



UML et ingénierie de projet – vision MOA

Le calcul des charges et des délais

Hervé DOMALAIN

2010-2011



Sommaire

- ✓ **Préambule**
- ✓ Méthode des points de cas d'utilisation
 - Présentation de la méthode
 - Estimation de la taille du logiciel
 - Estimation de la charge
- ✓ Les délais et les coûts

Les différentes méthodes de calcul des charges

✓ Les méthodes algorithmiques :

- La taille du logiciel est calculée à partir de la comptabilisation des caractéristiques quantitatives (nombre de cas d'utilisation, de classes ou de lignes de code...) de ce logiciel.
- Des algorithmes de calcul définissent la relation entre la taille du logiciel et la charge du projet. Ils ont été élaborés en étudiant un nombre significatif de projets terminés par des organismes divers.

✓ Les méthodes analytiques :

- Un projet est composé de phases, elles-mêmes composées de sous-phases, etc. Chaque étape du projet fait l'objet d'un ou plusieurs livrables (documentation, maquette, prototype, code...).
- L'évaluation de la charge de réalisation de ces livrables permet d'estimer la charge du projet.

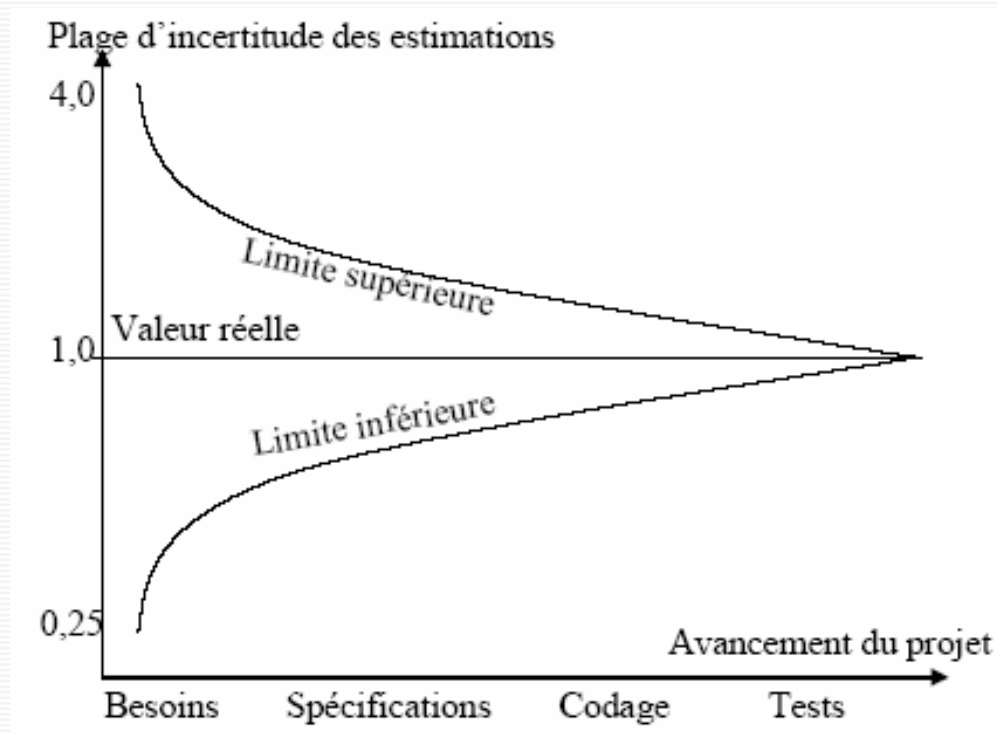
Conséquences de la sous-estimation et de la surestimation

- ✓ La planification rigoureuse et le pilotage du projet ne sont pas envisageables en absence d'une estimation sérieuse et fiable.
- ✓ La sous-estimation d'un projet entraîne :
 - Un sous-effectif, provoquant la « surchauffe »,
 - Une appréciation insuffisante de la charge d'assurance qualité, avec le risque de livrables de médiocre qualité,
 - L'établissement d'un planning trop serré décrédibilise le chef de projet lorsque les délais sont largement dépassés.
- ✓ La surestimation d'un projet peut s'avérer aussi néfaste si vous accordez à un projet plus de ressources que nécessaires :
 - Coûtera beaucoup plus cher (en grevant le budget du projet),
 - Durera plus longtemps que prévu (non respect des délais le cas échéant contractualisés),
 - Diffèrera la disponibilité des ressources pour le projet suivant.

Exactitude et précision des estimations (1/2)

- ✓ Qu'est-ce qu'une estimation exacte ? L'**exactitude** caractérise l'approche de la réalité, alors que la **précision** caractérise la finesse avec laquelle une grandeur est mesurée.
 - Par exemple, une estimation de taille de 70 ou 80 kilo-instructions peut à la fois être la plus **exacte** et la plus **précise** que l'on puisse faire à la fin de la phase. Si l'on simplifie cette estimation en annonçant 75 000 instructions, celle-ci **semble plus précise**. Mais en réalité elle est **moins exacte**. Vous pouvez également annoncer 75 281 avec la **précision** d'une instruction. Mais on ne pourra mesurer cette taille qu'à la fin de la phase de codage du projet, après décompte des instructions.
- ✓ Si l'on donne l'estimation de la taille sous forme d'une plage (intervalle entre une valeur minimale et une valeur maximale) plutôt qu'avec une simple valeur, toutes les valeurs ultérieurement calculées (charges, délais, coûts) seront également représentées par des plages.

Exactitude et précision des estimations (2/2)



Graphique de convergence des estimations « développement rapide » (Mc CONNELL - 1996) adapté des modèles de coûts pour le cycle de vie (BOEHM - 1995)

Recommandations pour une bonne évaluation (1/2)

- ✓ Utiliser au minimum deux méthodes d'évaluation :
 - Au moins une méthode algorithmique telle que la méthode des points de cas d'utilisation par exemple,
 - Une méthode analytique reposant sur une ventilation détaillée du travail à exécuter en tâches ou livrables et une agrégation des charges de chaque tâche ou livrable (chaque tâche ou livrable est estimé par analogie avec des éléments similaires ou selon un modèle mathématique),
 - Une approche analogique si l'entreprise possède une base de connaissance fiable (capitalisation systématique et nombre de projets suffisant),
- ✓ Itérer l'évaluation à chaque phase de développement afin de l'affiner au fur et à mesure de l'avancement dans le cycle de développement.

Recommandations pour une bonne évaluation (2/2)

- ✓ Faire apparaître les risques d'erreur, à savoir fournir, plutôt qu'une valeur, un intervalle indiquant une valeur mini (ou optimiste) et une valeur maxi (ou pessimiste).
- ✓ L'objectif en début de projet, pendant la consultation ou pendant la phase de spécification, est d'aboutir à une évaluation la plus fiable possible car elle engage l'entreprise.
- ✓ L'idéal est de déterminer un risque d'erreur ou écart maximum toléré ($< 20\%$ pour l'évaluation initiale et $< 10\%$ ensuite).



Sommaire

- ✓ Préambule
- ✓ Méthode des points de cas d'utilisation
 - Présentation de la méthode
 - Estimation de la taille du logiciel
 - Estimation de la charge
- ✓ Les délais et les coûts

Une méthode basée sur les spécifications fonctionnelles (1/2)

- ✓ La méthode des points de cas d'utilisation (UCP) est dérivée de la méthode des points de fonction publiée en 1979 par Alan Albrecht (l'International Function Points User Group (IFPUG) fondée en 1986 a participé à la normalisation ISO de cette méthode).
- ✓ La méthode des points de cas d'utilisation est basée sur une métrique indépendante de la technique de développement afin de s'affranchir des inconvénients du comptage du nombre de lignes de code.
- ✓ C'est une méthode, définie en 1993 par Gustave Karner, qui évalue la taille d'un logiciel et estime les charges correspondantes en se basant sur une spécification de cas d'utilisation.

Une méthode basée sur les spécifications fonctionnelles (2/2)

- ✓ Le nombre de points de cas d'utilisation dans un projet est fonction des éléments suivants :
 - Le nombre et la complexité des cas d'utilisation du système,
 - Le nombre et la complexité des acteurs du système,
 - Différentes exigences non techniques (telles que la portabilité, la performance, la maintenabilité...) qui ne sont pas exprimées dans les cas d'utilisation,
 - L'environnement dans lequel le projet sera développé (tel que le langage, la motivation de l'équipe, etc.).

Les étapes de l'estimation des charges avec la méthode UCP

1. Détermination du poids des acteurs non ajusté (UAW),
2. Détermination du poids des cas d'utilisation non ajusté (UUCW),
3. Calcul des points de cas d'utilisation non ajustés (UUCP),
4. Evaluation des facteurs de complexité technique (TCF),
5. Evaluation des facteurs environnementaux (EF),
6. Calcul des points de cas d'utilisation ajustés (AUCP),
7. Estimation des charges.



Sommaire

- ✓ Préambule
- ✓ Méthode des points de cas d'utilisation
 - Présentation de la méthode
 - Estimation de la taille du logiciel
 - Estimation de la charge
- ✓ Les délais et les coûts

Détermination du poids des acteurs non ajusté (UAW)

Classification	Métriques	Valeurs
Acteur simple	Les acteurs simples communiquent avec le système au travers d'API. Il s'agit en général d'autres systèmes.	1
Acteur moyen	Les acteurs moyens sont des acteurs qui interagissent avec le système au travers de protocoles (FTP, HTTP...). Il s'agit en général d'autres systèmes.	2
Acteur complexe	Les acteurs complexes sont des acteurs humains communiquant avec le système à l'aide d'IHM.	3

Détermination du poids des cas d'utilisation non ajusté (UUCW)

Classification	Métriques	Valeurs
CU simple	Les CU simples présentent de 2 à 3 transactions.	5
CU moyen	Les CU moyens présentent de 4 à 7 transactions.	10
CU complexe	Les CU complexes présentent de 8 à 12 transactions.	15

La fiabilité de l'estimation dépend de la complétude et de la précision de la spécification fonctionnelle via les cas d'utilisation. Il est également fondamental que tous les cas d'utilisation soient décrits avec le même niveau de détail.

Définition de la notion de transaction

- ✓ Un cas d'utilisation est composé de scénarios (un scénario nominal et des scénarios secondaires dits alternatifs ou d'exception).
- ✓ Chaque scénario est constitué de transactions.
- ✓ Chaque transaction consiste en 4 étapes :
 1. L'acteur envoie une requête et des données au système,
 2. Le système valide la requête et les données,
 3. Le système change son état interne,
 4. Le système répond à l'acteur avec le résultat de sa requête.

Exemple de décompte des transactions

Scénario principal :

1. L'utilisateur choisit le type d'article et saisit la quantité d'articles voulus.

1

2. Le système affiche les choix de l'utilisateur et propose une liste de fournisseurs.

2

3. L'utilisateur sélectionne le fournisseur dans la liste et valide sa commande.

4. Le système valide la commande.

Scénarios secondaires :

1a. Il n'y a pas suffisamment d'articles :

1a1. Le système indique à l'utilisateur le nombre d'articles disponibles.

1a2. L'utilisateur saisit la nouvelle quantité.

Le scénario reprend à l'étape 2.

3

3a. Le fournisseur n'existe pas dans le système :

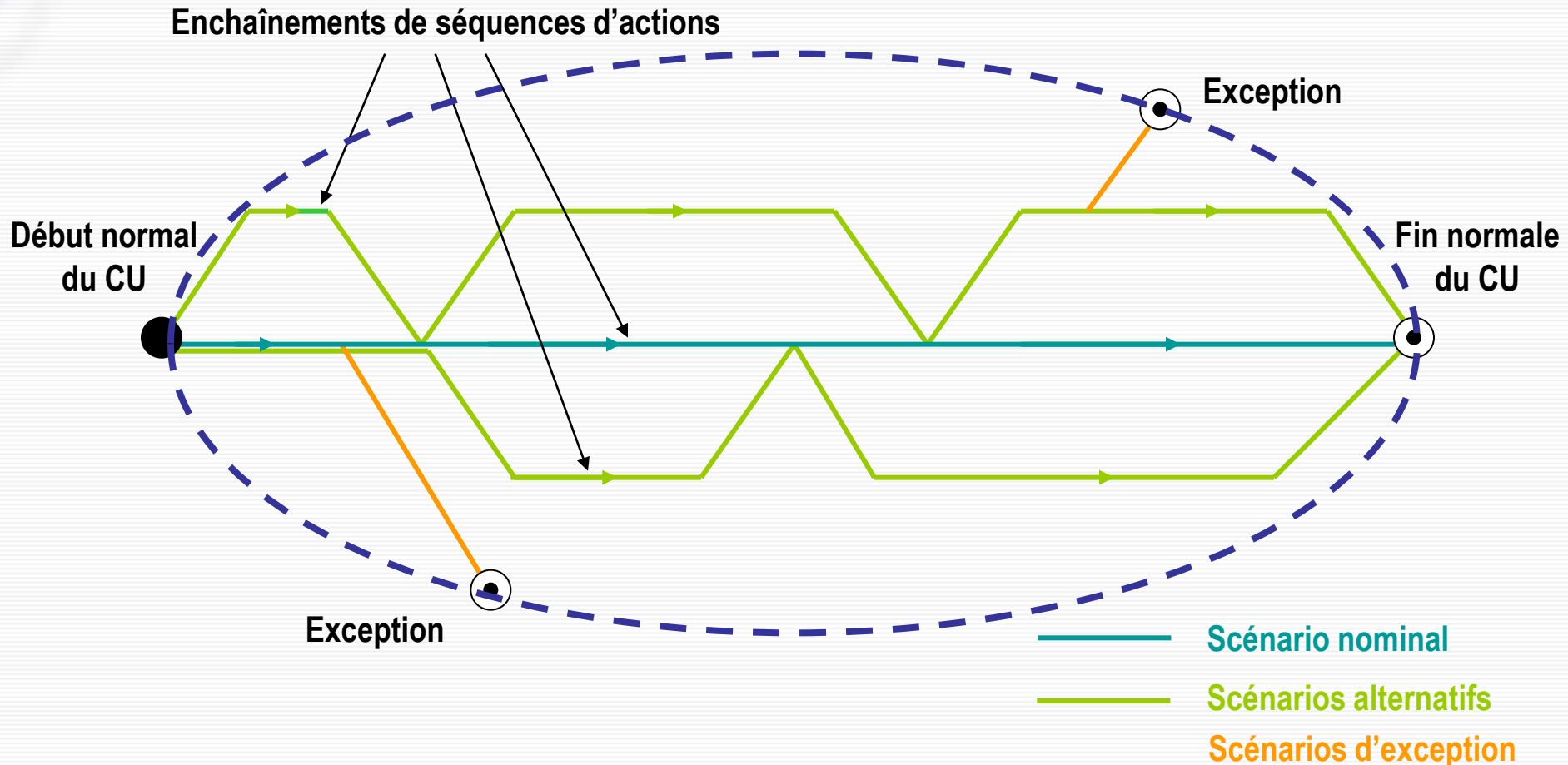
3a1. Le système propose la saisie d'un nouveau fournisseur.

3a2. L'utilisateur saisit le nom, l'adresse et les coordonnées du nouveau fournisseur.

Le scénario reprend à l'étape 4.

4

Cas d'utilisation et scénarios



Calcul des points de cas d'utilisation non ajustés (UUCP)

$$UUCP = \sum_{i=1}^n UAW + \sum_{j=1}^m UUCW$$

Exemple :

$$UUCP = [(4 \times 1) + (1 \times 3)] + [(15 \times 5) + (2 \times 10) + (3 \times 15)]$$

$$UUCP = [4 + 3] + [75 + 20 + 45]$$

$$UUCP = 147$$

Pour une application présentant 4 acteurs simples, 1 acteur complexe, 15 CU simples, 2 CU moyens et 3 CU complexes.



Sommaire

- ✓ Préambule
- ✓ Méthode des points de cas d'utilisation
 - Présentation de la méthode
 - Estimation de la taille du logiciel
 - Estimation de la charge
- ✓ Les délais et les coûts

Evaluation des facteurs de complexité technique (TCF) (1/3)

	Facteurs techniques	Poids	Descriptions
t1	Système distribué	2	Le système présente t-il une architecture distribuée ou centralisée ?
t2	Objectif de performance	2	Le client a-t-il besoin d'un système rapide ? Le temps de réponse est-il un critère majeur ?
t3	Efficacité pour l'utilisateur final	1	Quelle plus-value doit apporter le système à l'utilisateur final dans le cadre de son travail ?
t4	Complexité des processus internes	1	Y a-t-il des processus métier impliquant des algorithmes complexes ?
t5	Réutilisabilité du code	1	Le code doit-il être réutilisable pour une autre application (ce qui implique un haut niveau d'abstraction) ?
t6	Facilité d'installation	0,5	Le client recherche t-il une facilité d'installation (installation personnalisable et automatisée, mise à jour automatique quelle que soit la version...) ?
t7	Facilité d'utilisation	0,5	L'ergonomie est-elle une préoccupation prioritaire ?

Evaluation des facteurs de complexité technique (TCF) (2/3)

	Facteurs techniques	Poids	Descriptions
t8	Portabilité	2	Le client souhaite t-il une implémentation multi plate-forme ?
t9	Evolutivité	1	Le client cherche t-il une application facilement modifiable pour répondre aux futures évolutions ?
t10	Concurrence d'accès	1	Le client veut-il gérer la concurrence d'accès pour de nombreux utilisateurs (ce qui complexifie la gestion des sessions) utilisateurs) ?
t11	Contraintes de sécurité	1	Le client cherche t-il un système hautement sécurisé (authentification, confidentialité, intégrité) ?
t12	Accès à des tiers	1	Convient-il de prévoir l'accès aux données (via des services par exemple) pour des applications tiers ?
t13	Besoins de formation	1	Faut-il mettre en place une plate forme spécifique de formation pour les utilisateurs ?

Evaluation des facteurs de complexité technique (TCF) (3/3)

$$TCF = 0,6 + (0,01 \times \sum_{t=1}^{13} < poidsFacteurComplexitéTechnique > \times < influenceFacteurComplexitéTechnique >)$$

Où le poids des facteurs de complexité technique est fixé dans les 2 tableaux précédents et où l'influence des facteurs de complexité technique est une valeur à choisir entre 0 et 5 :

0 inexistante ou sans influence - 1 influence secondaire -
2 influence restreinte - 3 influence moyenne -
4 influence importante - 5 influence intensive partout

Evaluation des facteurs environnementaux (EF) (1/2)

	Facteurs env.	Poids	Descriptions
e1	Familiarité avec le processus de développement	1,5	Quelle est le niveau de maîtrise de la méthode de développement choisie ?
e2	Expérience dans le domaine applicatif	0,5	Les personnes travaillant sur le projet connaissent-elles le domaine et les détails techniques du projet ?
e3	Expérience en orienté objet	1	Si le projet est orienté objet, il est important de posséder une expérience en conception et en développement objet.
e4	Capacité à conduire l'analyse	0,5	Qu'elle est l'expérience de l'analyste du projet ?
e5	Motivation	1	Les programmeurs sont-ils motivés pour travailler sur le projet ? Un marché concurrentiel favorise la motivation des SSII.
e6	Stabilité des exigences	2	Le client sait-il ce qu'il veut ? Les exigences des utilisateurs constituent le facteur le plus important. Si le client est de nature changeante, mettre cette valeur au minimum.

Evaluation des facteurs environnementaux (EF) (2/2)

	Facteurs env.	Poids	Descriptions
e7	Travail à temps partiel	-1	Y a-t-il du personnel à temps partiel dans le projet, du turn-over ?
e8	Difficulté du langage de programmation	-1	Quel est le niveau de complexité du langage utilisé ?

$$EF = 1,4 + (-0,03 \times \sum_{e=1}^8 < poidsFacteurEnvironnemental > \times < influenceFacteurEnvironnemental >)$$

Où le poids des facteurs environnementaux est fixé dans les 2 tableaux précédents et où l'influence des facteurs environnementaux est une valeur à choisir entre 0 et 5 :

0 inexistante ou sans influence - 1 influence secondaire - 2 influence restreinte –
3 influence moyenne - 4 influence importante - 5 influence intensive partout

Calcul des points de cas d'utilisation ajustés (AUCP)

Le nombre de points de cas d'utilisation ajustés correspond au nombre de points de cas d'utilisation non ajustés pondérés par les facteurs de complexité technique et environnementaux.

$$AUCP = UUCP \times TCF \times EF$$

Evaluation de la charge (1/3)

- ✓ Karner propose 20 h.h (heures.hommes) par AUCP.
- ✓ Selon son expérience, Sharks pense que l'effort peut s'étendre de 15 à 30 h.h par AUCP.
- ✓ Schneider et Winters suggèrent que la charge dépend des facteurs environnementaux :
 1. Le nombre de facteurs de e1 à e6 dont l'influence est inférieure à 3 et le nombre de facteurs pour e7 et e8 dont l'influence est supérieure à 3 sont comptabilisés.
 2. Si le total est inférieur ou égal à 2, on compte 20 h.h par AUCP,
 3. Si le total est de 3 ou 4, on prend 28 h.h par AUCP,
 4. Si le total excède 5, on recommande habituellement des changements organisationnels car le risque est jugé beaucoup trop important. Une autre possibilité consiste à passer à 36 h.h par AUCP.

Evaluation de la charge (2/3)

- ✓ Le mode de calcul de la charge proposé par Schneider et Winters paraît le plus approprié.
- ✓ La charge calculée concerne l'ensemble du projet, à savoir :
 - Le pilotage dudit projet (20%) regroupant la gestion des réunions du projet, la gestion des contrats, la gestion des ressources...
 - Le développement du logiciel (80%) dont le détail est donné sur la diapositive suivante.
- ✓ Elle ne concerne pas les charges qui relèvent de la MOA (information et formation des utilisateurs, préparation du déploiement, expérimentation en sites pilotes).
- ✓ 1 année.homme $\approx$ 40 semaines.hommes
 $\approx$ 200 jours.hommes $\approx$ 1400 heures.hommes

Evaluation de la charge (3/3)

- ✓ Les charges de développement sont-elles mêmes ventilées de la manière suivante :
 - $\approx 5\%$ - étude préalable
 - $\approx 15\%$ - analyse et conception fonctionnelles
 - $\approx 20\%$ - conception technique
 - $\approx 50\%$ - réalisation et tests
 - $\approx 10\%$ - recette

Conclusions sur la méthode UCP

- ✓ Cette méthode permet d'évaluer la charge d'un projet à partir d'une spécification fonctionnelle. Elle peut donc être utilisée dans les phases amont d'un projet.
- ✓ La fiabilité de l'évaluation dépend de la qualité et de l'homogénéité de la spécification à l'aide des cas d'utilisation.
- ✓ Une fois la charge globale calculée, il est possible d'affiner l'évaluation en s'appuyant sur une autre méthode algorithmique telle que COCOMO II. En effet, COCOMO II permet de calculer le délai de réalisation (à partir de la charge) et de ventiler la charge et le délai dans les différentes phase du projet.



Sommaire

- ✓ Préambule
- ✓ Méthode des points de cas d'utilisation
 - Présentation de la méthode
 - Estimation de la taille du logiciel
 - Estimation de la charge
- ✓ Les délais et les coûts

Estimation des délais

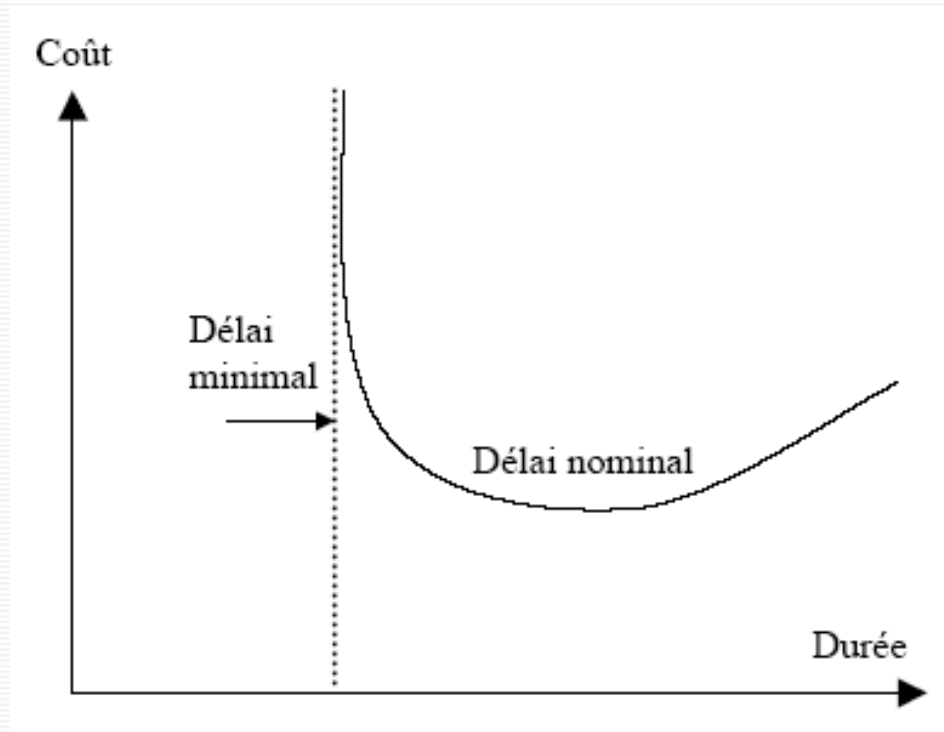
- ✓ Pour déterminer les délais, il faut généralement estimer les ressources affectées au projet (la structure de contribution), ce qu'elles devront faire (le WBS ou Work Breakdown Structure, c'est à dire l'organigramme des tâches), quand elles s'investiront dans le projet et quand elles le termineront. Puis il faut planifier les tâches.
- ✓ L'historique de la société ou, à défaut, des modèles classiques, peuvent être utilisés pour déterminer le nombre de personnes dont on a besoin.
- ✓ La formule empirique de Mc CONNELL (1996) donne une idée du temps total requis :

$$\text{Délai en mois} = 3 \times (\text{charge en mois.homme})^{1/3}$$

Gestion des délais (1/4)

- ✓ En tant que chef de projet, il faut trouver un compromis sur les fonctionnalités, les délais, les coûts et les effectifs, qui puisse être accepté à la fois par le directeur de projets et par le client !
- ✓ Choix stratégiques :
 - En allongeant les délais, on peut généralement réduire le coût global du projet en utilisant moins de personnes ou des personnes au coût unitaire moins important.
 - A l'inverse, il n'y a que trois décisions possibles pour réduire les délais :
 - Diminuer le périmètre fonctionnel,
 - Augmenter les effectifs, lorsqu'il existe des tâches que l'on peut paralléliser,
 - Faire travailler l'effectif en dépassement d'horaires.

Gestion des délais (2/4)



Relation entre coût et durée sur un projet logiciel « développement rapide » (Mc CONNELL - 1996). Le coût de réalisation selon une durée « normale » est très inférieur au coût de réalisation « à marche forcée ».

Gestion des délais (3/4)

S'il s'avère impossible de réduire le nombre de fonctionnalités, choisir l'une des deux autres possibilités peut grever le budget prévisionnel et augmenter les risques d'échec du projet :

- Ajouter des ressources peut accroître considérablement la quantité de travail car il faut gérer les communications supplémentaires et renforcer l'encadrement.
- Le recours aux heures supplémentaires est bénéfique à court terme, mais la productivité peut décroître à long terme, car les développeurs se fatiguent et commettent plus d'erreurs.

Gestion des délais (4/4)

- ✓ Les calculs de charges effectués avec la méthode des points de cas d'utilisation correspondent à des délais nominaux.
- ✓ A périmètre fonctionnel égal, on peut accepter une réduction de 25% du délai nominal. Au-delà, le projet n'est pas réalisable sans perte de qualité des livrables.
- ✓ 25% de délai en moins $\Rightarrow$ 40 % de charge en plus selon la méthode COCOMO.

Estimation des coûts (1/3)

- ✓ Il faut prendre en compte de nombreux facteurs pour estimer le coût total d'un projet :
 - Les charges de travail,
 - Le coût de la sous-traitance,
 - Les acquisitions (ou les locations) de matériels et de logiciels,
 - Les frais de déplacements (réunions, ...),
 - Les télécommunications (appels à longue distance, vidéoconférences, lignes dédiées aux tests, etc ...),
 - Les formations,
 - Les frais de structure, ...
- ✓ L'estimation exacte du coût total du projet dépend de la façon dont la société affecte les coûts. Au lieu d'être affectés aux projets, certains coûts peuvent être pris en compte en les intégrant dans des taux horaires forfaitaires (en Euros par heure).

Estimation des coûts (2/3)

- ✓ Souvent, un chef de projet estimera seulement le coût du travail et n'identifiera que les coûts additionnels qui ne sont pas considérés comme des frais généraux.
- ✓ Le coût du travail peut également être obtenu en multipliant l'estimation des charges en heure.homme par un taux horaire en Euros. De plus, un calcul plus précis peut résulter de l'utilisation d'un taux différent pour chaque catégorie de personnel (technicien, qualicien, encadrement, documentation, support, etc ...). Il faut alors déterminer quel pourcentage de la charge totale du projet doit être affecté à chaque profil. Une fois encore, les données historisées ou des modèles classiques peuvent apporter une aide.

Estimation des coûts (3/3)

- ✓ Souvent, la date de livraison est imposée. Si l'on connaît dès le départ la date de mise en œuvre du produit, la seule chose que l'on peut négocier est le champ des fonctionnalités à développer dans le temps imparti.
- ✓ Lorsque le temps imparti ne permet pas de tout faire, les fonctionnalités doivent être classées par priorité décroissante et regroupées en ensembles homogènes qui peuvent être développés en temps voulu.
- ✓ L'estimation à partir des délais impartis n'entraîne pas l'abandon des étapes du processus d'estimation indiqué ci-avant. Il convient toujours de définir la taille de l'ouvrage, de l'« éclater » en diverses parties que l'on peut soit sélectionner, soit soustraire de la livraison, d'estimer les charges, les délais et les coûts.
- ✓ Dans ce contexte, les outils d'estimation automatisés peuvent être réellement utiles afin de dérouler de nombreuses simulations, ce qui est beaucoup trop fastidieux manuellement.