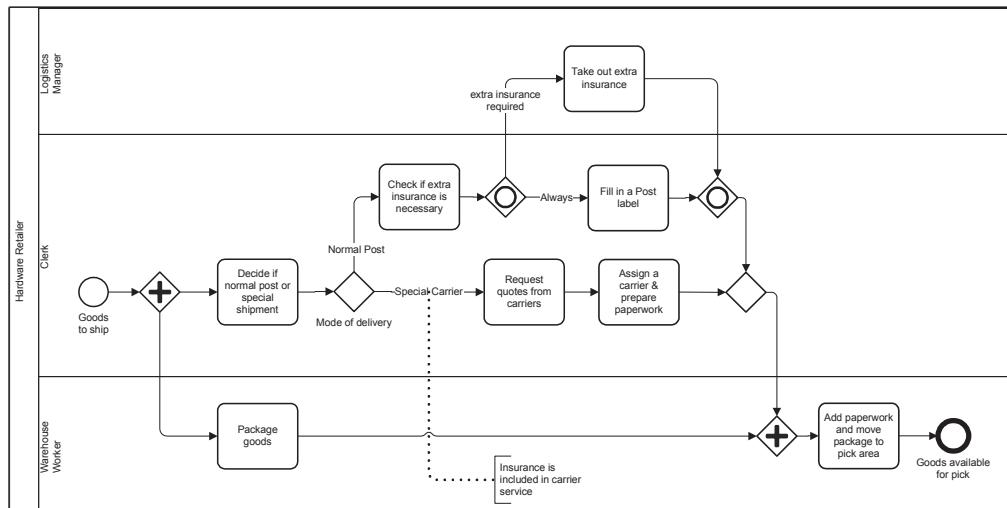
A **Group** is a grouping of graphical elements that are within the same Category. This type of grouping does not affect the Sequence Flows within the Group. The Category name appears on the diagram as the group label. Categories can be used for documentation or analysis purposes. Groups are one way in which Categories of objects can be visually displayed on the diagram.

Text Annotations, attached with an Association, are a mechanism for a modeler to provide additional text information for the reader of a BPMN Diagram.

Exemple : Collaboration diagram with black-box Pools



Exemple : Shipment Process of a Hardware Retailer



In this figure, you can find the preparing steps a hardware retailer has to fulfill before the ordered goods can actually be shipped to the customer.

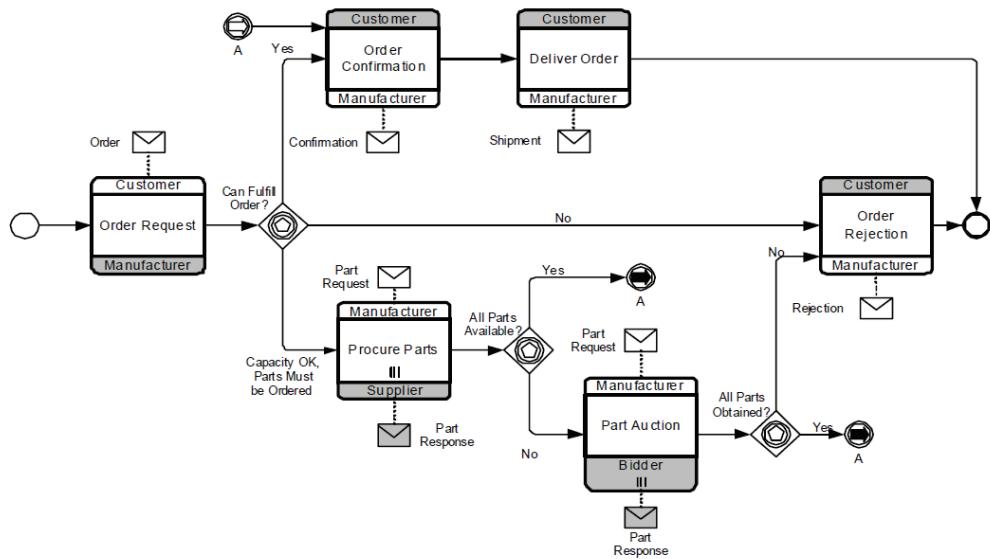
We used only one pool and different lanes for the people involved in this process, which automatically means that we blank out the communication between those people: We just assume that they are communicating with each other somehow.

The plain start event “goods to ship” indicates that this preparation should be done now. Right after the instantiation of the process, there are two things done in parallel, as the parallel gateway indicates: While the clerk has to decide whether this is a normal postal or a special shipment (we do not define the criteria), the warehouse worker can already start packaging the goods. This clerk's task, which is followed by the exclusive gateway “mode of delivery”, is a good example for clarifying the recommended usage of a gateway: The gateway is not responsible for the decision whether this is a special or a postal shipment. Instead, this decision is undertaken in the activity before. The gateway only works as a router, which is based on the result of the previous task, and provides alternative paths. A task represents an actual unit of work, while a gateway is only routing the sequence flow.

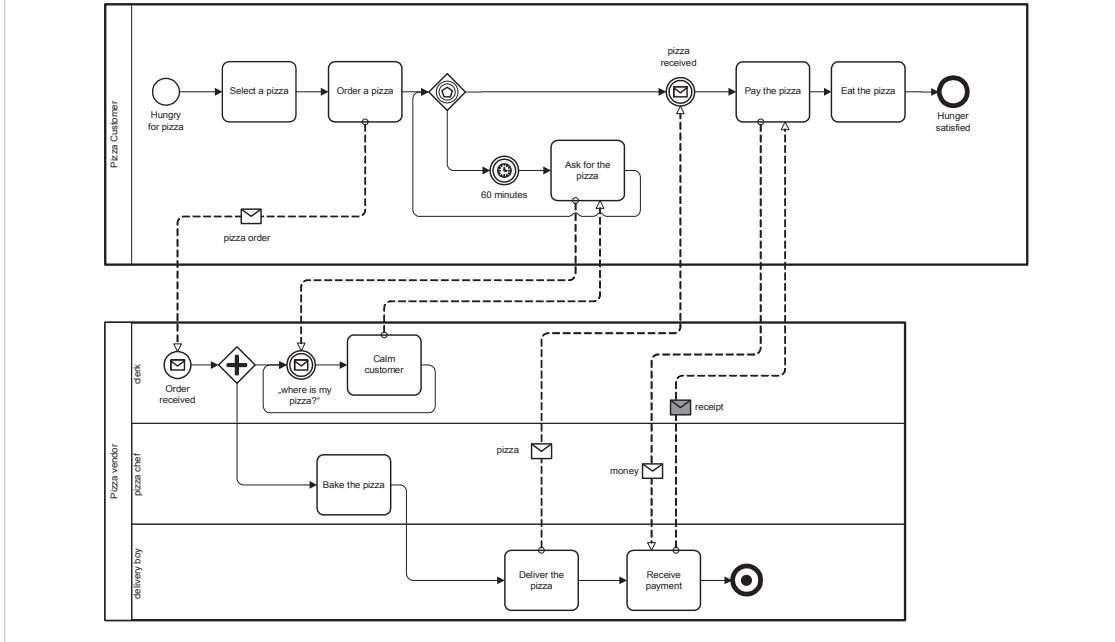
This gateway is called “exclusive”, because only one of the following two branches can be traversed: If we need a special shipment, the clerk requests quotes from different carriers, then assigns a carrier and prepares the paperwork. But if a normal post shipment is fine, the clerk needs to check if an extra insurance is necessary. If that extra insurance is required, the logistics manager has to take out that insurance. In any case, the clerk has to fill in a postal label for the shipment. For this scenario, the shown inclusive gateway is helpful, because we can show that one branch is always taken, while the other one only if the extra insurance is required, but IF it is taken, this can happen in parallel to the first branch. Because of this parallelism, we need the synchronizing inclusive gateway right behind “Fill in a Post label” and “Take out extra insurance”. In this scenario, the inclusive gateway will always wait for “Fill in a Post label” to be completed, because that is always started. If an extra insurance was required, the inclusive gateway will also wait for “Take out extra insurance” to be finished.

Furthermore, we also need the synchronizing parallel gateway before the last task “add paperwork and move package to pick area”, because we want to make sure that everything has been fulfilled before the last task is executed.

Exemple : standalone Choreography diagram



Exemple : The Pizza Collaboration



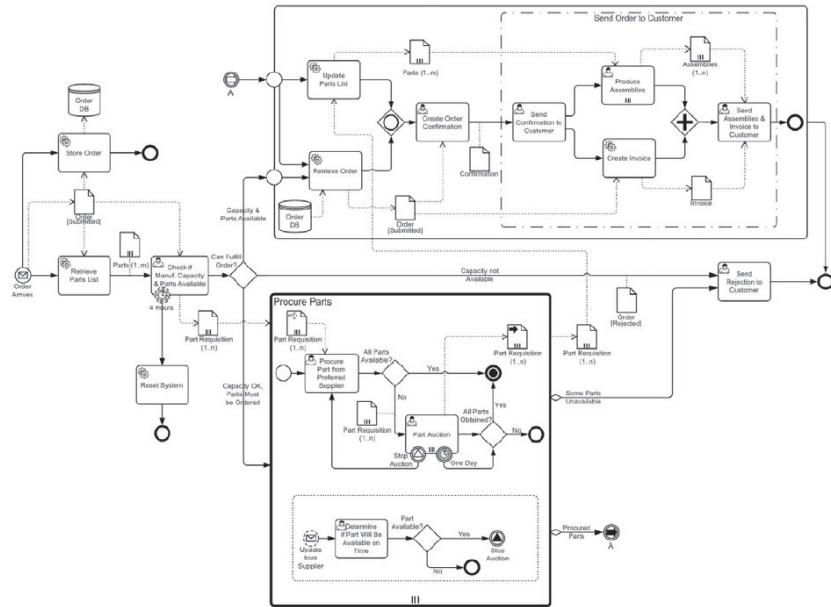
This example is about Business-To-Business-Collaboration. Because we want to model the interaction between a pizza customer and the vendor explicitly, we have classified them as “participants”, therefore providing them with dedicated pools. Please note that there is no default semantics in this type of modeling, which means you can model collaboration diagrams to show the interaction between business partners, but also zoom into one company, modeling the interaction between different departments, teams or even single workers and software systems in collaboration diagrams. It is totally up to the purpose of the model and therefore a decision the modeler has to make, whether a collaboration diagram with different pools is useful, or whether one should stick to one pool with different lanes, as shown in the previous chapter.

If we step through the diagram, we should start with the pizza customer, who has noticed her stomach growling. The customer therefore selects a pizza and orders it. After that, the customer waits for the pizza to be delivered. The event based gateway after the task “order a pizza” indicates that the customer actually waits for two different events that could happen next: Either the pizza is delivered, as indicated with the following message event, or there is no delivery for 60 minutes, i.e., after one hour the customer skips waiting and calls the vendor, asking for the pizza. We now assume that the clerk promises the pizza to be delivered soon, and the customers waits for the pizza again, asking again after the next 60 minutes, and so on. Let's have a closer look at the vendor process now. It is triggered by the order of the customer, as shown with the message start event and the message flow going from “order a pizza” to that event. After baking the pizza, the delivery boy will deliver the pizza and receive the payment, which includes giving a receipt to the customer.

In this example, we use message objects not only for informational objects, as the pizza order, but also for physical objects, like the pizza or the money. We can do this, because those physical objects actually act as informational objects inherently: When the pizza arrives at the customer's door, she will recognize this arrival and therefore know that the pizza has arrived, which is exactly the purpose of the accordant message event in the customer's pool. Of course, we can only use the model in that way because this example is not meant to be executed by a process engine.

Exemple :

standalone Process diagram



Éléments de gestion de projets

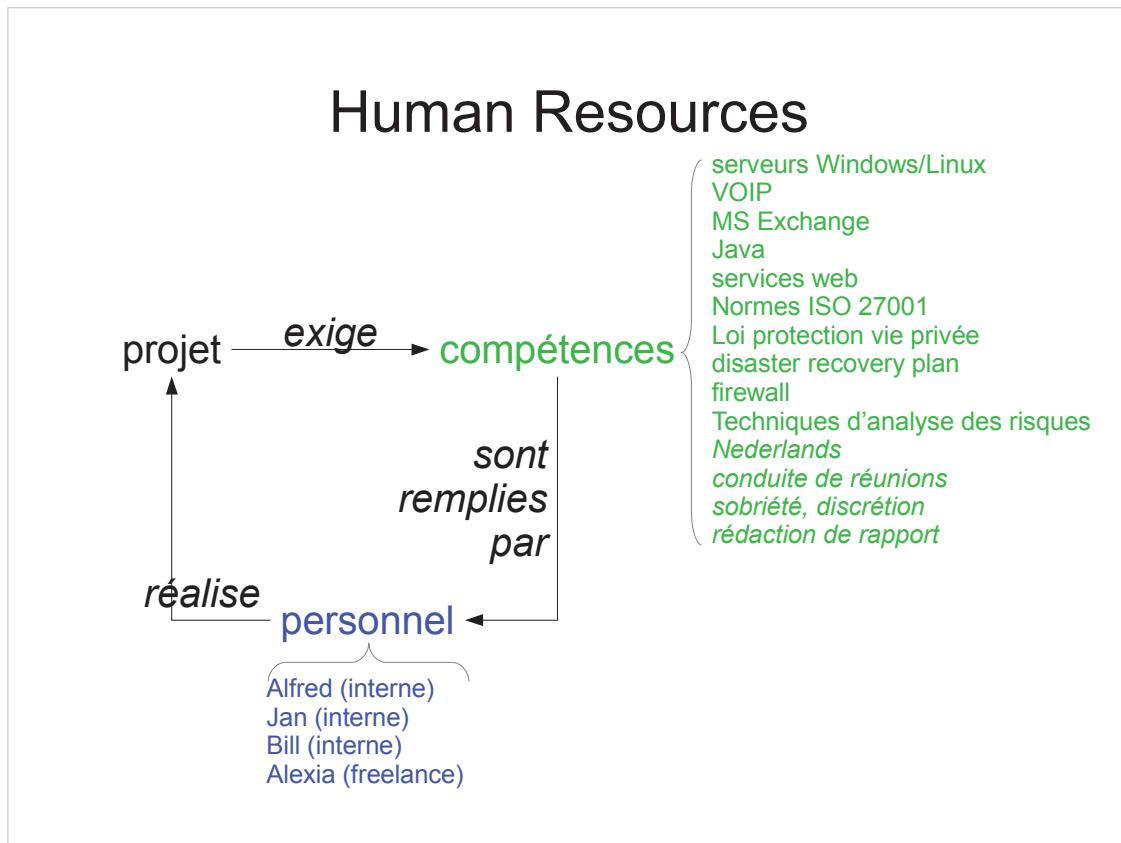
3. Initialisation

- 1) Spécifications détaillées
- 2) Ressources humaines**
- 3) Découpage
- 4) Planification du travail
- 5) Gestion des risques
- 6) Planification de l'acceptance
- 7) Budgétisation
- 8) Communication interne

Human Resources

- La réussite d'un projet passe par une organisation efficace de l'équipe projet.
- Acteurs principaux : MOA MOE CP
- Acteurs IT :
 - Utilisateurs
 - Utilisateur final : manipulation de l'outil livré
 - Administrateur : configuration
 - Décideur : approbation des fonctionnalités
 - Équipe de projet
 - Consultance
 - Développement (software)
 - Equipement (hardware)
 - Support
- A chaque profil, sa tarification propre !

Development	Application Architect
	Analyst
	Senior Analyst-Programmer
	Analyst-Programmer
	Programmer
	Interface Designer
	Technical Writer
Consultance	Senior Consultant, Senior Architect
	Consultant, Architect
	Tester
Support	Senior User Assistance (seconde ligne)
	User Assistance (première ligne)
	Webmaster
	Coordinator



L'approche présentée ici revêt un caractère absolu et n'est pas toujours applicable ainsi dans la vie réelle, surtout quand l'organisation est de petite taille. Et même quand l'organisation est de taille respectable, les contraintes font que souvent « on fait avec ce qu'on a » quand le personnel est affecté sur d'autres projets.

Chaque projet exige des compétences particulières, que l'on listera en établissant les fonctionnalités détaillées (cf texte en vert sur le slide).

C'est la responsabilité du MOE et/ou du chef de projet de former l'équipe en fonction des compétences recherchées. Le personnel peut être soit interne, idéalement quand « il fait banquette », soit externe (freelance) mais dans ce dernier cas les marges bénéficiaires sont plus réduites.

En principe, quand le personnel a été affecté sur un projet, il y reste juste à la fin. Un changement de personnel en cours de projet est toujours une épreuve lourde et coûteuse.

Human Resources

ACTEURS	Informer la hiérarchie des problèmes	Transmettre les consignes	Faire le compte-rendu d'activité	Régler le plan de découpe	Réplier le cahier de consignes	Lire le cahier de consignes	Intervenir sur les pannes mineures	Respecter l'hygiène et la sécurité	Respecter la propriété (environnement)	Régler les détecteurs de métaux	Régler les peseuses tritueuses	Démarrer les lignes de fabrication	Contrôler la fabrication	Préparer les changements de fabrication	Contrôler le produit fini	Suivre le planning
P1	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
P2	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
P3	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
P4	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
P5	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Responsable	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

<input checked="" type="checkbox"/>	Bonne connaissance théorique et pratique courante maîtrisée	<input type="checkbox"/>	Ni connaissance théorique ni pratique
<input checked="" type="checkbox"/>	Connaissance des principes avec pratique occasionnelle	<input checked="" type="checkbox"/>	A réaliser
<input type="checkbox"/>	Connaissance des principes sans pratique	<input type="checkbox"/>	

Le tableau ci-dessus est constitué de

- les membres du personnel impliqué dans les projets de l'entreprise : P1 P2 P3 ...
- les compétences ou activités requises par le projet : cf entête des colonnes
- le niveau atteint dans chaque compétence, par chaque membre du personnel : les carrés ou cercles en noir et blanc.

Cet outil de management des ressources humaines a plusieurs utilités :

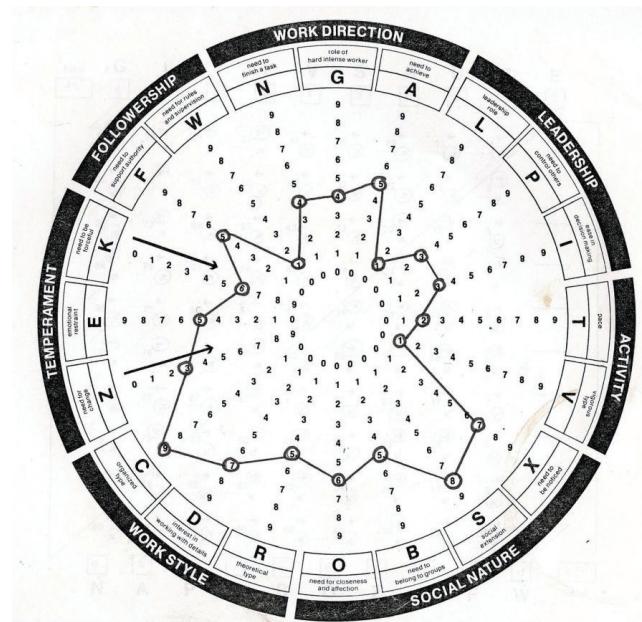
- Affecter le personnel le plus compétent aux tâches du projet
- Préparer les plans de formation pour le personnel en perte de compétences
- S'assurer que l'organisation (l'entreprise ou le département) possède suffisamment de compétences mais aussi de « réserves de compétence » (back-up). Une bonne pratique est d'avoir, pour chaque compétence, un équilibre entre les « maîtres », les « experts » et les « apprentis », afin de pallier aux absences planifiées ou non : affectation sur un autre projet, vacances, maladie, passage sous l'autobus :-), ...

<http://www.iseor.com/>

Human Resources

Test PAPI

Ce qui vous décrit le mieux ?
 J'aime raconter mes exploits
 Je me décide vite

Terminologie

PAPI : Personality and Preference Inventory (Inventaire des Perceptions et Préférences)

Une compétence n'est pas seulement technique et n'est pas composée que de « savoir » et de « savoir-faire ». De plus en plus, certainement pour des postes à responsabilités mais aussi dans pour des postes de junior, les employeurs attachent de l'importance au « savoir-être ». Le savoir-être regroupe les motivations, le comportement, certaines valeurs, les points d'intérêt, les capacités émotionnelles, etc. d'une personne.

Exemple pour bien comprendre les différences entre les trois types de compétences :

- Savoir : connaître les fonctionnalités de l'outil MS Powerpoint ou OO Impress
- Savoir-faire : réaliser dans les temps et avec soin un fichier PPT ou ODP de support à une présentation de 20 minutes
- Savoir-être : faire la présentation de 20 min et la terminer par une standing ovation :-)

Il est donc intéressant de définir son savoir-être. Des tests très sérieux ont été mis au point à cet effet, e.a. le test PAPI. Celui-ci dresse un radar de ses perceptions et préférences émotionnelles. Précisons toutefois que PAPI ne traite que des préférences des personnes et n'évalue pas leurs compétences ; si PAPI prête à quelqu'un p.ex. une forte propension au leadership, il ne dit pas qu'elle est un bon leader mais seulement que ce point l'attire.

Un chef de projet peut s'aider des tests PAPI des membres de son équipe pour constituer une équipe équilibrée. Il est bon d'avoir dans une équipe des gens aux perceptions et préférences complémentaires et pas trop homogènes ; en effet, comment p.ex. se dérouleront les réunions si trop de participants sont des « détaillistes chevronnés » et qu'il manque d'« esprit synthétique » ... ?

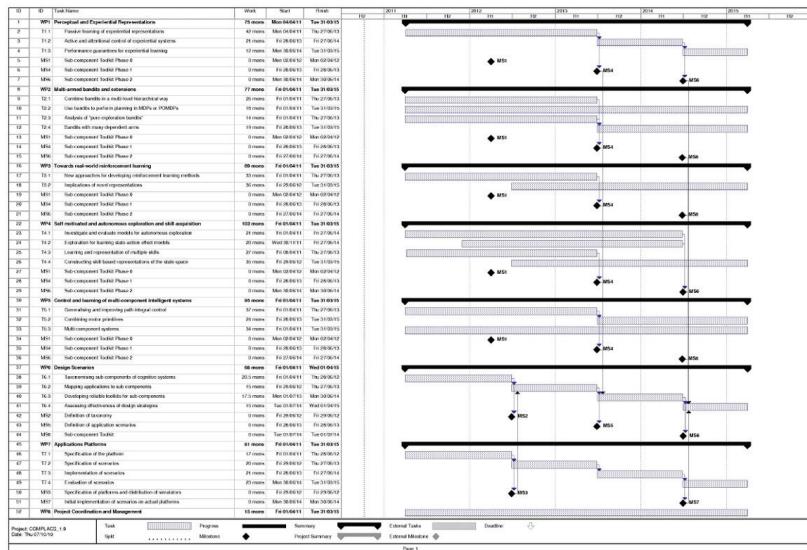
<http://www.cubiks.fr/>

Éléments de gestion de projets

3. Initialisation

- 1) Spécifications détaillées
- 2) Ressources humaines
- 3) Découpage**
- 4) Planification du travail
- 5) Gestion des risques
- 6) Planification de l'acceptance
- 7) Budgétisation
- 8) Communication interne

Workpackages, Tasks, Milestones, Deliverables



Définition des Workpackages

Le projet est en général divisé « workpackages ». Chacun possède son propre périmètre et a des interdépendances claires et limitées avec les autres. Si deux workpackages présentent trop d'interdépendance, alors il vaut mieux n'en faire qu'un seul workpackage, ou aborder la problématique sous un autre point de vue.

Définition des tâches

Une tâche est une action à mener pour aboutir à un résultat. A chaque tâche définie, il faut associer

- Un objectif précis et mesurable
- Des ressources humaines, matérielles et financières adaptées
- Une charge de travail exprimée en nombre de journées-homme
- Une durée ainsi qu'une date de début et une date de fin

Une tâche doit être assez courte (max 15 jours). Dans le cadre du planning, les tâches sont reliées entre elles par des relations de dépendance.

Définition des Milestones (jalons)

Les jalons d'un projet se définissent comme

- Des événements clé d'un projet, montrant une certaine progression du projet
- Des dates importantes de réalisation d'un projet
- Une réalisation concrète (production de livrables)

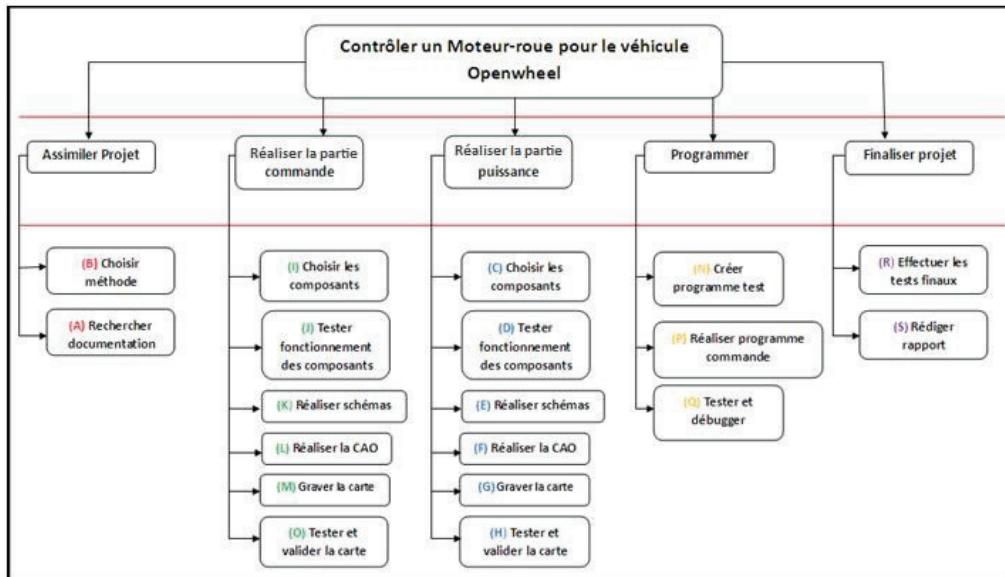
Dans le cadre du planning, les jalons limitent le début et la fin de chaque phase et servent de point de synchronisation. Sur les diagrammes de GANTT, les jalons sont représentés par des losanges.

Définition des Deliverables (livrables)

Un livrable est tout résultat, document, mesure, tangible ou vérifiable, qui résulte de l'achèvement d'une partie de projet ou du projet.

Exemples de livrables : spécifications détaillées, documentation technique, site d'e-commerce

Work Breakdown Structure (WBS)



C'est l'activité qui consiste à déterminer les workpackages et les tâches du projet, à estimer leurs charges et à déterminer les profils nécessaires à leur réalisation.

La WBS répond aux questions : Que doit-on faire? Comment doit-on s'y prendre?

La WBS est la structure hiérarchique des tâches du projet. Sa conception passe par :

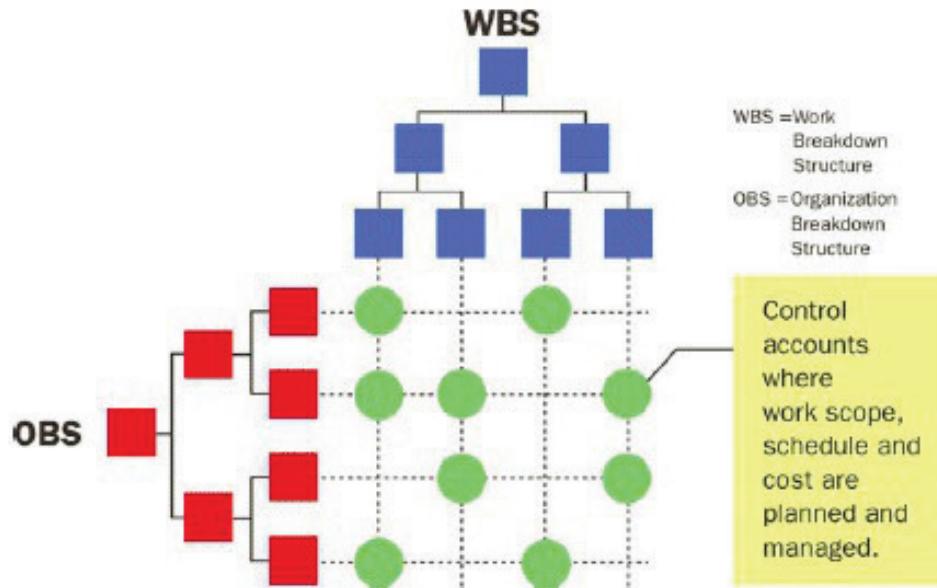
- L'établissement des deliverables les plus importants ; chaque workpackage doit fournir au moins un deliverable.
- La division (si nécessaire) de ces deliverables en sous-deliverables
- Pour chaque deliverable et sous-deliverable, le listage des tâches nécessaires à sa réalisation
- La division (si nécessaire) de ces tâches en sous-tâches

Au final, chaque workpackage doit être **de complexité et d'interfaçage maîtrisables**. En pratique, une tâche, et parfois un workpackage entier, se définit prioritairement en fonction des compétences et du personnel disponibles.

Un workpackage est composé d'un ensemble de tâches dont il faut arriver à déterminer – si possible chronologiquement :

- **Task** : Ce qui doit être fait
- **Resource** : Qui doit les réaliser
- **Deliverable** : Quels résultats doivent être présentés
- **Milestones** : Quand les valider

Organization Breakdown Structure (OBS)

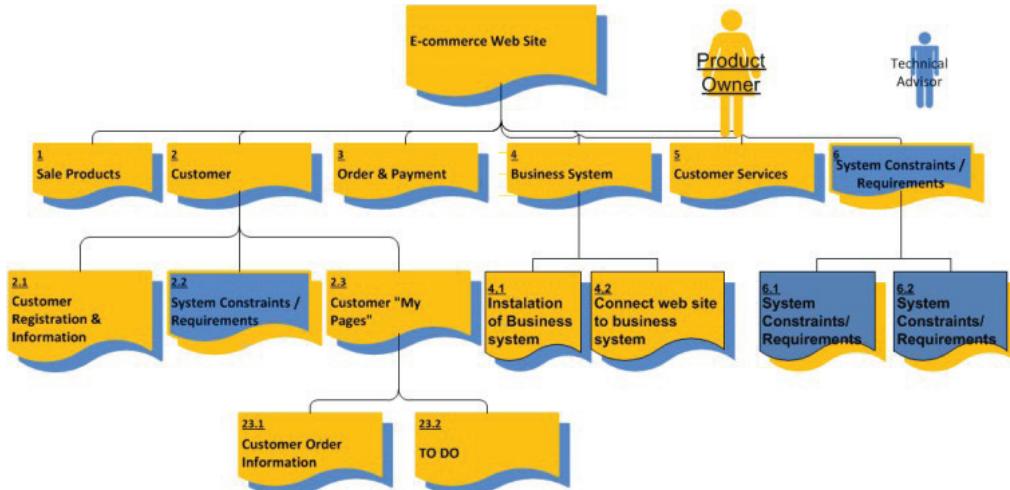


L'OBS reprend le WBS tout en faisant apparaître les noms des personnes responsables de la production des différents éléments.

On peut parler d'un mix du WBS avec l'organigramme fonctionnel.

L'avantage est de pouvoir préciser qui fait quoi et donc de préparer la planification.

Product Breakdown Structure (PBS)



Le PBS répond à la question : Comment est constitué le produit final ?

Le PBS consiste en un découpage structurel du produit final en ses différents composants élémentaires - les plus autonomes possibles - tels qu'imaginés en début de projet. Le PBS représente également les liens de composition entre les divers constituants d'un produit complexe.

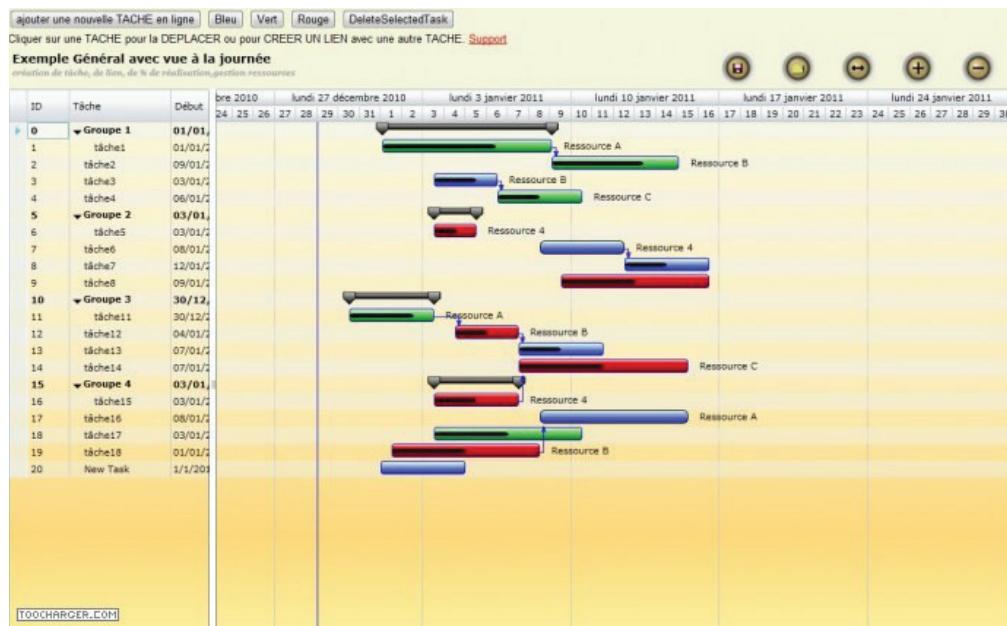
Ce n'est pas forcément une bonne idée d'appliquer le PBS quand, notamment, tous les développeurs sont impliqués dans tous les composants.

Éléments de gestion de projets

3. Initialisation

- 1) Spécifications détaillées
- 2) Ressources humaines
- 3) Découpage
- 4) Planification du travail**
- 5) Gestion des risques
- 6) Planification de l'acceptance
- 7) Budgétisation
- 8) Communication interne

Planification : planning



L'ordonnancement est l'élaboration du plan d'action permettant de déterminer les séquencements ou au contraire les parallélismes possibles entre l'exécution des tâches précédemment identifiées.

Dans certains projets, une marge de flexibilité peut être aménagée par le chef de projet pour l'ordonnancement des tâches, c'est à dire que le chef de projet peut prévoir plusieurs scénarios possibles concernant l'ordonnancement des tâches. En fonction de l'évolution du projet, un scénario d'ordonnancement des tâches peut être privilégié par rapport à un autre scénario.

Pour procéder à l'ordonnancement des tâches, il faut, pour chaque tâche élémentaire, lister les tâches antérieures, au vu des informations collectées sur le terrain et sélectionner les seules tâches immédiatement antérieures.

L'ensemble des tâches ordonnancées forme le **planning**. C'est l'indispensable outil de la planification. A la lecture du planning, on doit pouvoir à tout moment du projet

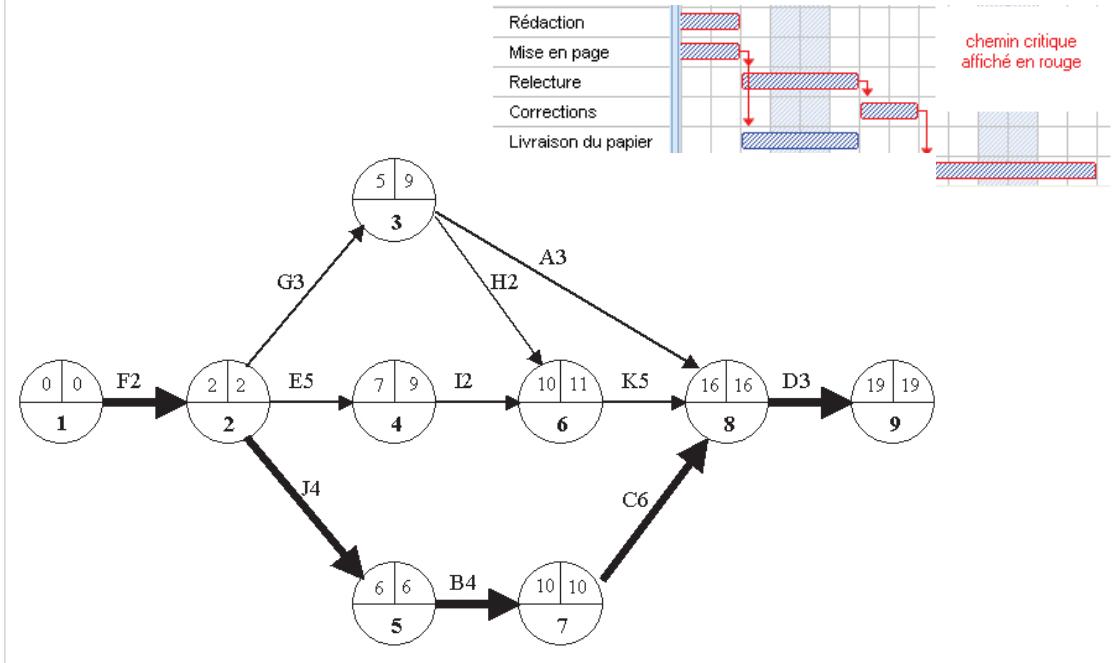
- déterminer si les objectifs sont réalisés ou dépassés
- suivre et communiquer l'avancement du projet
- affecter les ressources aux tâches

Cas d'entreprise

Prendons l'exemple d'un projet informatique. Supposons qu'une entreprise souhaite implémenter un ERP. Ce type de projet comporte plusieurs grandes étapes :

- Étude préalable détaillée (définition du périmètre, cahier des charges fonctionnel ...)
- Dossier de Paramétrage
- Réalisation du paramétrage et/ou Programmation
- Conception des Jeux d'essai pour préparer la recette (réception) de l'application/du module
- Recette (Réalisation des tests informatiques)
- Rédaction des Manuels utilisateurs
- Mise en production

Planification : chemin critique



Le Planning

Dates au plus tôt et au plus tard

Pour bâtir un planning, il faut associer à chaque tâche les dates « au plus tôt » (« début au plus tôt ») et « fin au plus tôt » de l'exécution de la tâche) et les dates « au plus tard » (« début au plus tard » et « fin au plus tard » de l'exécution de la tâche). La durée de la tâche est le temps ouvré qui s'écoule entre le début et la fin de la tâche.

Importance du chemin critique et des marges

Le **chemin critique** correspond à la séquence de tâches qui détermine la durée totale du projet. Ce chemin est continu depuis le début jusqu'à la fin du projet. Tout retard affectant une tâche du chemin critique est intégralement répercute sur la durée du projet et donc sa date de fin. Une **tâche critique** est une tâche du chemin critique. Toute modification sur la durée d'une de ces tâches impacte d'autant plus la durée totale du projet.

La **marge** est la possibilité qu'à une tâche d'être retardée sans impacter le délai du projet. Les tâches qui sont sur le chemin critique ont une marge nulle.

La **marge totale** (MT) est égale à la différence entre le début au plus tard de la tâche suivante la plus contraignante et la fin au plus tôt de la tâche elle-même. C'est aussi la différence entre les dates au plus tard et les dates au plus tôt de la tâche elle-même.

La **marge libre** (ML) est égale à la différence entre la date de début au plus tôt du successeur le plus précoce, et la date de fin au plus tôt de la tâche elle-même.

Planification : charges

INVESTISSEMENT MATERIEL		Montant euros TTC
Informatique		1 500
Total investissement matériel		1 500
DEPENSES EXTERNES		Montant euros TTC
Consultants programme SI-PME e-Commerce		47 300
Total dépenses externes		47 300
VALORISATIONS		Montant euros TTC
Organismes Consulaires : mise à disposition des locaux et des frais associés (chauffage, électricité, téléphone fixe...)		16 000
Total valorisations		16 000
SALAIRES ET FRAIS DE DEPLACEMENT		Montant euros TTC
Salaire chargé - Directeur		71 500
Salaire chargé - Conseiller TIC		40 000
Salaire chargé - Assistante		22 000
Frais de déplacement		18 000
Total salaires et frais de déplacement		151 500

CHARGES DE FONCTIONNEMENT		Montant euros TTC
Maintenance Informatique		300
Fournitures		1 200
Participation annuelle TIC Rhône-Alpes		300
Outil base de données des contacts entreprises (Ines)		700
Fonds documentaires - abonnements		300
Frais postaux + photocopies...		1 500
Téléphone mobile + abonnement FT ADSL + connexion...		2 500
Communication (plaquettes, cartes de visite, site Internet...)		1 200
Frais de réception		800
Honoraires commissaire aux comptes		2 750
Prime assurance Cyb@rdéche (RC, vol...)		850
Services bancaires		1 200
Total charges de fonctionnement		13 400

Le Planning (suite)

Estimation des charges des tâches et de la durée du projet

Différents besoins d'estimation se font valoir au niveau du projet, des workpackages et des tasks.

- Projet : estimer la charge du projet complet par la détermination d'une enveloppe budgétaire.
- WP : estimer la charge d'une phase spécifique, ajuster le découpage du projet et prévoir des ressources pour planifier l'affectation des intervenants.
- TK : estimer chacune des tâches qui font généralement l'objet d'une affectation individuelle.

Les coûts du projet doivent être évalués en fonction de leur nature : coûts en matériel, en ressources humaines internes, en frais de déplacement, en sous-traitance (personnel de prestataires extérieurs), ... Concernant les charges matérielles, on en distingue globalement deux types :

- Celles propres au projet, p.ex. l'achat de la licence d'un logiciel spécifique au projet, les consommables ;
- Celles intervenant dans un investissement à long terme, dont : p.ex. l'achat d'une nouvelle machine.

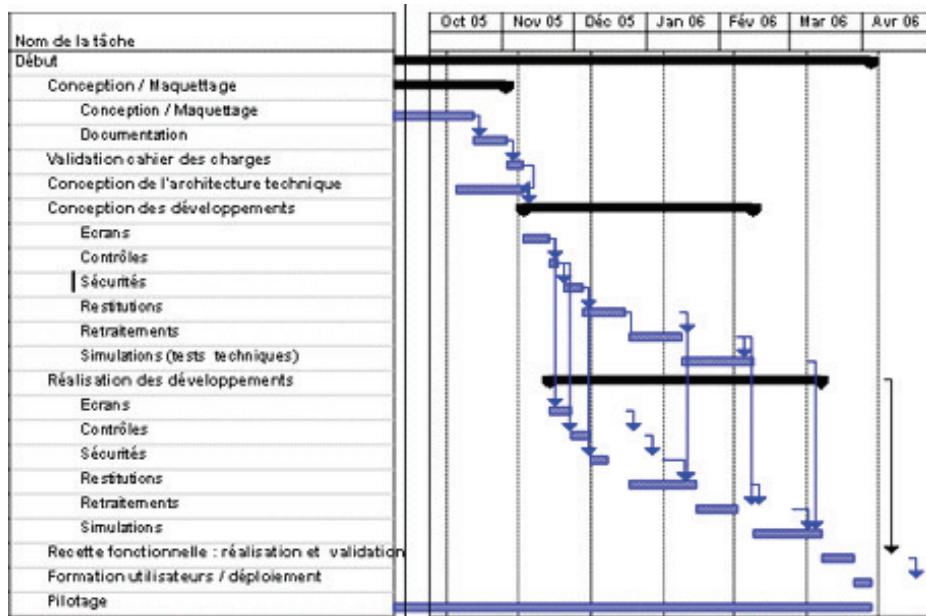
Les charges telles que locaux, ordinateurs, serveurs, logiciels doivent être estimées précisément.

Quand seulement du matériel commun est utilisé, alors on peut appliquer des montants forfaitaires calculés par la comptabilité.

Après cette phase de définition des besoins, il s'agit de définir les processus d'approvisionnement et d'établir les délais d'approvisionnement en fonction des fournisseurs. Il faut aussi évaluer le temps de recrutement des ressources humaines, du choix de prestataires éventuels.

L'évaluation de ces durées est importante dans le calcul total de la durée du projet.

Techniques de planification : GANTT



Techniques de planification

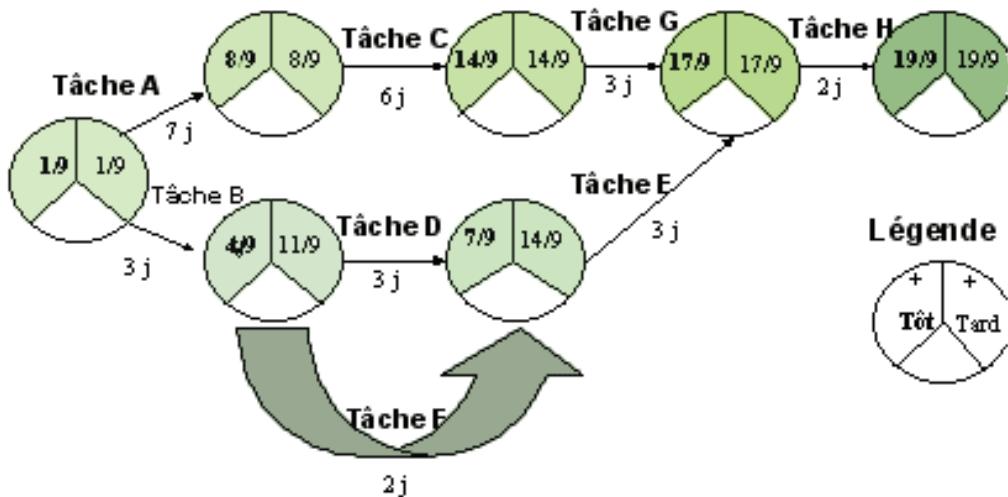
La construction du planning passe par la modélisation du réseau de dépendance entre tâches sous forme graphique. Il s'agit d'une décomposition structurée du travail. Il faut décomposer le projet en sous-ensembles plus simples (p.ex. WBS). Plusieurs représentations existent, à la base de toute construction de planning :

- GANTT : planning à barres
- PERT : méthode des potentiels étape et planning des tâches
- Réseau des antécédents : méthode des potentiels tâche

Le diagramme de GANTT

C'est la technique et représentation graphique permettant de renseigner et situer dans le temps les phases, activités, tâches et ressources du projet. Les lignes sont les tâches et les colonnes les jours, semaines ou mois. Les tâches sont représentées par des barres dont la longueur est proportionnelle à la durée estimée. Les tâches peuvent se succéder ou se réaliser en parallèle entièrement ou partiellement.

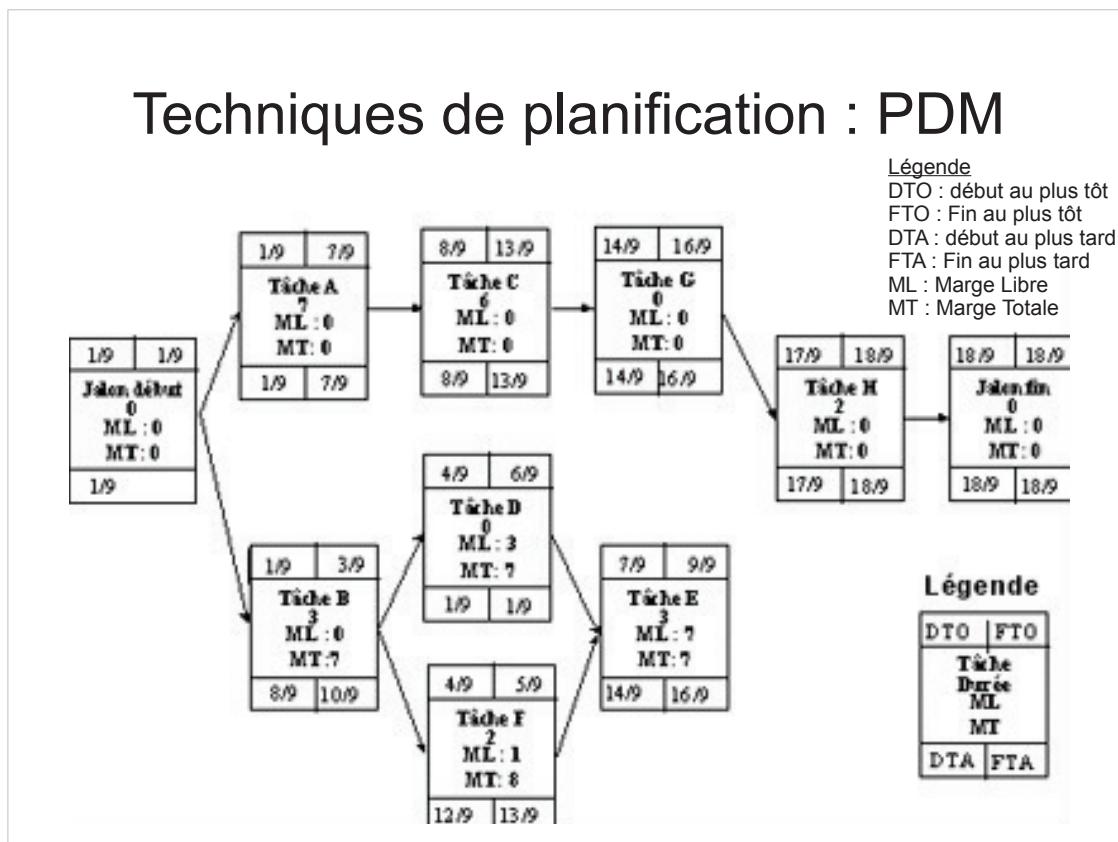
Techniques de planification : PERT



La technique PERT (« Program Evaluation and Review Technique »)

Cette technique américaine de modélisation de projet consiste à mettre en ordre sous forme de réseau plusieurs tâches qui grâce à leurs dépendances et à leur chronologie permettent d'avoir un produit fini. Les Caractéristiques de PERT sont les suivantes :

- Les tâches sont représentées par des flèches
- Le réseau visualise des dépendances entre tâches
- Limitation : pas de représentation de notion de durée et de date



Precedence Diagramming Method (ou « réseau des antécédents »)

L'établissement de ce réseau correspond à la méthode des potentiels tâches ou méthode du Chemin critique ou encore CPM (Critical Path Method). Ce réseau est aussi appelé PDM (Precedence Diagramming Method).

Chaque activité y est représentée par une boîte. Les activités sont liées entre elles par des liaisons de dépendances représentées par des flèches. C'est une représentation synthétique des relations logiques entre activités, construit de gauche à droite pour représenter la chronologie d'un projet.

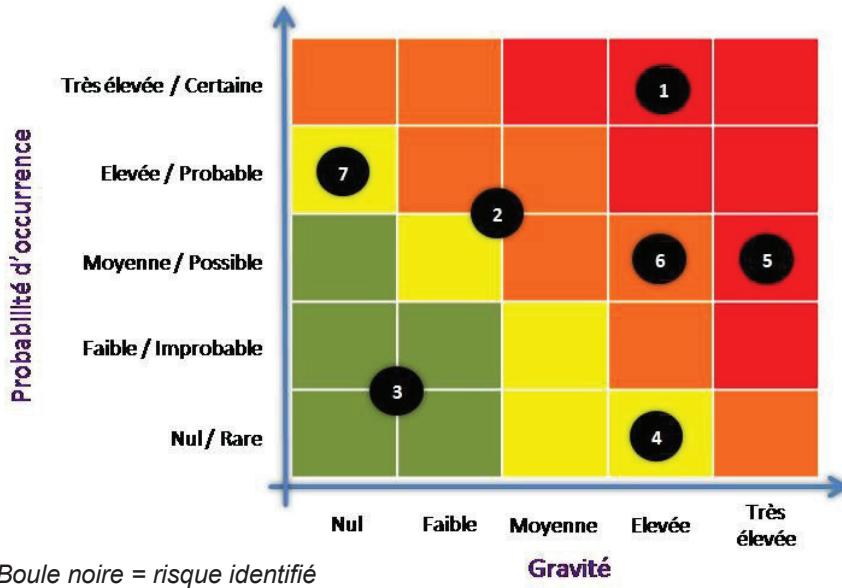
L'avantage de ce réseau des antécédents est qu'il permet une visualisation claire de la logique des dépendances, il donne la possibilité de relations avec des délais (écart/recouvrement). Cette représentation a été notamment retenue dans certains outils de gestion de projet : PMW (Project Management Workbench) et Microsoft Project.

Éléments de gestion de projets

3. Initialisation

- 1) Spécifications détaillées
- 2) Ressources humaines
- 3) Découpage
- 4) Planification du travail
- 5) Gestion des risques**
- 6) Planification de l'acceptance
- 7) Budgétisation
- 8) Communication interne

Planification : gestion des risques



Des études préalables permettent d'évaluer les risques liés au projet. La démarche d'identification des risques s'inscrit dans une volonté d'anticipation pour réagir au plus tôt. Cette démarche passe par l'identification des facteurs de risque associés à chaque tâche et de leur classification en fonction de leur criticité : ceux qui pourraient entraîner de légers retards dans le planning ou ceux qui bloquent la continuation du projet car appartenant au chemin critique.

Il est important d'introduire dans la planification le risque et l'incertitude associés à chaque tâche et d'en déduire une durée du projet assortie d'un niveau de probabilité. Différents types de risque peuvent être identifiés :

- humains (absence, décès d'une ressource importante sur le projet),
- coûts cachés (découverte de coûts à cours du projet qui grèvent l'enveloppe budgétaire dédiée au projet),
- retard dans les approvisionnements en matériaux indispensables au projet (risque de changement de la durée totale du projet),
- retard dans la livraison des livrables,
- technologiques (évolution de la technologie en cours de projet),
- manque de communication et de coordination,
- inadéquation des développements informatiques aux besoins exprimés.

Les risques doivent être classés par ordre d'importance. Il faut déterminer les conséquences potentielles liées à ces risques en terme d'impact financier, d'impact de délai ou d'impact sur la qualité. En cas de soucis importants mettant en péril le projet, un plan de secours peut être appliqué. Ce dernier est établi lors de l'étude préalable et lorsque les risques majeurs ont été identifiés.

La matrice des risques se formalise avec, en abscisse, le degré de gravité du risque et, en ordonnée, le degré de probabilité de l'occurrence du risque. Elle est évolutive au cours du projet : en fonction de la réalité du projet, un risque peut s'avérer beaucoup plus dangereux.

Éléments de gestion de projets

3. Initialisation

- 1) Spécifications détaillées
- 2) Ressources humaines
- 3) Découpage
- 4) Planification du travail
- 5) Gestion des risques
- 6) Planification de l'acceptance**
- 7) Budgétisation
- 8) Communication interne

Planification des tests

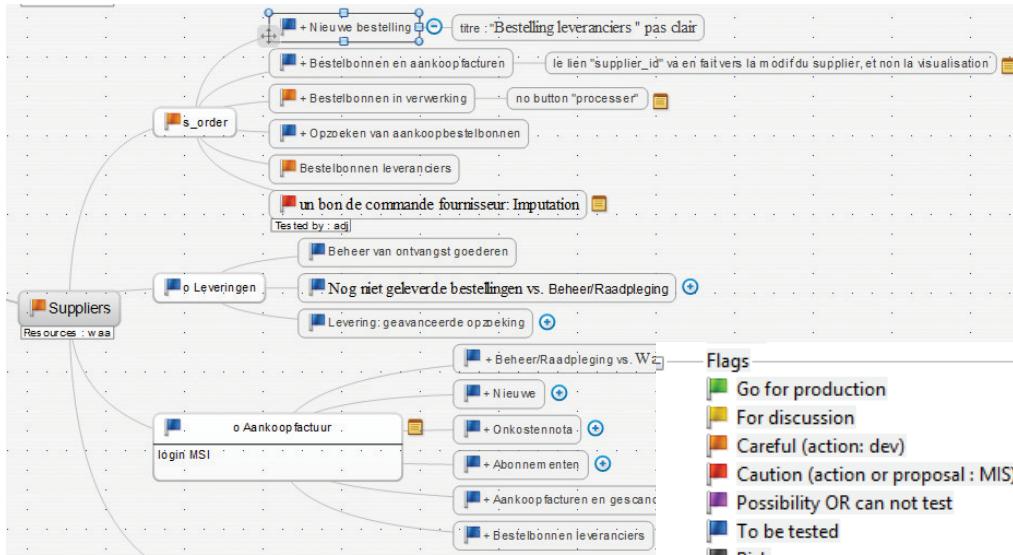


Les phases de test permettent de vérifier d'une part, le bon **fonctionnement** intrinsèque des applications livrées et d'autre part, l'**adéquation** entre les fonctions réalisées par ces applicatifs et les spécifications demandées. Les tests font l'objet d'une planification spéciale, indépendante de la planification du travail.

En fonction des circonstances, différents tests peuvent ou doivent être prévus :

- Automated Testing : test conduit automatiquement par un logiciel qui exécute des instructions programmées et valide le résultat obtenu.
- Manual Testing: test conduit par un être humain.
- Functional Testing : test validant des spécifications fonctionnelles, ou vérifiant certains comportements.
- Compatibility Testing : test validant la compatibilité du système avec d'autres systèmes existants ou avec des anciennes versions du même système.
- Integration Testing : test validant les composants individuels d'un système, dans le cadre de leur fonctionnement individuel.
- Performance ou Stress Testing: test vérifiant que la performance du système est conforme aux spécifications, en particulier s'il placé dans ses conditions limites de charge ; à noter que ce test suppose de pouvoir simuler le système dans des conditions de production.
- Regression Testing : test vérifiant que les modifications apportées au système pour corriger des problèmes les ont bien corrigés et n'en ont pas introduit de nouveaux.
- Conformance Testing: test validant la conformité du deliverable avec les spécifications.
- Workflow Testing : test validant la fonctionnalité et la robustesse du système d'un bout à l'autre du processus utilisateur réel ; à noter que ce test suppose de pouvoir utiliser une infrastructure réelle.
- Ad-Hoc Testing : test validant les fonctionnalités du système par le biais d'utilisateurs finaux libres de l'utiliser à leur guise (p.ex. en leur demandant de découvrir ce qui ne fonctionne pas).
- Acceptance Testing : batterie de tests vérifiant que le système livré respecte les critères d'acceptance pré-définis du projet ; à noter que ces tests prennent place après la réussite des tests précédents, notamment ceux purement techniques.

Deliverables de tests



Comme pour n'importe quel autre processus de projet, la phase de test nécessite ses propres délivrables :

- « Test Plan Deliverable » : une feuille de route pour le processus de test, en précisant les objectifs généraux, la portée, les hypothèses de test, la logistique, les risques, les tâches et les affectations de ressources.
- « Test Specification Deliverable » : détails des tests à effectuer en fonction du type, le but, résultat escompté et méthodologies.
- « Test Script Deliverable » : un ensemble spécifique d'instructions à exécuter, pour chaque élément du processus de test. Les scripts de test peuvent être automatisées (à l'usage des outils de test de logiciel) ou manuel (à exécuter par les utilisateurs finaux ou le personnel IT selon le cas). En outre, les scripts de test doivent toujours prévoir l'enregistrement des résultats.

Une bonne base pour écrire ces deliverables est le scénario utilisateur. Reprenez celui-ci, détaillez davantage les opérations à effectuer et les résultats escomptés (p.ex. copies d'écran), et vous obtenez aisément votre test script deliverable.

La phase de test étant un deliverable comme un autre, les tests seront définis, approuvés et planifiés – en termes de temps comme de charge - lors de la phase d'initialisation du projet, comme n'importe quelle autre tâche du projet (développement etc.).

Il ne faut pas négliger le temps à consacrer aux tests, car ils ont pour but de mettre en évidence des problèmes qui exigeront d'être résolus à court terme. Soyez donc prévoyants !

Éléments de gestion de projets

3. Initialisation

- 1) Spécifications détaillées
- 2) Ressources humaines
- 3) Découpage
- 4) Planification du travail
- 5) Gestion des risques
- 6) Planification de l'acceptance
- 7) Budgétisation**
- 8) Communication interne

Budgétisation

Budget prévisionnel pour un nouveau site Internet					
Tarifs journaliers des ressources humaines du projet					
€ HT	consultant	infographiste	Développeur	Ingénieur Expert	chef de projet
tarif/jour	760	450	570	750	900
Coût total des ressources humaines du projet					
	consultant	infographiste	Développeur	Ingénieur Expert	chef de projet
Consulting stratégique	10				1
Charte graphique et navigation		5	3		1
Mailing liste			5		
Profiling (dont formulaire inscription)			10	3	1
Personnalisation		2	10	3	1
Création d'une base de données			10		
Création d'un moteur de recherche			10	2	
Sécurisation du site			5	5	
Création d'un forum	1	5			
Création d'un panier	2	5			
Gestion d'adresses e-mails			5		
formulaire de contact			1		
Sondage et vote en ligne	1	2			
Test, installation & maintenance			3		
Suivi de projet					15
Référencement				25	
TOTAL HT					101 080
TOTAL TTC					663 041

Nous allons voir ici comment établir le budget d'un projet, et en guise d'exemple la budgétisation des ressources humaines d'un site Internet.

Evaluation du coût du projet

Le coût du projet est la somme des coûts :

- des ressources humaines du projet
- des ressources matérielles et logicielles du projet

Ce coût dépend évidemment de la durée du projet.

Exemple de budgétisation

Ce projet exemplatif est la création d'un site d'e-commerce pour une entreprise qui souhaite vendre en ligne ses produits.

Premier tableau : Un tarif journalier en fonction du profil de compétence est établi.

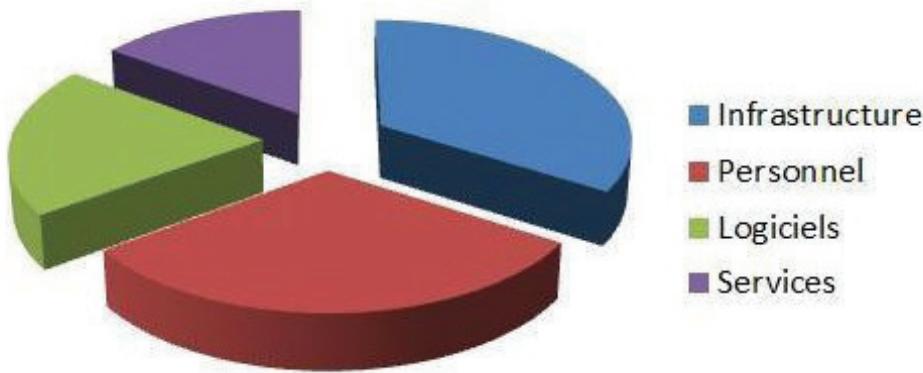
Second tableau : chaque ligne représente une tâche, avec le nombre de jours d'intervention par compétence qu'elle requiert et son coût total.

Par exemple, la tâche "sondage et vote en ligne" a comme coût : $1 * 450 + 2 * 570 = 1590$ euros, soit 1 jour d'infographie à 450 euros la journée et 2 jours de développement à 570 euros la journée.

Concernant les ressources internes, on peut calculer leur coût journalier en se basant sur leur salaire-coût (salaire brut + charges patronales). Ensuite, il suffit de multiplier ce coût journalier par le nombre de jours prévus d'intervention sur le projet.

Il reste enfin à évaluer les autres coûts (mais c'est plus facile), notamment le coût des licences des logiciels nécessaires au projet, des serveurs de tests et des serveurs de production.

Achat, Investissements, Sous-traitance, Frais généraux



Toute dépense aura fait l'objet d'une budgétisation préalable en début de projet. Il sera souvent difficile de procéder à un achat non budgétisé en cours de projet, car c'est une atteinte directe à la marge bénéficiaire escomptée.

On distingue plusieurs types de dépense matérielle :

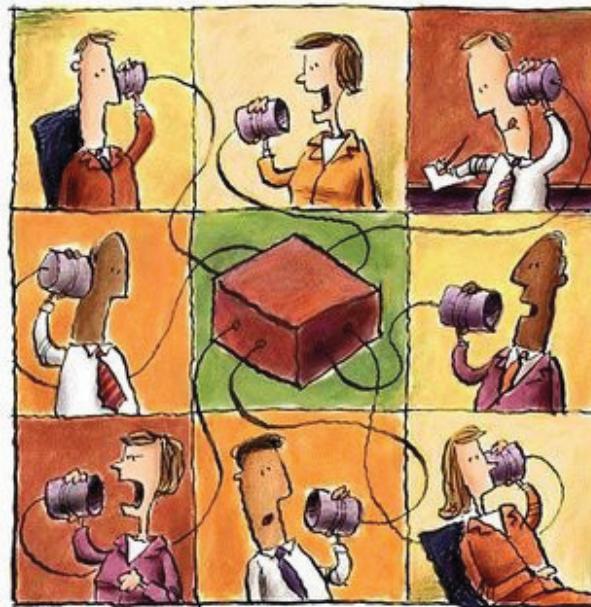
- Un **achat** est une dépense faite exclusivement et spécifiquement dans le cadre du projet. En d'autres termes, l'entièreté du montant de la dépense sera comptabilisé sur le projet.
p.ex. Une machine qui sera configurée pour être remise au client en fin de projet
p.ex. Les consommables propres au projet
 - Un **investissement** est une dépense qui se répartit sur plusieurs projets, passés, présents ou futurs. En d'autres termes, seule une fraction du montant de la dépense sera affectée au projet.
p.ex. Un serveur de développement ; si on estime que ce serveur sera utilisé pendant 6 années, dont 2 exclusivement pour ce projet, alors 1/3 du coût du serveur sera comptabilisé sur le projet ;
 - Une **sous-traitance** est un appel à une ressource humaine extérieure et n'est donc pas une dépense matérielle. Toutefois, sur le plan comptable, il s'agit d'une dépense du même type qu'un achat. L'entièreté du montant de la sous-traitance est comptabilisé sur le projet.
 - Les **frais généraux** couvrent l'utilisation des ressources générales de l'entreprise, hors-projet :
 - Matériel : locaux, ordinateurs, serveurs, logiciels office, ...
 - Sous-traitance : catering, nettoyage, ...
 - Humain : secrétariat, département HRM, département IT et aussi ... le Management, en bref, tout le personnel qui n'est pas affecté aux projets est comptabilisé dans les frais généraux.
- Il y a deux modes d'imputation des frais généraux au projet :
- Analytique : on calcule précisément la part des frais généraux qui a été affectée au projet.
 - Forfaitaire : Par souci de simplification, on calcul la part des frais généraux comme étant une **fraction** du coût du personnel affecté au projet, p.ex. 15%.

Éléments de gestion de projets

3. Initialisation

- 1) Spécifications détaillées
- 2) Ressources humaines
- 3) Découpage
- 4) Planification du travail
- 5) Gestion des risques
- 6) Planification de l'acceptance
- 7) Budgétisation
- 8) Communication interne**

Communication interne au projet



Une communication interne efficace est un impératif de tout projet, alors que les besoins en information et les méthodes de diffusion varient considérablement. La planification des communications formelles doit être incluse dans la planification globale du projet, notamment déterminer la fréquence nécessaire des réunions. En particulier, il faut évaluer :

- Les relations de responsabilités entre parties prenantes et organisation en charge du projet
- Les disciplines, services et spécialités impliquées dans le projet
- L'implantation géographique et la mobilité géographique des acteurs du projet
- Les besoins éventuels en information externe (p.ex. la communication avec les medias)

Dans le cadre des projets impliquant de nombreux acteurs et services, il est recommandé d'établir un **plan de communication** en début de projet pour faciliter la communication pendant le projet.

Le plan de communication présente :

- Les méthodes utilisées pour collecter et conserver différents types d'informations. Les procédures doivent également préciser les modalités de collecte et de diffusion des mises à jour et des corrections.
- Les destinataires de l'information en fonction de la nature des informations (rapports d'avancement, données, calendrier, documentation technique...) , les méthodes utilisées pour diffuser les divers types d'information et les diffuseurs de ces informations.
- Une description de l'information à diffuser : le format, le contenu, le degré de détail, les conventions et définitions à utiliser.
- Les calendriers d'émission qui précisent à quel moment chaque type d'information est émis
- Les méthodes pour accéder à l'information entre deux communications prévues
- Une méthode de mise à jour et de redéfinition du plan de communication au cours du projet

Le plan de communication peut être formalisé ou informel, peu ou très détaillé, suivant la nature du projet.

Technologies et support de « com »



L'obtention d'une communication efficace pendant le projet passe par le choix des outils et des supports adaptés au projet. Pour ce faire, plusieurs points doivent être soulevés:

- L'urgence du besoin d'information : le succès du projet dépend-il d'une information mise à jour régulièrement et disponible à tout moment, ou est-ce que des rapports écrits réguliers suffisent ?
- La technologie disponible : les technologies déjà en place suffisent-elles pour communiquer ou faut-il mettre en place de nouveaux moyens de communication (Intranet, installations de logiciels de télé-conférence...)
- Le niveau de qualification des parties prenantes du projet : Les systèmes de communication proposés sont-ils compatibles avec l'expérience ou le niveau de compétence des participants ? Si tel n'est pas le cas, des formations aux outils de communication sont à prévoir.
- La durée du projet : Dans le cas de projets longs, il faut se demander si la technologie actuelle n'est pas susceptible d'évoluer et ne nécessite pas de mise à jour au cours du projet.

Plusieurs types de communication peuvent être utilisés :

- réunions informelles entre deux parties prenantes du projet,
- réunions officielles entre responsables d'un projet.

Les documents produits peuvent être des simples documents Ms-Word diffusés par e-mail, uploadés sur un intranet ou sur le cloud (Google Drive) voire discutés via visio-conférence (Skype) si la dispersion géographique des acteurs du projet l'impose.

Informations pertinentes



La détermination des informations pertinentes du projet à communiquer passe par l'analyse des besoins en communication des acteurs du projet. Les informations à communiquer dépendent de la nature des interlocuteurs :

- Direction générale
- Instances de pilotage (chefs de mission, chefs de projet)
- Acteurs du projet de type « fonctionnel » (Maîtrise d'Ouvrage ou encore MOA)
- Acteurs du projet de type « technique » (Maîtrise d'oeuvre ou encore MOE)

La **Direction** s'intéresse à l'avancement global du projet, aux grandes lignes de celui-ci, aux risques majeurs, aux obstacles potentiels intervenant éventuellement au cours du projet.

Les **instances de pilotage** regroupent les différents responsables du projet, qui récoltent régulièrement les informations auprès de leur équipe. Ils se réunissent au cours de comités de pilotage pour faire un point global sur l'avancement du projet, sur les difficultés importantes rencontrées et prennent des décisions quant à d'éventuelles mesures correctives. Par après, ils communiquent les compte-rendu de réunion aux acteurs du projet sous leur responsabilité, concernant les décisions prises, les éventuelles conséquences des interactions entre les équipes projet, et leur donne une visibilité globale qu'ils n'ont pas sur l'avancement, sur les difficultés apparues dans d'autres équipes et pouvant les impacter ou sur les difficultés qui risquent d'affecter l'ensemble des équipes projet.

Les **acteurs de type fonctionnel** et **les acteurs de type technique** sont amenés à communiquer tout au long d'un projet. Les premiers rédigent une étude préalable puis un cahier des charges fonctionnel pour répondre aux objectifs du projet. Les seconds, après lecture de ces documents rédigent un cahier des charges technique, qui constitue la réponse en terme de développement technique. Tous ces documents sont validés par les responsables projet. Par ailleurs, ils peuvent être amenés à communiquer entre eux pour comparer leurs méthodes de travail, leurs solutions à une problématique donnée, les difficultés rencontrées ou les astuces trouvées permettant de gagner du temps.

Éléments de gestion de projets

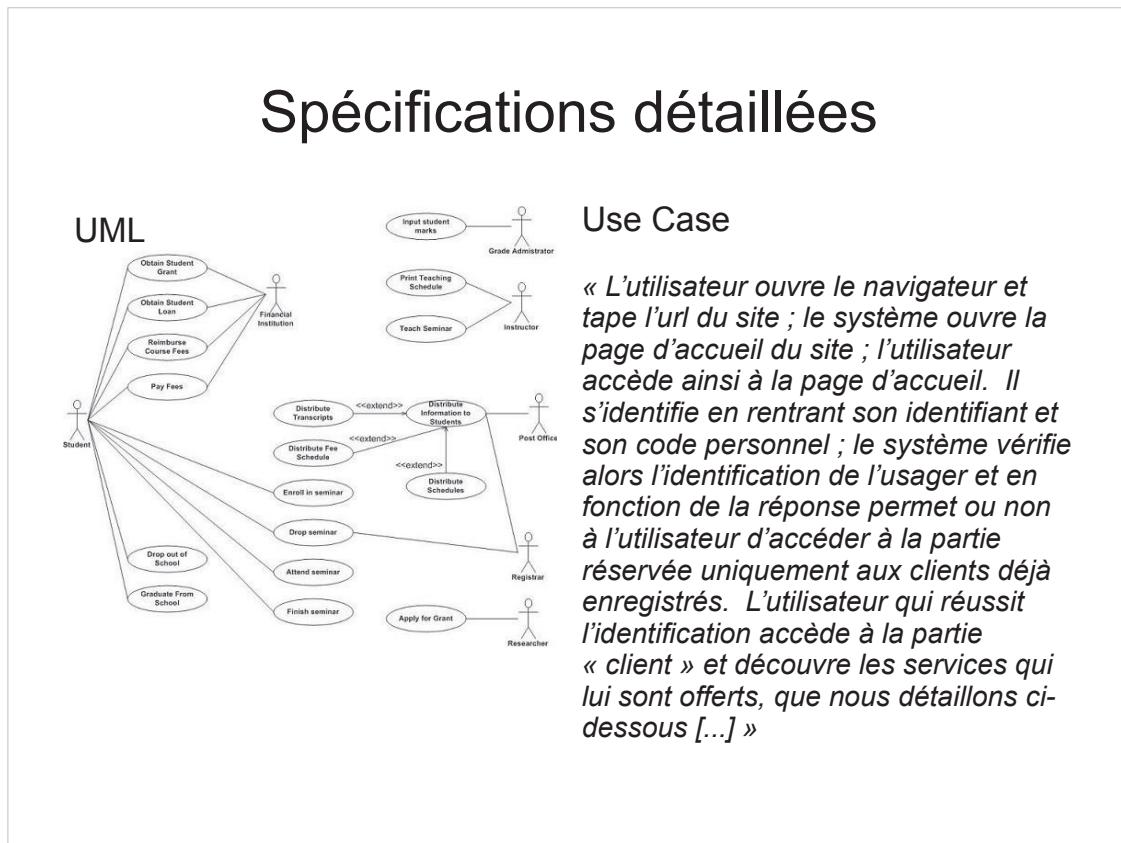
4. Réalisation

- 1) Étude détaillée
- 2) Recherche de solutions
- 3) Développement & Configuration
- 4) Documentation

La réalisation du projet peut prendre des formes extrêmement variées. Nous verrons dans ce cours les éléments que l'on retrouve le plus souvent dans des projets IT :

- Etude détaillée de la problématique
- Recherche de solutions et « make or buy ? »
- Développement et configuration de logiciel
- Documentation

Cette liste est tout à fait « officieuse » et peut varier d'un projet à un autre.



Le but de cette phase est de concevoir ou de spécifier ce qui doit être réalisé ou fabriqué pour atteindre l'objectif, sur base du cahier des charges (s'il existe). Ces études associent la maîtrise d'ouvrage et la maîtrise d'œuvre. Le niveau de détail doit être suffisamment fin pour éviter toute incompréhension mutuelle, toujours dommageable quand elle est découverte plus (si pas trop) tard.

Cette étude détaillée peut prendre plusieurs formes bien différentes :

- Listing des fonctionnalités
- Scénario d'utilisation typique, dans un style un peu romanesque, avec comme illustration éventuelle des esquisses grossières des écrans utilisateur
- Diagrammes UML « use case »

Dans le cas d'un système IT, l'étude commencera souvent par lister les catégories d'utilisateurs ou de fonctions : p.ex. client lambda, helpdesk, responsable de service, manager, etc., puis de préciser les workflows dans lesquels ils sont impliqués. Dans ce cas, il peut être indiqué d'utiliser des schémas BPMN.

Cette étude détaillée doit finalement être approuvée par la MOA et la MOE. Elle servira de référence en cas de litige lors de l'acceptation !

Elle peut aussi contenir des informations techniques fort détaillées qui, si elles ne sont pas utiles au MOA, seront précieuses pour le MOE à un certain moment.

Recherche de solutions

Général	Open Office.org	Microsoft Office
Prix 10%	Gratuit	Payant(139 €=> 599 €)
Traitement de texte 5%	Oui (OpenOffice Writer)	Oui (Microsoft office word)
Tableur 5%	Oui (OpenOffice Calc)	Oui (Microsoft office Excel)
Outil de présentation 3%	Oui (OpenOffice Impress)	Oui (Microsoft office Powerpoint)
Gestionnaire d'informations personnelles 2%	Non	Oui (Microsoft office Outlook) Non disponible dans la version familiale (139€)
Gestionnaire de base de donnée 2%	Oui (OpenOffice Base)	Non
Traitement de textes		
	Writer	Word
Poids sans image 5%	~35'000 Ko	~25'000 Ko
Poids avec image 5%	~55'000 Ko	~60'000 Ko
Compatibilité 6%	Ouvre la majorité des fichiers(dont ceux de word)	Ouvre une grande partie des fichiers (mais pas ceux de writer)

- Table
- Col : solutions
- Row : specs
- Évaluation
- Code couleur
- Pondération

	Open Office.org	Microsoft Office
Format	Tous	Tous
Dépendance de tiers	Oui (Apache OpenOffice)	Oui (Microsoft office)
Tableau	Oui (Apache OpenOffice Calc)	Oui (Microsoft office Excel)
Outil de présentation	Oui (Apache OpenOffice Impress)	Oui (Microsoft office Powerpoint)
Gestionnaire d'informations personnelles	Oui (Apache OpenOffice Base)	Oui (Microsoft office Outlook) Non disponible dans la version familiale (139€)
Gestionnaire de base de données	Oui (Apache OpenOffice Base)	Oui (Microsoft office Access)
Traitement de textes		
Format	Tous	Tous
Poids sans image	35'000 Ko	25'000 Ko
Poids avec image	55'000 Ko	60'000 Ko
Compatibilité	Ouvre la majorité des fichiers (dont ceux de Word)	Ouvre la grande partie des fichiers (mais pas ceux de Word)
Grammaire et orthographe	Ouvre les fichiers préparés par Microsoft Word	Ouvre les fichiers préparés par Microsoft Word
Qualité correction	Très bonne qualité de correction	Qualité basse de correction
Corriger automatiquement	Oui	Oui
Tableau	Oui	Oui
Poids sans image préparés	35'000 Ko	25'000 Ko
Poids avec image préparés	55'000 Ko	60'000 Ko
Compatibilité préparés	Ouvre les fichiers préparés par Microsoft Word	Ouvre la grande partie des fichiers préparés par Microsoft Word
Autres		
Format de fichiers	OpenOffice.org	Microsoft Office
Poids des fichiers - initial	35'000 Ko	25'000 Ko
Poids fichiers - final	55'000 Ko	60'000 Ko
Compatibilité	Ouvre les fichiers préparés par Microsoft Word	Ouvre la grande partie des fichiers préparés par Microsoft Word
Autorisé	Oui	Oui
Total des fichiers	OpenOffice.org	Microsoft Office

Cette phase consiste à étudier **différentes solutions** ou architectures techniques et fonctionnelles en fonction de contraintes de compétences, d'équipement, de délais ainsi que des aspects financiers et de commercialisation. Les choix doivent être ensuite validés par la réalisation de maquettes ou de prototypes (et éventuellement la mise sur un marché test). Les différentes solutions retenues, accompagnées de leur évaluation, seront présentées aux instances de pilotage qui sont les mieux habilités à choisir la solution optimale.

Dans les projets informatiques, cette phase prend en compte les préoccupations d'urbanisation et d'architecture. En d'autres termes, le choix d'une solution IT doit être compatible avec les processus, la culture et les systèmes existants du MOA, afin d'assurer une migration rapide sur les plans tant technique que humain.

Marché public

Lors d'un choix d'une solution du marché (cas des progiciels notamment), cette phase s'articule autour d'un appel d'offres. Si le MOA est une organisation publique, alors l'appel d'offre fera l'objet d'un **marché public**.

« make or buy » ?

Il y a souvent un choix à faire : « make or buy ? ». Fabriquer en interne ou acheter auprès d'un fournisseur externe ? Une entreprise peut décider d'acheter si

- Il est moins cher à acheter qu'à produire
- Il n'y a pas les compétences ou les capacités suffisantes pour produire en interne,
- La réactivité autour du produit est faible (càd : le produit ne demande pas trop d'adaptation)
- Le produit offre une faible valeur ajoutée ou n'est pas stratégique

Mais la même entreprise peut décider de produire en interne si

- La capacité de production est présente
- Le contrôle qualité exigé est élevé
- Un secret de fabrication couvre le produit

Développement et configuration

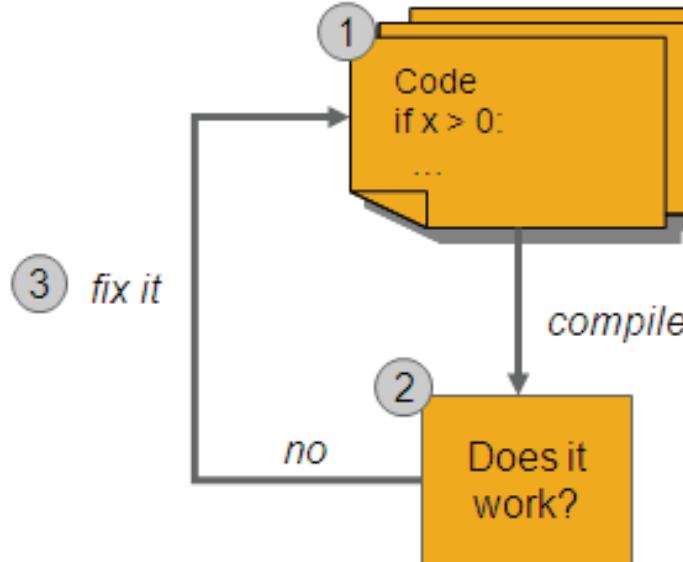


Souvent, dans un projet IT, le développement interne d'une application ou la configuration d'un progiciel jouent un rôle important. A l'heure actuelle, il devient un peu obsolète de différencier les deux activités. En effet, on développe en général à partir d'un framework et rarement « from scratch », tandis que toute configuration de progiciel demande des capacités de programmation.

Partant des scénarios utilisateur ou des diagrammes UML « use case », on distingue en général les phases suivantes dans un développement ou une configuration :

1. L'interface utilisateur
2. La base de données et/ou les web services
3. Le middleware, le « cerveau » de l'application qui relie tous les éléments internes et externes, constitué typiquement de
 1. Le « contrôle », mécanisme d'aiguillage traitant les informations provenant de l'utilisateur
 2. Les accès aux données et les algorithmes de manipulation de données
 3. La présentation de la réponse pour l'utilisateur (p.ex. page HTML)

Code and fix model



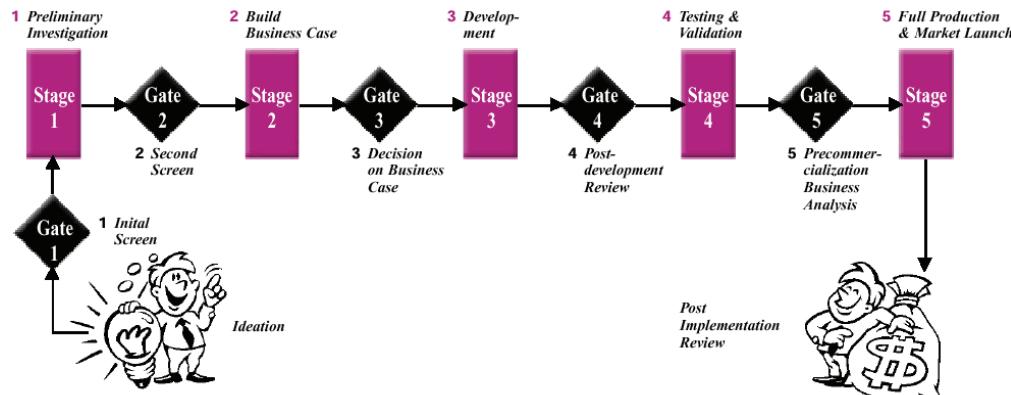
« Code and fix model » or « cow-boy coding » is a very simple type of model. Mainly it consists of two steps :

- Write some code
- Fix the problems in that code

This model is used in the early days of the software development. It can be used with the short systems which do not require maintenance.

"Code and fix" development is not so much a deliberate strategy as an artifact of naïveté and schedule pressure on software developers. Without much of a design in the way, programmers immediately begin producing code. At some point, testing begins (often late in the development cycle), and the unavoidable bugs must then be fixed before the product can be shipped.

jalonnement, stage gate, waterfall , cascade



The waterfall model is a sequential design process, used in software development processes, in which progress is seen as flowing steadily downwards (like a waterfall) through the phases of Conception, Initiation, Analysis, Design, Construction, Testing, Production/Implementation and Maintenance.

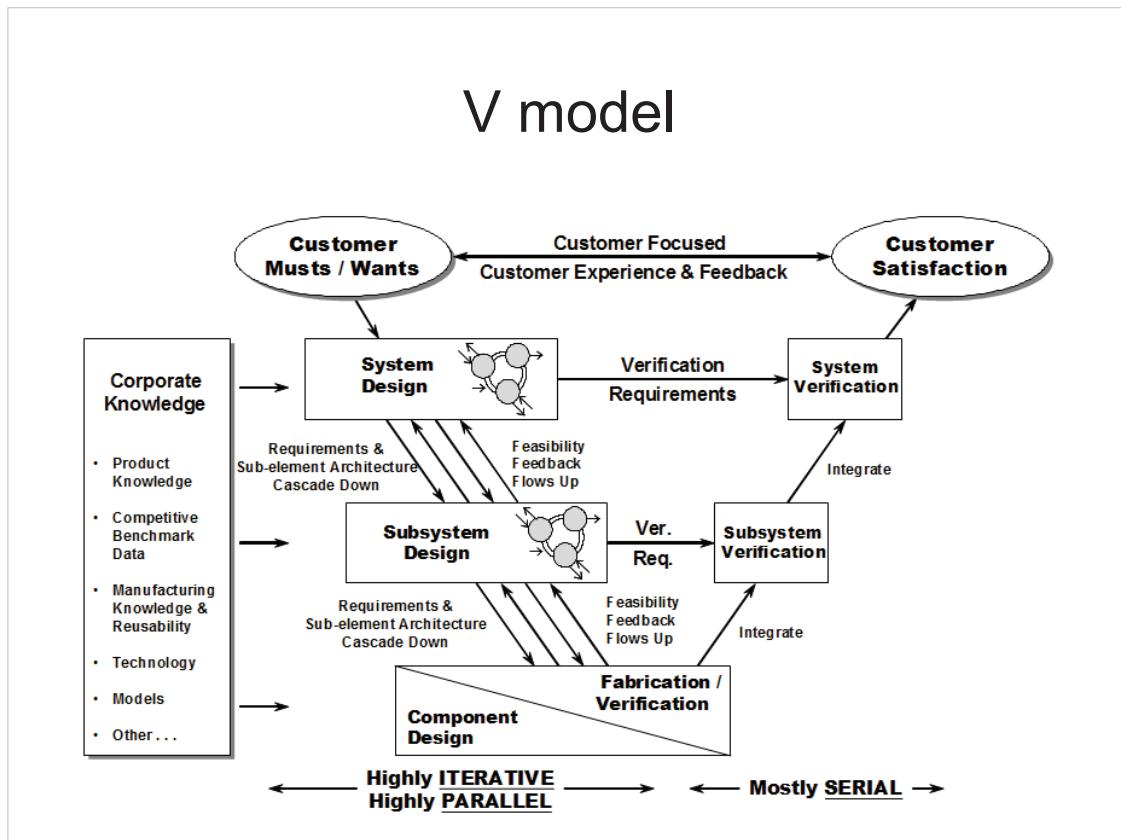
The waterfall development model originates in the manufacturing and construction industries; highly structured physical environments in which after-the-fact changes are prohibitively costly, if not impossible. Since no formal software development methodologies existed at the time, this hardware-oriented model was simply adapted for software development.

Supporting arguments

- A bug found in the early stages (such as requirements specification or design) is cheaper in money, effort, and time to fix than the same bug found later on in the process.
- It places emphasis on documentation (such as requirements documents and design documents) as well as source code. New team members or even entirely new teams should be able to familiarize themselves by reading the documents.
- The waterfall model can be suited to stable software projects where it is possible that designers fully predict problem areas of the system and produce a correct design before implementation.

Criticism

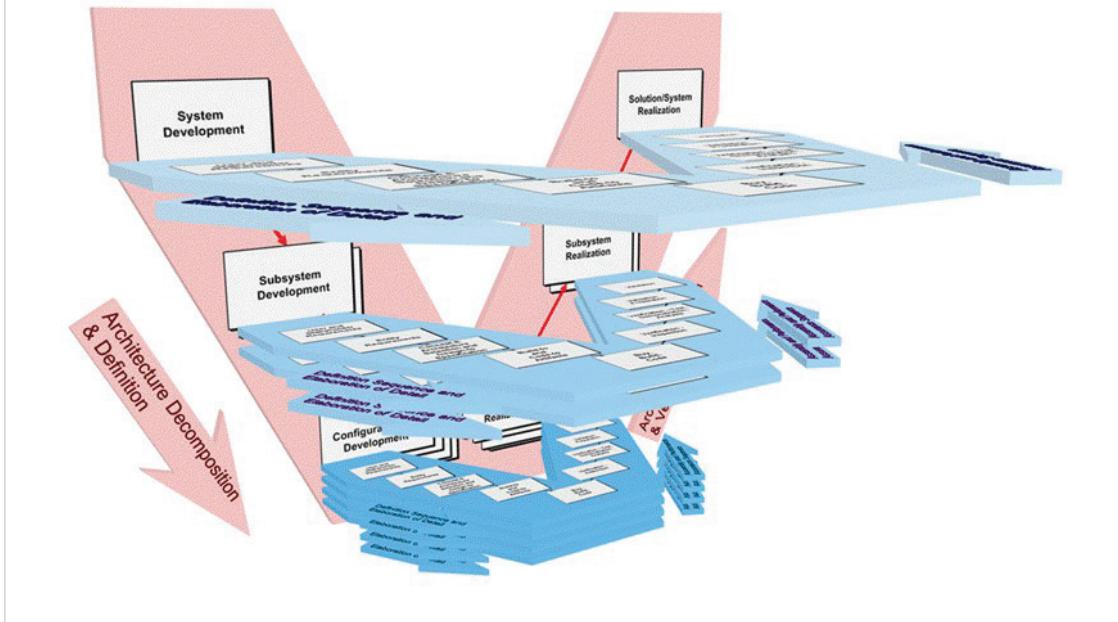
- Designers may not be aware of future implementation difficulties when writing a design for an unimplemented software product.
- Requirements and limitations cannot be entirely known before completion. The implication of this is that it is impossible to perfect one phase of software development, thus it is impossible if using the waterfall model to move on to the next phase.
- The project stakeholders may not be fully aware of the capabilities of the technology being implemented. This can lead to a design that does not use the full potential of what the new technology can deliver. This can cause substantial changes to the implementation requirements once the stakeholders become more aware of the functionality available from the new technology.



The V-model represents a software or hardware development process which may be considered an extension of the waterfall model. Instead of moving down in a linear way, the process steps are bent upwards after the coding phase, to form the typical V shape. The V-Model demonstrates the relationships between each phase of the development life cycle and its associated phase of testing. The horizontal and vertical axes represents time or project completeness (left-to-right) and level of abstraction (coarsest-grain abstraction uppermost), respectively.

« V » n'est pas seulement la lettre éponyme du schéma, c'est aussi l'initiale de « validation » et de « vérification ».

V V model

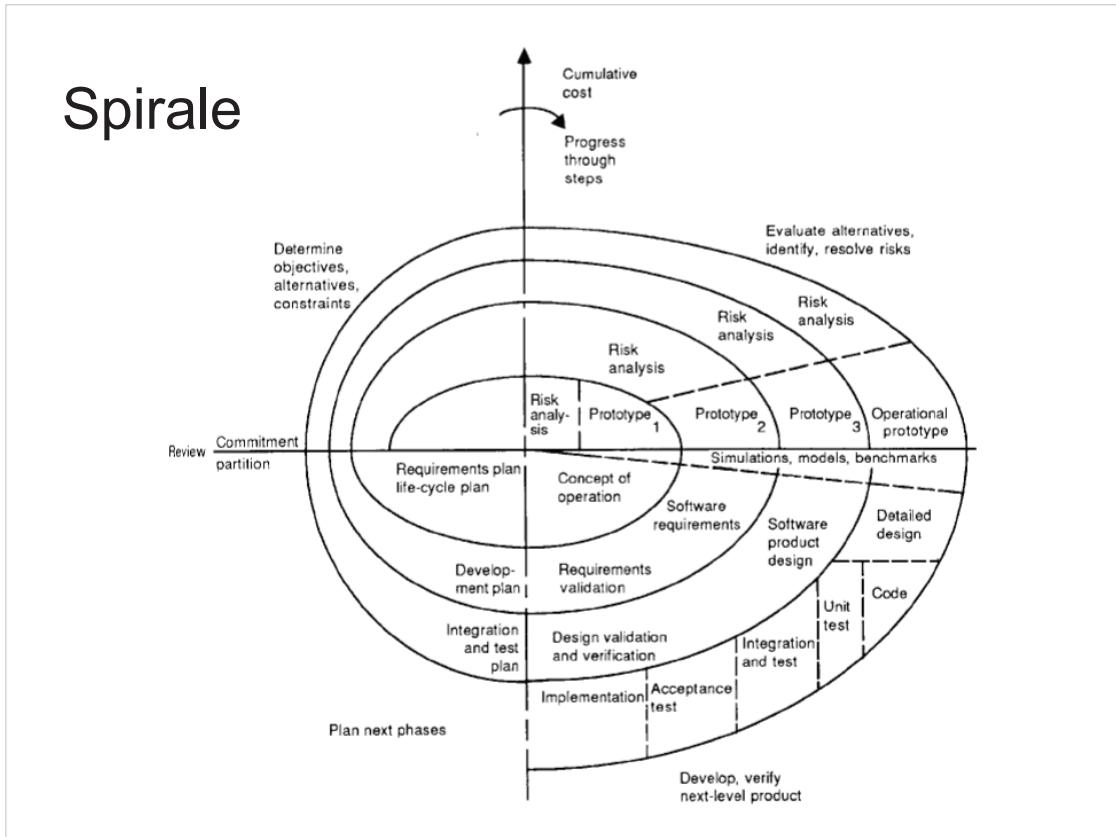


The Dual Vee Model builds on the Vee Model to manage a system of systems.

An architecture Vee manages the system and entity Vees branch off the architecture Vee to manage sub-systems.

Example

GPS includes a constellation of satellites, a ground control network, and millions of users worldwide. Each satellite, ground control center, and GPS receiver is a complex system that could be managed by a separate Entity Vee. Development of a satellite can affect the design, manufacturing, or cost of receivers. Similarly, development of a receiver can affect design, manufacturing, or cost of satellites. So everything must be integrated into a system of systems that are developed within a larger Architecture Vee.

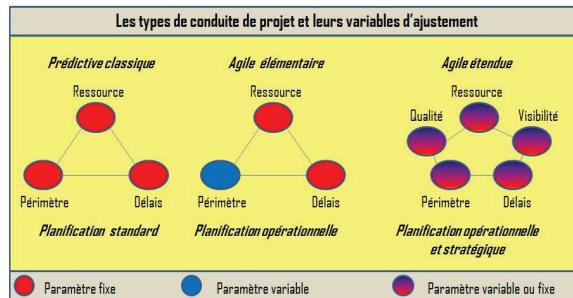
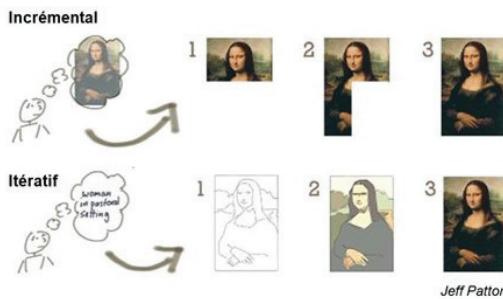


Le modèle en spirale (spiral model) est un modèle de cycle de développement logiciel qui reprend les différentes étapes du cycle en V. Par l'implémentation de versions successives, le cycle recommence en proposant un produit de plus en plus complet et dur. Le cycle en spirale met cependant plus l'accent sur la gestion des risques que le cycle en V.

On distingue quatre phases dans le déroulement du cycle en spirale :

- détermination des objectifs, des alternatives et des contraintes ;
- analyse des risques, évaluation des alternatives ;
- développement et vérification de la solution retenue ;
- revue des résultats et vérification du cycle suivant.

Agile



Une méthode agile est itérative, incrémentale et adaptative.

Une méthode agile est avant tout **itérative** sur la base d'un **affinement du besoin** mis en œuvre dans des fonctionnalités (partiellement) réalisées. Cet affinement, indispensable à la mise en œuvre du concept adaptatif, se réalise en matière de génie logiciel sous deux aspects :

- **fondamentalement**, par adaptation systématique du produit aux changements du besoin détecté par l'utilisateur lors de la conception-réalisation du produit ;
- **techniquement**, par remaniement régulier du code déjà produit (refactoring).

Une méthode agile est ensuite **incrémentale**. Lorsque le projet dépasse en durée une dizaine de journées en moyenne, la production de ses fonctionnalités s'effectue en plusieurs incrémentums.

Une méthode agile est enfin **adaptative**. Elle permet au client de modifier ses exigences en cours de projet, avec pour conséquence l'éventualité d'un périmètre variable (figure : Critères Itératif - Incrémental - Adaptatif). La notion **d'adaptatif** nécessite la mise en œuvre de techniques de contrôle de l'évolution du livrable et d'une **métrique formelle des modifications**, avant, après et en cours de la production.

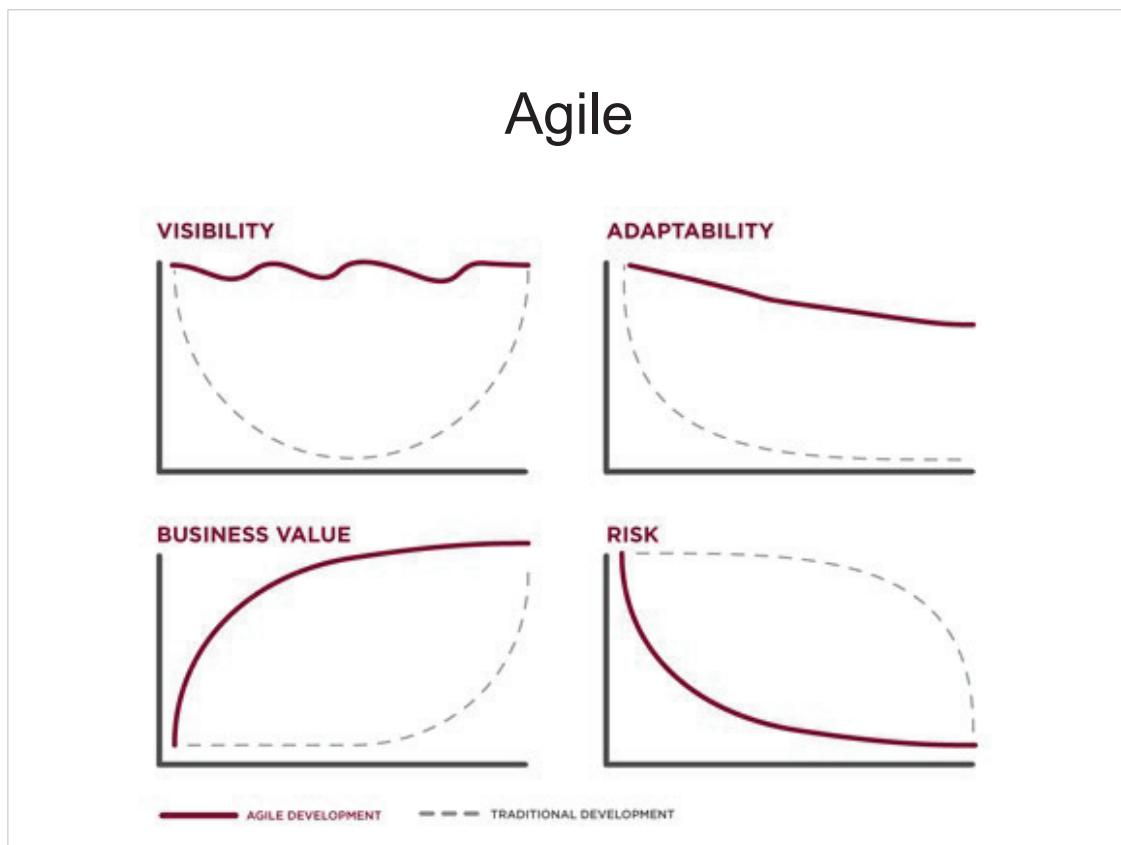
Agile



Valeurs

Les méthodes agiles prônent quatre valeurs fondamentales :

- L'équipe
L'équipe est bien plus importante que les outils ou les procédures de fonctionnement. Il est préférable d'avoir une équipe soudée et qui communique.
- L'application
Il est vital que l'application fonctionne. Le reste, et notamment la documentation technique, est une aide précieuse mais non un but en soi. Une documentation précise est utile mais comme moyen de communication.
- La collaboration
Le client doit être impliqué dans le développement. Le client doit collaborer avec l'équipe et fournir un compte rendu continu sur l'adéquation du logiciel avec ses attentes.
- L'acceptation du changement
La planification initiale et la structure du logiciel doivent être flexibles afin de permettre l'évolution de la demande du client tout au long du projet.



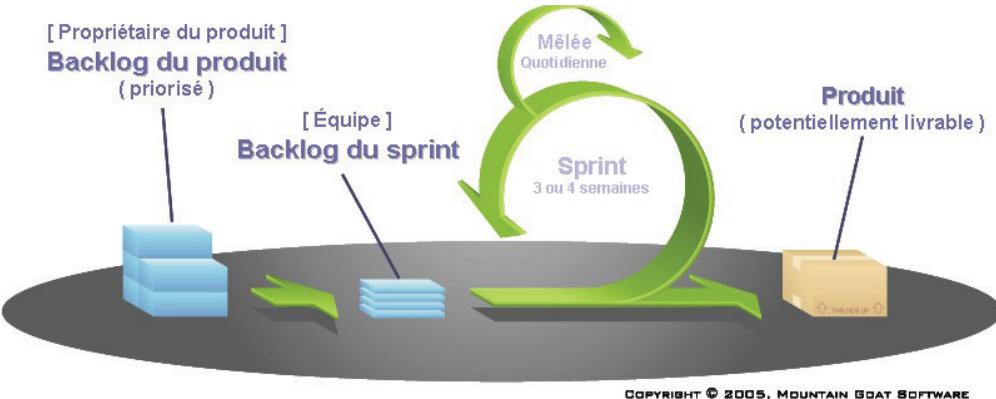
Principes

Ces 4 valeurs se déclinent en 12 principes généraux communs à toutes les méthodes agiles :

- La plus haute priorité est de satisfaire le client en livrant rapidement et régulièrement des fonctionnalités à forte valeur ajoutée.
- Le changement est accepté, même tardivement dans le développement, car les processus agiles exploitent le changement comme avantage concurrentiel pour le client.
- La livraison s'applique à une application fonctionnelle, toutes les deux semaines à deux mois.
- Le métier et les développeurs doivent collaborer régulièrement au projet.
- Le projet doit impliquer des personnes motivées. Donnez-leur l'environnement et le soutien dont elles ont besoin et faites leur confiance quant au respect des objectifs.
- La méthode la plus efficace de transmettre l'information est une conversation en face à face.
- L'unité de mesure de la progression du projet est un logiciel fonctionnel (ce qui exclut de comptabiliser les fonctions non formellement achevées).
- Les processus agiles promeuvent un rythme de développement soutenable.
- Les processus agiles recommandent une attention continue à l'excellence technique et à la qualité de la conception.
- La simplicité et l'art de minimiser les tâches parasites, sont appliqués comme principes essentiels.
- Les équipes s'auto-organisent afin de faire émerger les meilleures architectures, spécifications et conceptions.
- À intervalle régulier, l'équipe réfléchit aux moyens de devenir plus efficace, puis accorde et ajuste son processus de travail en conséquence.

Une méthode qualifiée d'agile doit donc se composer d'un ensemble de pratiques instrumentant le cadre décrit par les 12 principes généraux agiles et en conséquence s'inscrire dans le respect des 4 valeurs fondamentales ayant inspiré le Manifeste agile.

Scrum



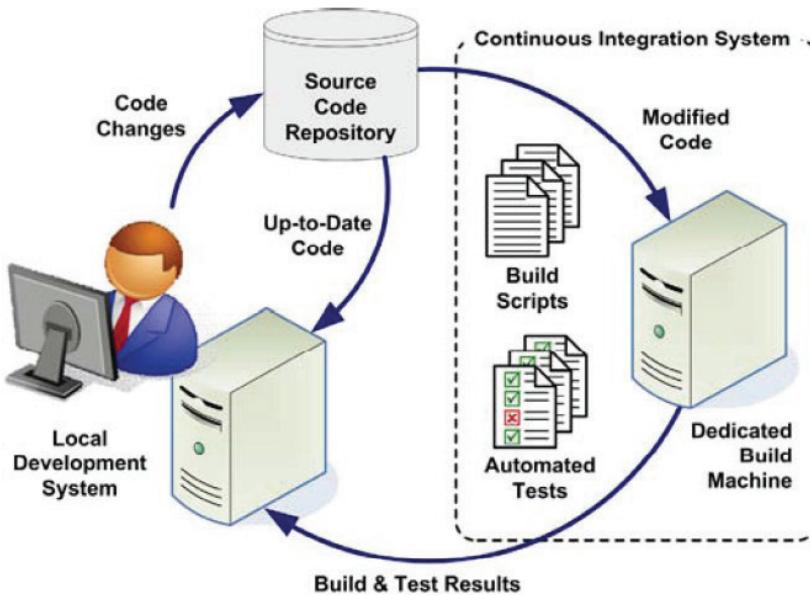
Scrum, une méthode de type agile, est basé sur la conviction que le développement logiciel est une activité par nature non-déterministe et que l'ensemble des activités de réalisation d'un projet complexe ne peut être anticipé et planifié longtemps à l'avance. Pour répondre à cette imprédictibilité, Scrum propose un modèle de contrôle de processus basé sur l'empirisme, via l'adaptation continue aux conditions réelles de l'activité et une réaction rapide aux changements. L'analyse des conditions réelles d'activité et le plan d'amélioration continue qui en découle sont réalisés à intervalle de temps régulier, donnant lieu à un cycle de développement itératif.

Scrum s'appuie sur le découpage d'un projet en boîtes de temps, nommés « sprints ». Les sprints durent, de préférence, environ deux semaines. Chaque sprint commence par une vérification de la planification opérationnelle. Le sprint se termine par une démonstration de ce qui a été achevé et contribue à augmenter la valeur du produit. Avant de démarrer un nouveau sprint, l'équipe réalise une rétrospective : elle analyse ce qui s'est passé durant ce sprint, afin de s'améliorer pour le prochain. L'adaptation et la réactivité de l'équipe de développement est facilitée par son auto-organisation.

Un principe fort des méthodes Agiles est la participation active du client, qui permet de choisir plus finement les fonctionnalités réalisées à chaque incrément. Le propriétaire du produit peut à tout moment compléter ou modifier la liste des fonctionnalités à produire pour les prochains sprints. Sans modifier le but du sprint en cours, celui-ci peut être affiné et faire l'objet d'une renégociation entre le propriétaire du produit et l'équipe de développement à la suite de nouvelles connaissances. Si le but du sprint devient obsolète, le propriétaire du produit a la capacité d'annuler un sprint en cours. Chaque sprint constitue donc un incrément, facilitant le pilotage du projet. La notion d'itération couvre l'adaptabilité au quotidien. Cette adaptabilité est limitée par le but immuable d'un sprint.

La méthode Scrum ne couvre volontairement aucune technique d'ingénierie du logiciel. Pour l'utiliser afin de développer une application, il est nécessaire de la compléter avec des pratiques de qualité du logiciel.

Intégration continue



L'intégration continue est un ensemble de pratiques utilisées en génie logiciel consistant à vérifier à chaque modification de code source que le résultat des modifications ne produit pas de régression dans l'application développée. En pratique, quand on parle d'intégration continue, on se réfère généralement à l'extreme programming (XP).

Pour appliquer cette technique, il faut remplir un certain nombre de conditions :

- code source partagé (logiciels de gestion de versions) ;
- les développeurs intègrent quotidiennement leurs modifications ;
- tests d'intégration développés pour valider l'application, de préférence automatisés
- outil d'intégration continue (p.ex. CruiseControl ou Jenkins pour Java).

Avantages

- test immédiat des unités modifiées
- prévention rapide en cas de code incompatible ou manquant
- problèmes d'intégration détectés et réparés de façon continue
- une version est toujours disponible (pour un test, une démonstration ou une distribution)

Pratiques

- Maintenir un dépôt unique de code source versionné
- Automatiser les compilations
- Rendre les compilations auto-testantes
- Tout le monde commit tous les jours
- Tout commit doit compiler le tronc (trunk) sur une machine d'intégration
- Maintenir une compilation courte
- Tester dans un environnement de production cloné
- Rendre disponible facilement le dernier exécutable
- Tout le monde doit voir ce qui se passe
- Automatiser le déploiement

eXtreme Programming



Extreme Programming (XP) est une méthode agile plus particulièrement orientée sur l'aspect **réalisation d'une application**, sans pour autant négliger l'aspect gestion de projet. XP est adapté aux **équipes réduites avec des besoins changeants**. XP pousse à l'extrême des principes simples.

Dans les méthodes traditionnelles, les besoins sont définis et souvent fixés au départ du projet informatique ce qui accroît les coûts ultérieurs de modifications. XP s'attache à rendre le projet plus flexible et ouvert au changement en introduisant des valeurs de base, des principes et des pratiques poussées à l'extrême :

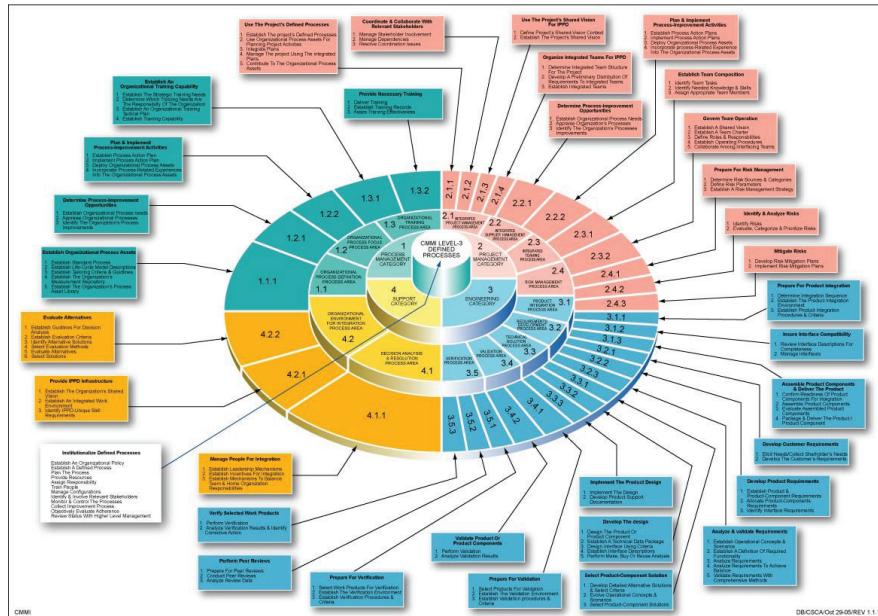
- puisque la revue de code est une bonne pratique, elle sera faite en permanence (par un binôme)
- puisque les tests sont utiles, ils seront faits systématiquement avant chaque mise en œuvre
- puisque la conception est importante, elle sera faite tout au long du projet (refactoring)
- puisque la simplicité permet d'avancer plus vite, nous choisirons toujours la solution la plus simple
- puisque la compréhension est importante, nous définirons et ferons évoluer ensemble des métaphores
- puisque l'intégration des modifications est cruciale, nous l'effectuerons plusieurs fois par jour
- puisque les besoins évoluent vite, nous ferons des cycles de développement très rapides

XP repose sur des cycles rapides de développement (itérations de quelques semaines) dont les étapes sont les suivantes :

- une phase d'exploration détermine les scénarios client qui seront fournis pendant cette itération
- l'équipe transforme les scénarios en tâches à réaliser et en tests fonctionnels
- chaque développeur s'attribue des tâches et les réalise **avec un binôme**
- lorsque tous les tests fonctionnels passent, le produit est livré

Le cycle se répète tant que le client peut fournir des scénarios à livrer. Généralement le cycle de la première livraison se caractérise par sa durée et le volume important de fonctionnalités embarquées. Après la première mise en production, les itérations peuvent devenir plus courtes (une semaine par exemple).

Documentation



La documentation fait partie du projet et est soumise à une acceptation. Et pourtant, elle est souvent négligée, alors que c'est un élément en soi de la démarche commune de gestion de projet. La documentation, contrairement à ce qu'on pourrait penser, est intimement liée à la phase de réalisation.

On distingue plusieurs types de documentation :

- Le guide d'utilisation, à destination des utilisateurs
- Le guide d'administration et de maintenance, à destination du département IT
- Le guide de développement, contenant e.a. les API
- La documentation technique détaillée, contenant e.a. :
 - Infrastructure
 - Architecture
 - Structure de la base de données
 - Web services
 - Détails de l'implantation des fonctions
 - Etc.

Il est conseillé de rédiger quotidiennement la documentation, simultanément avec le développement, la configuration et l'écriture des tests.

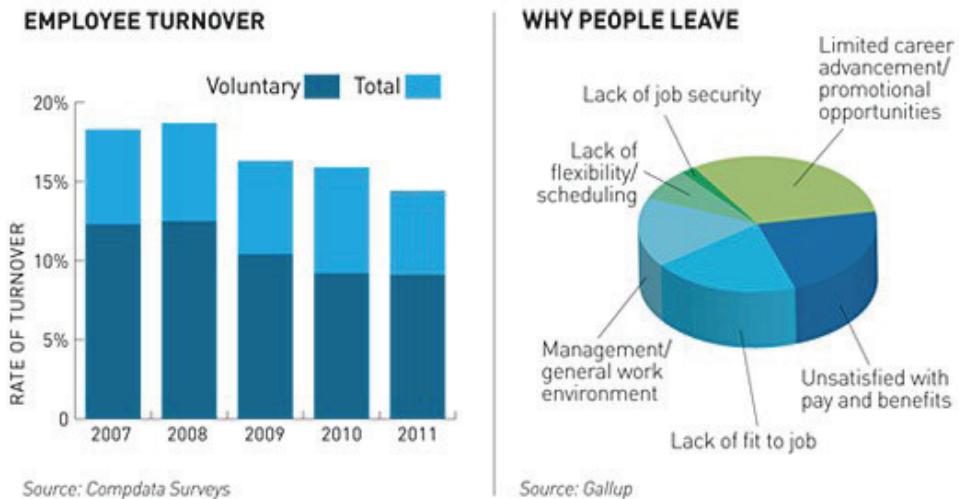
Éléments de gestion de projets

5. Pilotage

- 1) Suivi des ressources
- 2) Indicateurs
- 3) Calcul de rentabilité
- 4) Démarche qualité

Tout au long de sa vie, le pilotage permettra de comparer le réalisé avec le prévisionnel, éventuellement de réviser les plannings et les charges. Le pilotage permet d'assurer un suivi fiable du projet grâce à l'obtention d'une vue d'ensemble sur le projet, de mesurer précisément l'avancement du projet, de valider les dates jalons et de prendre les bonnes décisions en cas de difficulté.

Pilotage



Le pilotage efficace des ressources humaines et matérielles est indispensable à la réussite du projet.

Chacun à sa place

Les ressources humaines du projet sont l'ensemble des acteurs du projet. Elles sont des facteurs clés de succès du projet et doivent donc être particulièrement bien choisies et bien pilotées. Pour maximiser l'efficacité des ressources, elles doivent être affectées aux tâches où elles pourront le mieux faire valoir leurs compétences.

C'est la responsabilité du HRM, en collaboration avec les chefs de département,

- de veiller à ce que les compétences du personnel restent compétitives par rapport au marché ;
- de gérer les rôles de backup (« plan autobus ») ;
- de prévenir et résoudre les conflits entre personnes.

Le climat, l'ambiance de travail

Le climat général dans l'équipe projet joue un rôle majeur dans la motivation et l'implication des ressources du projet. Le chef de projet a un rôle clé concernant l'instauration d'un bon climat de travail, en adoptant une attitude positive et équitable vis à vis des membres de son équipe, en favorisant l'intégration de nouvelles ressources par des formations et un encadrement adapté. Beaucoup de projets s'accompagnent d'événements pour souder les équipes : restaurant, jeux, etc.

La (délicate) gestion des ressources humaines

Un des risques des projets de longue durée est le turnover des personnes travaillant sur le projet à cause des démissions, des départs en congé maternité ou en congé maladie prolongé, ... Pour limiter le turnover dû à la démotivation, les prévisions de charge de travail pour chaque tâche doivent être évaluées au plus juste afin d'éviter des surcharges trop fréquentes ou des plannings d'inactivité.

En cas de départ annoncé d'une ressource, il faut s'assurer du transfert des compétences avant son départ effectif, afin d'assurer un suivi et de ne pas mettre en risque le projet. D'une part, une phase de recouvrement avec le remplaçant doit être prévue, ou sinon avec une ressource intermédiaire qui formera plus tard le remplaçant. D'autre part, le démissionnaire consacrera une partie importante de son préavis à documenter son travail.

Pilotage

Project Name	Start	Finish	Work	% Work Comp	Remaining Work	Schedule KPI
Enterprise Project Type Name: Basic Project Plan	11/7/2012	8/15/2014	34,152h		14,800.43h	
Campus wireless upgrade	11/7/2012	11/7/2012	0h	0%	0h	○
Light Rail Phase 2 - Infrastructure Support Scoping	1/28/2013	4/1/2013	408h	0%	408h	●
System Architectural Analysis	1/7/2013	5/1/2013	380h	33%	254h	○
Architectural Analysis for Farmington	12/3/2012	5/13/2013	388h	10%	350h	●
Charleston Application Development API Standards	10/1/2013	1/9/2014	1,077h	0%	1,077h	●
Charleston Cell Phone Design	8/1/2013	1/16/2014	1,699h	32%	1,207.43h	○
Develop Rio Verde System	1/7/2013	2/18/2014	4,816h	84%	148h	●
Hillcrest Publishing System Design and Development	2/11/2013	3/24/2014	4,816h	74%	1,176h	●
Design and Develop Stock Analysis System	3/18/2013	4/29/2014	4,816h	71%	1,476h	●
Spectrum Auto CAD Design Tool	5/20/2013	7/3/2014	4,816h	63%	2,540h	●
Cantina Airlines Reservation System	6/3/2013	7/15/2014	4,816h	61%	2,756h	●
Heads-up Auto Display	6/6/2013	8/15/2014	6,120h	61%	3,408h	●

Le suivi des ressources humaines

Il s'agit de suivre au cours du projet le maintien de la correspondance entre les ressources et les besoins éventuellement ré-évalués.

Le pilotage consiste à suivre l'adéquation des prévisions à la réalité et éventuellement ré-évaluer les besoins en terme de ressources et les profils de compétence requis. Le suivi des ressources passe par la révision éventuelle du taux d'affectation de ressources. Par exemple, si une tâche s'avère plus longue que prévue initialement, une ressource non affectée pendant la période concernée peut alors l'être en renfort. Le but est toujours d'optimiser l'affectation des ressources.

Le « plan de charges », qui montre l'affectation des personnes en nombre de jours sur une tâche donnée, permet de présenter pour chaque mois combien de jour*homme ont été utilisés. Cet outil apporte une visibilité à une date donnée sur ce qui reste à faire, des révisions éventuelles par rapport au planning initial, un comparatif sur le réalisé par rapport au planifié. On évalue souvent les besoins - c'est à dire les ressources humaines nécessaires pour accomplir une tâche - en « jour-hommes ».

Exemple

La charge de travail nécessaire pour effectuer la tâche A s'élevant à 20 j.h correspond à un homme travaillant à temps plein sur vingt jours ou un homme à mi-temps pendant quarante jours ou encore deux hommes travaillant à temps plein pendant dix jours ou deux hommes à mi-temps pendant vingt jours.

Le suivi des ressources matérielles

Au cours du projet, les besoins en ressources matérielles peuvent évoluer et il faut pouvoir répondre rapidement à de nouveaux besoins et gérer les risques liés à d'éventuelles indisponibilités.

L'indisponibilité d'un environnement informatique est par exemple un facteur bloquant qu'il faut savoir gérer.

Indicateurs de pilotage

Tout projet implique la détermination d'indicateurs de pilotage du projet qui sont des outils de navigation et de décision. Ils permettent de mesurer une situation ou un risque, de donner une alerte ou au contraire de signifier l'avancement correct du projet. Le choix des indicateurs dépend des objectifs du projet.

Les indicateurs de pilotage peuvent être regroupés sous la forme d'un tableau de bord, véritable outil de gestion des responsables du projet. Les tableaux de bord sont aussi souvent nommés « scoreboards ».

Des indicateurs spécifiques au projet doivent être établis.

Exemples d'indicateurs

- Utilisation des ressources (en %)
 - Tâches réalisées / tâches planifiées
 - Jalons
 - Date de fin initiale
 - Date de fin finale
 - Avancement en délai (%)
 - Nombre de tâches terminées par rapport au nombre de tâches prévues
 - Nombre de changements
 - Nombre de risques réalisés

Calcul de rentabilité



Restons simples ...

Bénéfice du projet = Revenus du projet – Coûts du projet

Rentabilité = Bénéfice / Revenus (ou Profit / Coûts)

Revenus = Facturation forfaitaire + Facturation en régie

Facturation forfaitaire = somme prévue au devis

Facturation en régie = somme des Jours-hommes * Tarif journalier

Coûts = Manpower + Sous-traitance + Achats + Investissement + Frais généraux

Manpower = Somme des jours-hommes (cf time sheet) * Coût journalier

+ Déplacements (0.35 €/km)

+ Notes de frais (voyage, ...)

Sous-traitance : sur base des devis fournis par le sous-traitant (forfait ou régie)

Achats = somme des achats affectés au projet, sans amortissement

Investissement = somme des investissements affectés au projet, avec amortissement

= coût original total * fraction affectée au projet / nombre d'années d'amortissement

Frais généraux : soit sur base forfaitaire

soit par calcul analytique

La rentabilité ne sera évidemment connue qu'une fois le projet terminé et tous les coûts précisément connus. Mais on peut aussi, dans le cadre du pilotage du projet, calculer une « rentabilité intermédiaire » intégrant :

- Pour les tâches terminées : les coûts réels
- Pour les tâches restant à accomplir : les coûts budgétisés, éventuellement adaptés suite à l'expérience acquise en cours de projet

Démarche qualité



Les projets IT, souvent ceux liés à l'évolution ou à l'implémentation d'un système, sont la plupart du temps sujets à des dérives en terme de coûts et de durée, et recouvrent rarement le périmètre initialement défini. Certaines statistiques évoquent que 5 projets sur 6 n'arrivent pas à terme avec la qualité, le coût et/ou le délai prévus initialement.

La « Démarche Qualité » est nécessaire pour fiabiliser la gestion de projet, au moyen de différents outils de normalisation des méthodes de travail et de validation. L'adoption de cette démarche doit amener à une meilleure maîtrise des coûts et de la durée du projet.

La **démarche qualité** consiste à trouver l'adéquation entre la réponse aux besoins du projet, l'expression correcte de ces besoins par des spécifications adéquates qui passent par une écoute attentive du client, et une réalisation répondant à l'expression des besoins.

Si les spécifications sont conformes aux besoins mais que la réalisation ne répond pas aux spécifications et donc aux besoins, on parle de **défauts** dans la réalisation. Si la réalisation est conforme aux spécifications mais que ces dernières ont surévalué les besoins, on parle de **sur-qualité**.

Démarche qualité : le long du projet



Any project can be done...
Quick - Cheap - High Quality
Pick any 2 of the above.

Phases d'études préalables et de définition fonctionnelle du besoin

Lors des phases d'étude préalable et de définition fonctionnelle des besoins, la qualité se caractérise par la capacité des études produites à **adresser les véritables objectifs du projet** et de complètement **satisfaire les attentes** associées. Lors de ces phases, les contrôles qualité reposent d'une part, sur la vérification du respect des principes de la méthodologie mise en œuvre et d'autre part, sur des revues visant à garantir l'adéquation et la cohérence des solutions proposées avec les attentes entre les différentes phases.

Phases de planification

Il faut s'assurer que le réseau des tâches est **complet et exhaustif**, que le chemin critique et les risques sont bien identifiés. Il faut vérifier que les objectifs sont atteints en terme de délai, que les livrables du projet ont été bien identifiés.

Phases de réalisation

Le contrôle de la qualité porte d'une part, sur la **vérification de la méthodologie** (de développement) mise en œuvre et d'autre part, sur des **revues de code** permettant de garantir les performances et la maintenabilité des applications développées.

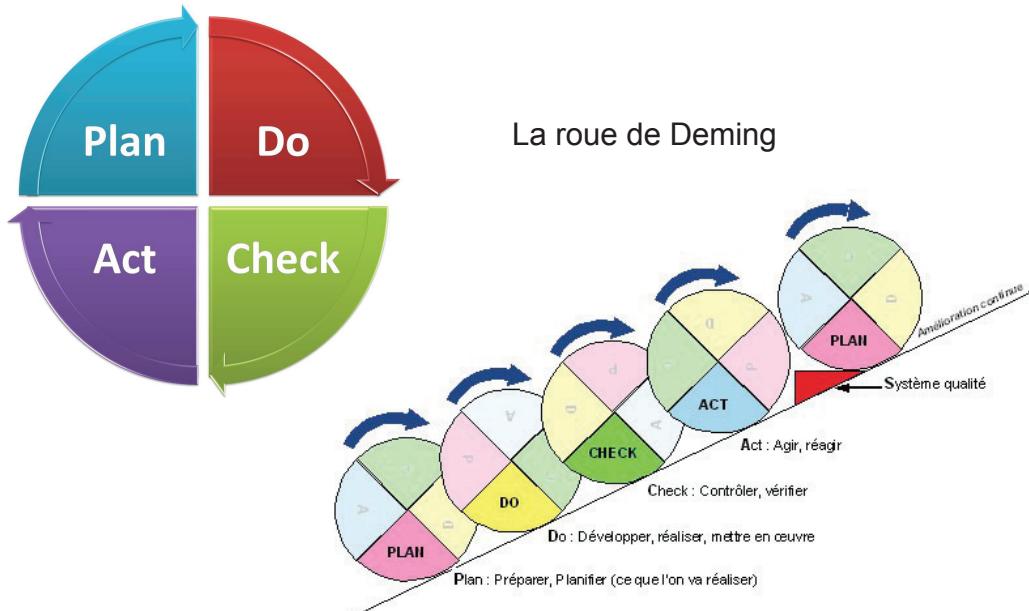
Phases de test

Les phases de test permettent de vérifier d'une part, le bon fonctionnement intrinsèque des applications livrées et d'autre part, l'adéquation entre les fonctions réalisées par ces applicatifs et les fonctions spécifiées dans les dossiers de définition du besoin. En conséquence, lors de ces phases, le contrôle de qualité va consister à **vérifier le bon fonctionnement de ce processus**.

Contrôle du cycle de traitement des anomalies

Les anomalies détectées lors des phases de test doivent donner lieu à une correction en terme de paramétrage ou développement. De nouveaux tests doivent alors avoir lieu dans le cadre de la démarche qualité, afin de **s'assurer que les corrections apportées fonctionnent**.

Démarche qualité : le cycle Dewing



Le célèbre cycle de Shewart-Deming est encore appelé le cycle Deming ou le cycle PDCA (Plan, Do, Check and Act). Au départ, dans les années 1920, le statisticien Shewart est à l'origine d'une première version du cycle : planifier, agir et voir le résultat. Ce cycle a été revu et amélioré par Edward Deming selon les modalités que nous allons voir.

Le cycle Deming montre comment appliquer les principes de la Démarche Qualité au processus projet dans son ensemble et aussi à ses tâches élémentaires. Il se compose de quatre phases:

- la planification ("Plan"): On y formalise les attentes, les moyens et les objectifs
- la réalisation ("Do"): Elle s'appuie sur les modes opératoires préalablement définis
- la vérification ("Check"): Elle doit, elle aussi, se conformer aux processus définis en amont
- l'action correctrice ("Act"): correction des erreurs constatées et mise en place de mesures pour leur prévention

Résultat : la réussite de cette démarche dépendra de la mobilisation de chacun pour une recherche permanente d'amélioration et des moyens mis en œuvre par les instances de décision pour l'application de la démarche.

Éléments de gestion de projets

6. Acceptance

- 1) Acceptances intermédiaires
- 2) Acceptance finale

Dès la mise à disposition ou la réception du livrable, il est nécessaire de procéder à des vérifications de manière à contrôler la conformité du résultat fabriqué avec la commande qui avait été passée lors des spécifications. Les contrôles s'effectuent sous forme de tests rigoureux à partir des cahiers de tests qui ont été préparés. À l'issue de la phase de recette est signé un procès-verbal de réception définitive. Les acceptances intermédiaires sont requises pour la sous-traitance.

Acceptance



L'acceptance est la phase pendant laquelle le MOA vérifie, à l'aide d'une batterie de tests documentés appelée « Acceptance Testing », que le système livré respecte les critères d'acceptance définis en début de projet. Ces tests prennent place après la réussite de tous les autres tests, notamment ceux purement techniques.

L'acceptance concerne certainement le système livré dans son état final, mais peut également concerner des composants, sous-systèmes ou deliverables intermédiaires. Ce sera le cas notamment pour valider le travail d'un sous-traitant.

L'acceptance couvre non seulement la livraison du système, mais également toute sa documentation.

Il y a deux conséquences au passage d'une acceptance par le MOA :

- Le système auquel elle réfère ne sera en principe plus modifié, qu'il s'agisse d'un système final ou intermédiaire.
- S'il s'agit du système final, la période de garantie commence ; pendant cette période, des corrections de malfaçons (bug) et certaines adaptations mineures peuvent être apportées au système.

Éléments de gestion de projets

7. Déploiement et Migration

Déploiement et Migration



Vu par l'utilisateur final, le **déploiement** est le passage d'un ancien à un nouveau système. Vu par les acteurs du projet, le **déploiement** est la mutation de l'application pilote vers l'application opérationnelle.

On considère deux dimensions au déploiement :

- Géographique : tous les sites passeront au nouveau système simultanément, ou région par région, ou par entité de l'organigramme, p.ex. siège central puis agences.
- Fonctionnelle : d'un bloc ou progressif

Il est plus simple et plus rapide de procéder au déploiement simultanément et d'un bloc, mais on court le risque de rencontrer de nombreux problèmes, voire un échec. Par contre, quand le déploiement se passe par phase, on court moins de risque mais il faut toujours prévoir une passerelle entre l'ancien et le nouveau système.

La **migration** est la mise en place des données nécessaires au démarrage du nouveau système, en veillant à limiter les désagréments aux utilisateurs : pas de perte des anciennes données, pas d'interruption de service, pas de perturbation du fonctionnement des autres systèmes.

La migration comprend un aspect technologique - changement de matériel, de logiciel – et un aspect fonctionnel – adaptation de la structure de données au nouveau système.

Éléments de gestion de projets

8. Communication et formation

Communication et formation



Différentes activités sont couvertes par la gestion du changement :

La **communication** veut réduire le risque d'un rejet du nouveau système par l'utilisateur concerné, qu'il soit interne (management, cadre, exécutants, Conseil d'Entreprise, ...) ou externe (client, fournisseur, ...). En fonction de la cible, on choisira un style de communication approprié.

La **formation** veut faciliter l'appropriation du nouveau système et des nouvelles procédures par les utilisateurs. Les modalités sont à définir en fonction du contexte humain et des ressources du projet et peuvent être de l'auto-formation, de la formation directe ou de la formation indirecte (formation de formateurs). Cf section « documentation » et le « guide de l'utilisateur ».

Éléments de gestion de projets

9. Période de Garantie

Tout ce qui concerne la vie du produit, une fois confronté à la réalité de son utilisation :

- évaluation du deliverable en "live"
- corrections *a posteriori*

Garantie, Project Review

Dear Helpdesk,
Thank you for creating my account so fast.
Though it is very funny, but how could I sell our products
with an email address as bitch@accorhotel.com?
Please change it ASAP.

Thank you in advance.
Bill Tchavlovsky

Dear Mr Tchavlovsky
Unfortunately account names are generated from your
initials, we cannot change it.
You can believe me.

Regards,
Ajani Erkson ajerk@accorhotel.com

Période de garantie

Après l'acceptance finale, la **période de garantie** commence pendant laquelle, bien qu'on ne modifie plus le système sur le plan fonctionnel, des corrections de malfaçons (bug) et certaines adaptations mineures (tuning) peuvent encore y être apportées.

Account management

La période de garantie ne doit pas forcément être considérée comme du « stand-by ». Une démarche pro-active par laquelle l'account manager rencontre le MOA après quelque temps permet à l'entreprise d'évaluer le deliverable du projet dans des conditions « live & long term ».

Post-project review (PPR)

C'est un débriefing mais à l'échelle du projet. La raison d'être du PPR est de tirer des enseignements du projet une fois achevé. En effet, il est important de partager ses expériences – notamment ses erreurs – avec ses collègues afin d'améliorer la qualité des projets délivrés par l'entreprise.

Notons qu'une entreprise soucieuse de s'améliorer peut appliquer d'autres méthodes : les rotations internes du personnel entre projets et la « knowledge base » où chacun décrit les problèmes qu'il a rencontrés et les solutions qu'il a apportées.