



Diplôme Universitaire de
Technologie en Génie Civil

المدرسة العليا للتكنولوجيا - الميوز
+KICH +eXKH+ I +KXIH8IZ+ - HAZBI
ÉCOLE SUPÉRIEURE DE TECHNOLOGIE - LAÏYOUNE



Chapitre 3 :

Estimation des charges

idougliq@gmail.com

Estimation des charges

❑ Notion de base: **Charge** et **durée**

- La **CHARGE** représente une quantité de travail nécessaire, indépendamment du nombre de personnes.
 - Elle permet d'obtenir un coût prévisionnel.
 - Elle s'exprime en mois/homme.
 - Elle aide à définir la taille d'un projet.
 - $\text{Projet} < 6 \text{ m/H} \Rightarrow$ très petit
 - $\text{Projet} > 100 \text{ m/H} \Rightarrow$ très grand (année/homme).
- La **DURÉE** est le temps consommé par le projet.
 - Elle dépend du nombre de personnes, mais l'évaluation n'est pas isotrope
 - (100 personnes pendant un mois ne sont pas équivalentes à 1 personne pendant 100 mois)

Les besoins en estimation

- ❑ Au niveau du **projet** global
- ❑ Au niveau de **l'étape**
 - Ordre de grandeur : semaine/homme
 - Ajuster le découpage
 - Sous-traiter
 - Prévoir des délais pour planifier l'ordonnancement des étapes

Les besoins en estimation

❑ Au niveau de la **phase**

- Faire une planification précise
- Annoncer un calendrier de remise des différents résultats intermédiaires
- Prévoir et effectuer un suivi, pour surveiller les écarts
- Prévoir l'affectation des ressources

Les besoins en estimation

- ❑ Au niveau de la **tâche**
 - Affectation des ressources individuelles
 - Planification au niveau le plus fin
- ❑ Visibilité croissante du projet vers la tâche
- ❑ Utilisation de techniques différentes selon le niveau de granularité

Les méthodes d'estimation

❑ **Les non méthodes:**

- Loi de Parkinson : « le travail se dilate jusqu'à remplir le temps disponible »
- « méthode du marché »: la charge correspond au prix à proposer pour remporter l'appel d'offre.

❑ Quatre « **vraies** » méthodes :

- Delphi, Cocomo/Diebold, évaluation analytique et « points fonctionnels »

Les méthodes d'estimation

❑ Schéma général

- Construire une **BC** (base de connaissances) rassemblant l'expertise des projets antérieurs
- Faire une estimation de la taille du projet à l'aide d'une unité de mesure
- Ajuster la taille ou la charge brute en fonction des spécificités du projet
- Répartir la charge entre les différentes étapes.

La méthode de répartition proportionnelle

- ❑ S'appuie sur le découpage temporel classique
- ❑ Trois types d'utilisation
 - Estimation globale du projet que l'on cherche à répartir dans le temps : **descendante**
 - Évaluation d'une des étapes au moyen d'une autre méthode, et on veut généraliser : **ascendante**
 - En cours de déroulement de projet, on a observé le temps déjà consommé et on veut estimer celui des phases à venir : **dynamique**

La méthode de répartition proportionnelle

Etape	ratio
ÉTUDE PRÉALABLE	10% du total du projet (hors mise en œuvre)
ÉTUDE DÉTAILLÉE	20 à 30 % du total du projet
ÉTUDE TECHNIQUE	5 à 15% de la charge de réalisation
RÉALISATION	40 à 60 % du total du projet
MISE EN ŒUVRE	30 à 40 % de la charge de réalisation

La méthode de répartition proportionnelle

- ❑ Ces ratios sont issus de l'expérience
- ❑ Ce sont des recommandations
- ❑ Dans l'étape ÉTUDE PRÉALABLE, on utilise une répartition proportionnelle entre phases
 - Observation : 30 à 40 %
 - Conception/Organisation 50 à 60 %
 - Appréciation : 10 %

La méthode de répartition proportionnelle

❑ L'ÉTUDE DÉTAILLÉE est la plus difficile à évaluer

Deux critères de variation :

- La **couverture** : partie du domaine étudiée.
 - PETITS PROJETS : ÉTUDE PRÉALABLE et ÉTUDE DÉTAILLÉE confondues sans surcharge pour l'EP
- La **maille** : précision de la description.

Dans certains cas, le modèle conceptuel représentant les entités est presque complet, dans d'autres cas la structuration statique doit encore être affinée.

❑ La charge de l'ÉTUDE TECHNIQUE est liée à la charge de réalisation (éventuellement augmentée d'un facteur de nouveauté)

La méthode de répartition proportionnelle

- ❑ La charge de l'étape de RÉALISATION est liée à l'ETUDE DÉTAILLÉE.
- ❑ On évalue la charge de réalisation par une autre méthode et on divise par deux pour obtenir celle de l'ED.
- ❑ La charge de l'étape de MISE EN ŒUVRE ne relève pas d'un système standard.
 - Elle est proportionnelle à la complexité des programmes écrits, et au nombre de sites.
 - Le ratio appliqué sur la charge de réalisation doit être complété par les problèmes de basculement (ancien système vers nouveau)

Les charges complémentaires

- ❑ La méthode est aussi appliquée pour l'estimation des charges complémentaires au développement de l'application
 - Tâche d'encadrement de projet
 - Recette
 - Documentation utilisateur

Tâche	ratio
Encadrement du projet :	20 % de la charge de réalisation
➤ Étape de réalisation	
➤ Autres étapes	10% de la charge de l'étape
Recette	20% de la charge de réalisation
Documentation utilisateur	5% de la charge de réalisation

La méthode DELPHI

- ☐ Elaborée en 1948 par la Rand Corporation
- ☐ Fondée sur le jugement d'experts
- ☐ Consiste à rechercher des analogies avec des projets antérieurs.
- ☐ Repose sur un raffinement successif de jugements porté par plusieurs experts jusqu'à obtention d'une convergence.

Les méthodes à modèle: COCOMO et DIEBOLDD

- ❑ Constructive Cost Model (**COCOMO**) Boehm 1981
- ❑ 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 (3^{ème} génération)

Les méthodes à modèle: COCOMO et DIEBOLÐ

- ❑ L'unité : l'instruction source
- ❑ Le modèle permet d'obtenir la **charge** de réalisation en m/H et le **délai** normal recommandé
- ❑ Formules de calcul :

$$\text{Charge (en mois/Homme)} = a (\text{Kisl})^b$$

➤ **Kisl** = kilo instruction source testée

Les méthodes à modèle: COCOMO et DIEBOLD

- ❑ **Durée** normale en mois = $c(\text{charge})^d$
- ❑ Les paramètres **a**, **b**, **c** et **d** dépendent de la catégorie du projet. Soit **L** la taille du logiciel.
 - Projet **simple** si $L < 50 \text{ Kisl}$, spécifications stables, petite équipe.
 - Projet **moyen** si $50 \text{ Kisl} \leq L < 300 \text{ Kisl}$, spécifications stables, petite équipe.
 - Projet **complexe** si $L \geq 300 \text{ Kisl}$, grande équipe.

La méthode COCOMO

Type de projet	Charge en mois homme	Durée en mois
Simple	$C=3,2 (Kisl)^{1,05}$	$D=2,5(C)^{0,38}$
Moyen	$C=3 (Kisl)^{1,12}$	$D=2,5(C)^{0,35}$
Complexe	$C=2,8 (Kisl)^{1,2}$	$D=2,5(C)^{0,32}$

La méthode COCOMO (Exemple)

- Soit un projet visant à développer un logiciel de 40 000 instructions source
- C'est un petit projet par la taille du logiciel.
- Charge = $3,2 (40)^{1,05} = 154$ mois/homme
- Durée normale =
- $2,5 (154)^{0,38} = 17$ mois
- Ce qui donne une taille moyenne de l'équipe = $154 / 17 = 9$ personnes.

La méthode COCOMO

- ❑ Il faut tenir compte des « **facteurs correcteurs** » d'estimation de charge.
- ❑ Quatre sources de risque sur l'estimation
 - Exigences attendues du logiciel
 - Caractéristiques de l'environnement technique (matériel)
 - Caractéristiques de l'équipe projet
 - Environnement du projet lui-même

La méthode COCOMO

❑ Les facteurs logiciels sont :

- Fiabilité du logiciel : influence forte si exigence dans ce sens
- Base de données : mesuré par le ratio
 - (volume de données gérées en octets) / (taille du logiciel en lignes)
 - L'influence du facteur est faible si le ratio < 10, très forte si ratio > 1000
- Complexité : celle des algorithmes
- Temps d'exécution : crucial si temps réel

La méthode COCOMO

❑ Les facteurs matériels sont :

- Taille mémoire : s'il est nécessaire de l'optimiser
- Stabilité de l'environnement : celle du logiciel de base
- Contrainte de délai : se mesure par rapport au délai calculé « normal ».

La méthode COCOMO

❑ Démarche en cinq étapes:

1. Estimation du nombre d'instructions source.
2. Calcul de la charge « brute ».
3. Sélection des facteurs correcteurs.
4. **Calcul de la charge nette = produit (valeurs des facteurs correcteurs) * Charge brute**
5. Evaluation de la durée sur la charge nette.

La méthode DIEBOLD

- ❑ Version antérieure et simplifiée de COCOMO.
- ❑ Connaît le nombre d'instructions à écrire et donne le temps en jours

$$\text{Temps(jours)} = (\text{complexité}) * (\text{savoir-faire}) * (\text{connaissance}) * (\text{kisl})$$

La méthode DIEBOLD

❑ Complexité : celle du logiciel.

➤ $10 \leq c \leq 40$

❑ Savoir-faire : mesure l'expérience du programmeur

➤ Beaucoup de savoir faire : **0,65**

➤ Peu de savoir faire : **2**

❑ Connaissance: celle de l'environnement technique :

➤ **1 = bonne K et 2 = faible K**

La méthode Analytique

- ❑ S'appuie sur la typologie des programmes à développer
- ❑ Affecte un poids par type de programme et niveau de difficulté dans l'environnement
 - UNITÉ : **jour/homme**
- ❑ La **charge** obtenue est celle de **réalisation**
- ❑ Pour les test d'enchaînement : 10% charge
- ❑ Pour l'encadrement : 20% charge

La méthode Analytique

TYPE DE PROGRAMME	FACILE	MOYEN	DIFFICILE
MENU	0.25	0,5	1
CONSULTATION	1	2,5	4
MISE A JOUR	1,5	3	5
EDITION EN TEMPS RÉEL	1	2	4
EXTRACTION	0,5	1	1,5
MISE A JOUR PAR LOT	2	3	5
EDITION PAR LOT	1,5	2,5	4

La méthode Analytique

❑ Charge de réalisation **$C_r = \text{somme } (p_i * t_i)$**

➤ **p_i** est le poids

➤ **t_i** le nombre de programmes du type i

❑ Charge globale **$C_g = 1,3 * C_r / 22$ (en m/H)**

❑ Pour les projets dont la charge est comprise entre 3 et 30

➤ **Durée incompressible = $2,5 (C_g \text{ (en m/H)})^{1/3}$ en mois**

La méthode Analytique (Exemple)

Réalisation des composants d'une application intranet Notes/Domino

Complexité	Facile			Moyen			Difficile			Total
Type composant	N	P	C	N	P	C	N	P	C	
Vue	8	0,25	2	20	0,5	10	12	1	12	24
Masque	9	0,5	4,5	6	1	6	10	2	20	30,5
Navigateur web				12	0,5	6				6
Agents							20	0,5	10	10
Accès bases données externes				4	0,25	1				1
Programme lotusScript	6	0,5	3	3	2,5	7,5	5	3	15	25,5
Reprise données							1	9	9	9
Total Réalisation										106
Intégration								10 %		10,6
Encadrement								20 %		21,2
TOTAL PROJET										138

N=Nombre
P=Poids
C=Charge

La méthode de points fonctionnels

- ❑ A. Albrecht (IBM) 1979
- ❑ Groupe d'utilisateurs : guide en 1984
- ❑ En France, groupe FFPUG en 1992
- ❑ Principe :
 - Estimation à partir d'une description externe du futur système, et de ses fonctions.
 - 5 types d'unité d'œuvre et 3 degrés de complexité

La méthode de points fonctionnels

- ❑ Pour un projet donné on calcule son **poids** en « **points de fonction** » **PF**.
- ❑ Méthode:
 - Comptage des points au début du projet
 - Comptage en fin
 - Écart = changement d'envergure
 - Évaluation :
 - Calcul de la taille, ajustement de la taille, transformation en charge.

La méthode de points fonctionnels

❑ Composants fonctionnels :

- Groupe logique de données internes (GDI)
- Groupe logique de données externes (GDE)
- Entrée de traitement (ENT)
- Sortie de traitement (SORT)
- Interrogation (INT)

La méthode de points fonctionnels

❑ Complexité d'un composant :

- Faible
- Moyenne
- Elevée

❑ Nombre de points de fonction du composant :

- Tableau de correspondance entre la complexité et le type du composant = > poids

Calcul du nombre de points de fonction brut: (Exemple)

Entité	Complexité	Nb de composants	Poids	Nb de Points de fonction
GDI	Faible	3	7	21
	Moyenne	1	10	10
	Elevée	1	15	15
GDE	Faible	2	5	10
	Moyenne	2	7	14
	Elevée	3	10	30
ENT	Faible	4	3	12
	Moyenne	6	4	24
	Elevée	2	6	12
SORT	Faible	3	4	12
	Moyenne	4	5	20
	Elevée	0	7	0
INT	Faible	2	3	6
	Moyenne	5	4	20
	Elevée	4	6	24
PFB				230

La méthode de points fonctionnels

- ❑ Le PFB (points de fonction brut) est ensuite ajusté par une appréciation des spécificités du projet.
 - 14 points sont identifiés, auxquels est attribuée une note de 0 à 5 en fonction du degré d'influence (réutilisabilité, portabilité, ...)
- ❑ Le PFA ou nombre ajusté de points
 - **$PFA = (0,65 * (SOMME (Di_i, i = 1 \text{ à } 14))/100) * PFB$**

La méthode de points fonctionnels

- ❑ Le PF permet de donner le nombre d'instructions source utile pour COCOMO ou DIEBOLD avec la formule :
 - **$ISL (l_{\text{procédural}}) = 118,7 * PFA - 6490$.**
 - Dans l'exemple, si $PFA = PFB$ alors $ISL = 20811$.
- ❑ Mais on calcule la charge en général en convertissant directement les points.

La méthode de points fonctionnels

- ❑ En fin d'étude préalable
 - 3 j/H / pF
 - 2 jours si petit projet
 - 4 jours si grand projet
- ❑ En fin d'étude détaillée: 1 à 2 j/pf selon l'environnement
- ❑ Avec un L4G 1j /10 pf en réalisation.
- ❑ En RAD, productivité élevée : 0,5 j/H/pF

Exercices

Exercice1 :

L'étape "étude préalable" d'un projet est estimée à 10 jours. En utilisant la méthode de la répartition proportionnelle, estimer les charges des différentes étapes du projet.

Exercice2 :

Un projet de logiciel est estimé à 50 000 instructions source. En utilisant les méthodes Cocomo, puis la méthode de répartition proportionnelle, déterminer la charge des différentes étapes du projet