



Estimation des charges

« Le travail se dilate jusqu'à remplir le temps disponible »

Le savoir faire du chef de projet

- Pas d'excès d'optimisme
- Retour sur expérience
- Toujours garder de la marge
- Identifier les points à risque
- Ne pas se limiter à une vision mono phase

Notion : Unité de charge

- Charge :
 - quantité de travail exprimée en ressources*temps
- Temps :
 - mois pour les grands projets
 - Jour pour les projets de taille plus réduite
- Remarque : ne pas confondre charge et durée

Unité de charge corrigée

- But : calculer la charge réelle d'une ressource
- Exemple :
 - Personne travaillant à plein temps sur le projet = 260j ouvrable (52*5)
 - 12j fériés, 30j de congés, 3j de maladie, formation 4j, réunions 6j = 55j
 - Coeff de correction = $260 / (260 - 55) = 1.26$

Les méthodes

- Méthode du « marché »
 - Méthodes Analytiques
 - Les points de fonction
 - COCOMO
 - Méthodes Simulatrices
 - Prototypages
 - Méthodes analogiques
 - experts
- La répartition proportionnelle

Méthode du marché

- La plus mauvaise car la plus risquée
- La charge correspond au prix proposé pour remporter l'appel d'offre
- Peut être utilisée s'il y a possibilité de compenser la perte occasionnée (ex: TMA)

Méthode analogique

- Méthode Expert
- Séries d'échanges entre différents experts qui estiment la charge probable du projet par rapport à leur expérience
- Méthode rapide mais demandant des personnes possédant beaucoup d'expérience
- $C_{moyenne} = (C_{optimiste} + C_{pessimiste} + 4 * C_{probable}) / 6$
- Méthode dite « à la louche »

La Méthode COCOMO (1)

- Deux hypothèses :
 - Un informaticien évalue mieux la taille du logiciel à développer que la quantité de travail nécessaire.
 - Il faut toujours le même effort pour écrire un nombre donné de lignes de programme, quel que soit le langage.
- Variable fondamentale :
 - Kilo Instructions Sources Livrées (KISL)
 - Kilo Line of code (KLOC)
 - Un tableau de référence permet d'obtenir la relation KISL/Charge en fonction du type de langage utilisé
- 3 catégories de projets :
 - dont la spécification est parfaitement définie (<50KLOC)
 - algorithmiques résolvant une classe de problèmes (<300KLOC)
 - réagissant à des stimuli de l'environnement (>300KLOC)

La Méthode COCOMO (2)

COCOMO(81) types de modèles :

- Le modèle de base : estimation de l'effort en fonction du nombre de KLOC, la productivité et un facteur d'échelle qui dépend du type de projet
- Le modèle intermédiaire : ajout de 15 facteur de productivité
- Le modèle détaillé : prise en compte de chacune des étapes du cycle du projet + prise en compte de la hiérarchisation du projet

■ COCOMO(00) type de modèles :

- Modèle de composition d'application (développement via des outils graphique)
- Modèle avant projet : estimation approximative avant définition de l'architecture définitive
- Modèle post-architecture : le plus détaillé, utiliser après le développement de l'architecture générale du projet

La Méthode COCOMO (3)

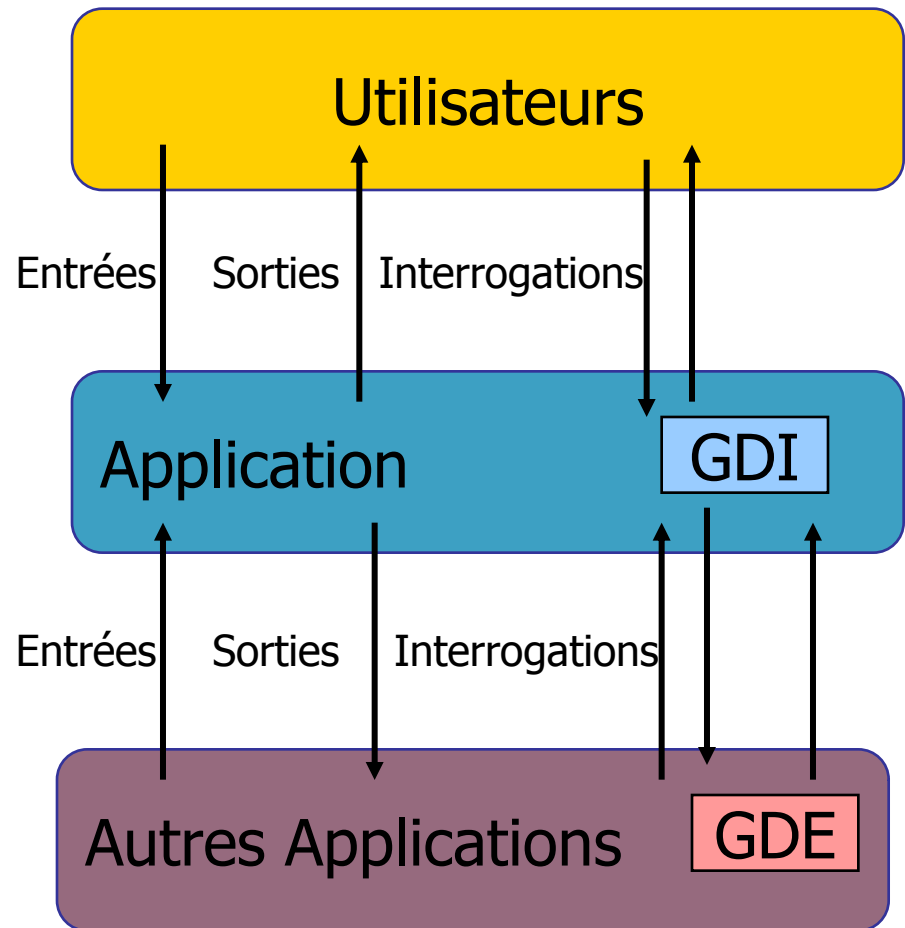
- 9 étapes :
 - Détermination de la taille du logiciel à livrer,
 - Détermination du mode de développement,
 - Calcul de la charge brute,
 - Détermination de la valeur de quinze facteurs correctifs, regroupé en 4 catégories :
 - Attributs du produit
 - Attributs de l'ordinateur
 - Attributs du personnel
 - Attributs du projet
 - Calcul de la charge nette en homme-mois,
 - Répartition de la charge par activité,
 - Répartition des charges par phases,
 - Répartition des délais par phase.

Les points de fonction (1)

- Méthode orienté vision fonctionnelle
- Version la plus connue : IFPUG
- 5 étapes
 - Définir le contexte : périmètre, phase de chiffrage, type de projet, les acteurs
 - Identifier les composants: les données manipulées (internes, externe) , les fonctions (cas d'utilisation) et leur type (entrée, sortie, interrogation)
 - Calculer pour chaque composant le nombre de points de fonction associé
 - Evaluer les facteurs d'ajustements
 - Calculer le nombre de PF net

Les points de fonctions (2)

- Vision utilisateur
- Composants :
 - Groupe logique de données internes
 - Groupe logique de données externes
 - Les entrées
 - Les sorties
 - Les interrogations



Les points de fonction (2)

- Sont associés à ces cinq entités 2 paramètres supplémentaires :
 - Données Élémentaire (GDI et GDE)
 - Une DE = un champ de données
 - Sous-ensemble Logique de Données (GDI/GDE)
 - Groupements logiques de GDI ou GDE traitées simultanément
 - Groupes de données référencées (ENT/SOR/INT)
 - Groupements logiques de GDI ou GDE mis à jour ou consultés simultanément par les différents ENT, SOR ou INT.

Les points de fonction (3)

- GDR = Groupe de données référencées
- Calcul des Points de fonction Bruts (exemple fonctionnalité d'entrée)

| ENT | | 1 à 4 DE | 5 à 15 DE | >15 DE |
|---------|------------|----------|-----------|--------|
| | 0 ou 1 GDR | 3 PdF | 3 PdF | 4 PdF |
| 2 GDR | 3 PdF | 4 PdF | 6 PdF | |
| > 2 GDR | 4 PdF | 6 PdF | 6 PdF | |

(ex : performance, portabilité)

- En moyenne 1PdF = 3J/h
- Une méthode simplifiée qui ne passe pas par l'estimation du nombre de DE utilisés existe. Il s'agit alors de prendre la valeur moyenne dans tous les cas. Elle peut être utilisée pour les macro-chiffrages
- Il existe également des tableaux qui permettent d'obtenir le nombre de lignes de code à partir des PdF (pour les adeptes de COCOMO)

Répartition Proportionnelle

- Basé sur l'estimation précise d'une des phases

| Etape | Ratio |
|---------------------------|--------------------------------------|
| Etude préalable | 10% du projet (- mise en œuvre) |
| Etude détaillée | 20 à 30% du total du projet |
| Etude technique | 5 à 15% de la charge de réalisation |
| Réalisation | 2 fois la charge d'étude détaillée |
| Mise en oeuvre | 30 à 40% de la charge de réalisation |
| Encadrement du projet | 15% de la charge de réalisation |
| Recette | 20% de la charge de réalisation |
| Documentation utilisateur | 5% de la charge de réalisation |
| Qualité | 12% de la charge totale |