

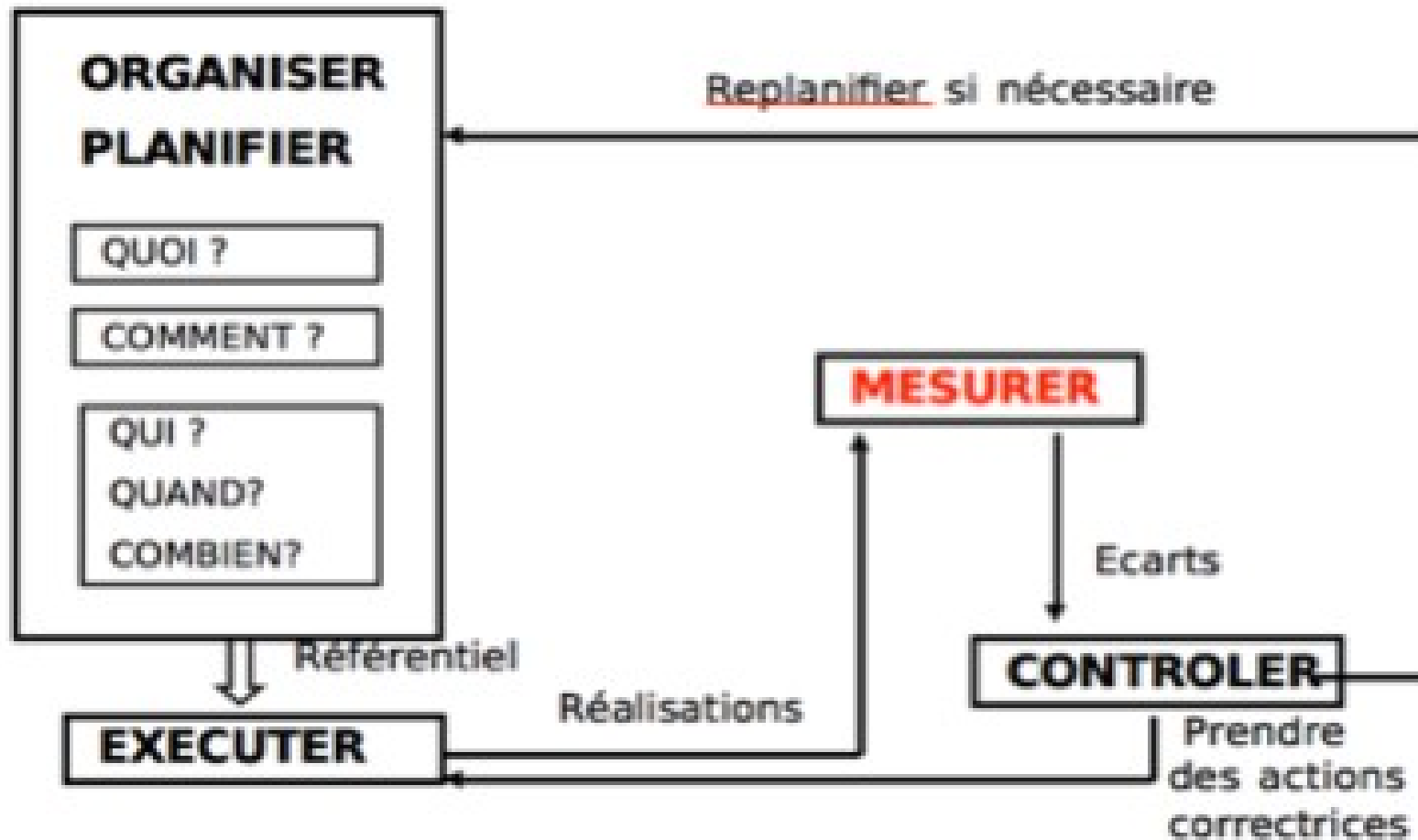
Licence Sciences, Technologies, Santé  
Mention SPI, parcours Informatique

Introduction au Génie Logiciel  
L3 / 175EN002

# C5 – Introduction à la gestion de projet informatique *Estimation*

Claudine Piau-Toffolon  
[claudine.piau-toffolon@univ-lemans.fr](mailto:claudine.piau-toffolon@univ-lemans.fr)  
[www-ic2.univ-lemans.fr/~cpiau](http://www-ic2.univ-lemans.fr/~cpiau)

# Activités de la gestion de projet



# Pourquoi mesurer le logiciel ?

- 80% des **coûts** du logiciel sont des coûts de maintenance
- Il y a en moyenne 55 **bogues** par 1000 lignes de code
- La méthode X permet d'accroître la **productivité** des études de 25%
- En consacrant plus de temps à la spécification, on diminue le **coût** total du projet
- Ce logiciel est **convivial, fiable, maintenable**
- Tel algo de tri un a une **efficacité** de  $n \log n$

## Pourquoi mesurer ?

L'acte de mesurer consiste à :

*Affecter des valeurs (nombres, symboles) à des attributs d'entités du monde réel de manière à ce qu'on les décrive selon des règles clairement établies.*

**Exemple** : On affecte la valeur 10 à l'attribut « temps passé en homme-mois » de l'entité « codage du programme P5 »

Les exemples proposés correspondent-ils à cette définition ?

Pourquoi mesurer ?

*La production du logiciel n'est pas contrôlée si l'on  
ne fait pas de mesure et  
on ne peut contrôler ce que l'on ne peut pas  
mesurer*

HTTP://WWW.COLORAGE-ENFANTS.COM



Tom DeMarco

# Pourquoi mesurer ?

## Au niveau projet :

- Déterminer une enveloppe budgétaire
- Estimer le poids du projet en termes d'effort
- Faire une estimation de la rentabilité de l'investissement
- Evaluer une durée vraisemblable du projet

## Au niveau étape :

- Ajuster le découpage
- Sous-traiter
- Prévoir les délais
- Prévoir les ressources

## Au niveau phase :

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

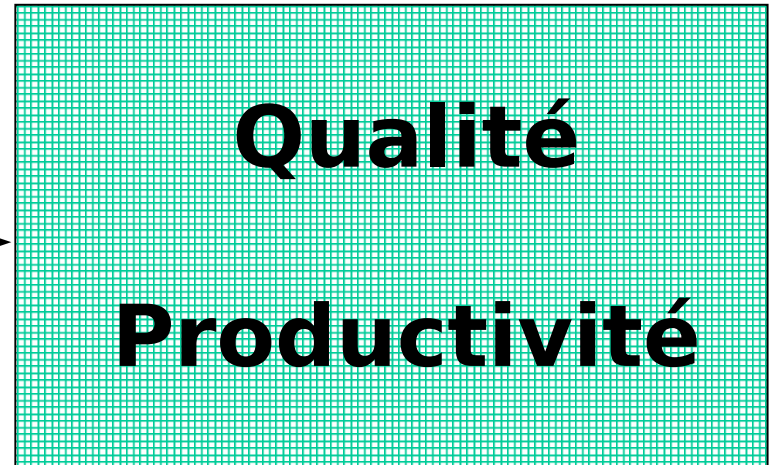
# Pourquoi mesurer ? Quel Besoin ?:

- Le coût de différents processus pendant le développement (déterminer le prix)
- *Mesurer la productivité du personnel (définir les salaires)*
- Mesurer la qualité du logiciel (comparer les projets, faire des prévisions, se donner des objectifs d'amélioration)
- *Définir des objectifs mesurables (ex: couverture de test à effectuer)*
- Mesurer plusieurs des attributs et ressources d'un processus particulier (déterminer quels facteurs affectent le coût et la productivité)
- *Evaluer l'efficacité de diverses méthodes et outils*
- Mesurer les modifications faites pendant la conception, les erreurs trouvées en phase révision ou de test (surveiller la qualité des systèmes en évolution)
- *Définir des attributs mesurables (spécifier des exigences de qualité et de performance qui soient testables)*
- Mesurer des attributs de produits et de processus (obtenir une certification)
- *Mesurer des attributs de produits existants et de processus en cours (faire des prévisions)*

**EVALUER  
et PREDIRE**

Que mesure-t-on ?

- Coût
  - bogues
  - fiabilité
  - maintenabilité
  - convivialité
  - efficacité
  - ...





# Les composants du coût d'un logiciel

- Coût du matériel
- Coût des logiciels de base
- Coût de la formation et des voyages
- Coût de l'effort
  - Salaires des ingénieurs
  - Coût des bâtiments, chauffage, éclairage
  - Coût des communications, réseaux
  - Coût des pensions, assurance maladie,...
  - ...

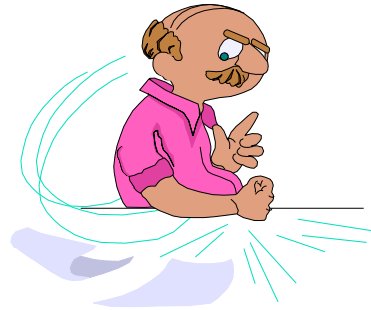
**OBJECTIF** : Découvrir combien va coûter le développement d'un logiciel !

# PRODUCTIVITE

- Définition ?
- Objectif ?
- Facteurs de productivité
- Métriques

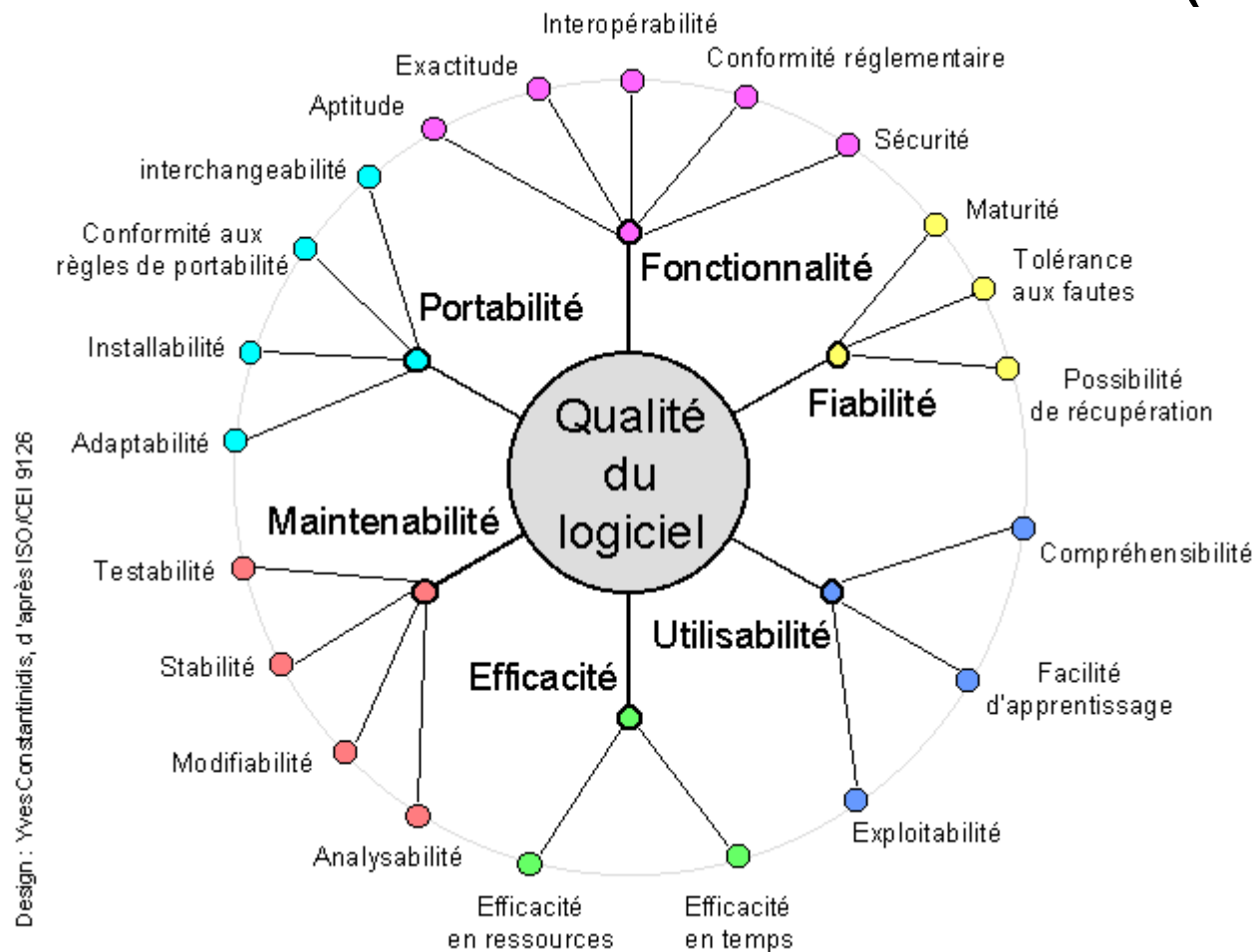
# QUALITE ?

- En génie logiciel



La qualité d'un logiciel est son aptitude à satisfaire les besoins (exprimés ou potentiels) des utilisateurs [Martin, 1987] (NF X50-120)

# Facteurs Critères et métriques de Qualité (ISO 9126)



# DEFINITION de la Mesure

## **PRODUIT :**

**l'objectif du projet**

**PROJET / Ressources:**  
**les travaux et ressources qui**  
**permettent d'atteindre l'objectif**

**PROCESSUS :**  
**les étapes à parcourir dans le**  
**déroulement nominal des travaux**

# DEFINITION de la Mesure

## *Attributs internes et externes*

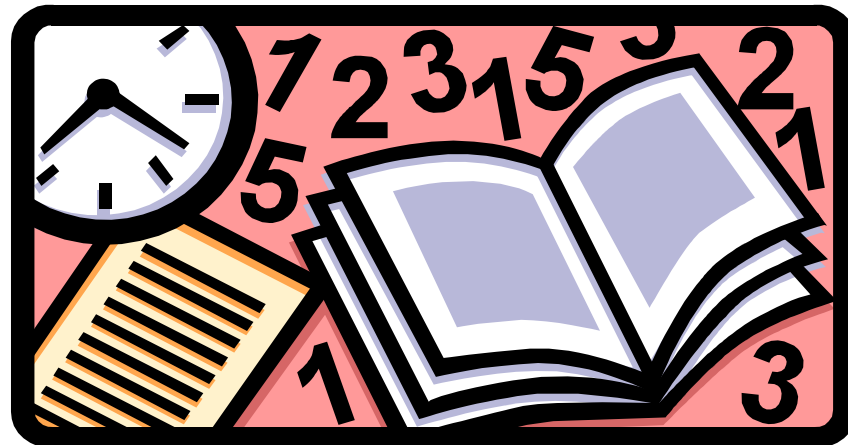
**Attributs internes** : Ceux qui peuvent être mesurés en termes de produits, processus, ressource

- La ***longueur*** pour représenter la taille d'un produit :
  - spécification, conception, code
- La ***modularité***, le ***couplage***, la ***cohésion*** d'une conception
- La ***structuration du flot de contrôle*** du programme
- La ***complexité cyclomatique*** du code
- L'***effort*** en nombre de personnes et en temps pour réaliser tel processus
- Le ***nombre d'incidents*** en cours de processus

## Exemples de métriques

- **Lignes de code** : mesure de la taille d'un logiciel
- **Points fonctions** : mesure des fonctionnalités du logiciel
- **Homme-mois** : mesure de l'effort de développement

## EXERCICE



# DEFINITION de la Mesure

## *Attributs internes et externes*

**Attributs externes** : Ceux qui peuvent seulement être mesurés relativement à la façon dont le produit, le processus, la ressource sont reliés à leur environnement

- La ***maintenabilité*** d'une spécification, d'une conception, du code
- La ***qualité*** de conception



## Effort ? Durée ? Taille ?...

**Effort ou charge** : quantité de travail nécessaire, indépendamment du nombre de personnes qui vont réaliser ce travail

- Permet d'obtenir un coût prévisionnel
- S'exprime en homme/jour, homme/mois ou homme/année
- Un homme/mois (HM) représente l'équivalent du travail d'une personne pendant un mois, généralement 20 jours

=> Un homme/mois (HM) = ?  
heures de travail par mois

### Exemple:

Un projet de 60 mois/homme représente l'équivalent du travail d'une personne pendant 60 mois

Si on évalue le coût du mois/homme à 50 K€ en moyenne, le projet sera estimé à 3 M€

# Effort ? Durée ? Taille ?...

## *L'Homme-Mois*

- Unité de mesure de l'effort
  - Un homme pendant un mois
  - Deux hommes-mois = 2 hommes pendant 1 mois, ou 1 homme pendant deux mois
- Selon F. Brooks : « *L'homme-mois comme unité pour mesurer la taille d'un travail est un mythe dangereux et trompeur. Il implique que les hommes et les mois sont interchangeables. Les hommes et les mois sont des biens interchangeables seulement lorsqu'une tâche peut être partitionnée entre plusieurs employés sans qu'il faille une communication entre eux* ».

## Effort ? Durée ? Taille ?...

On mesure la taille des projets à leur charge

- Si Charge < 6 HM : très petit projet
- Si 6 HM < Charge < 12 HM : petit projet
- Si 12 HM < Charge < 30 HM : projet moyen
- Si 30 HM < Charge < 100 HM : grand projet
- Si Charge > 100 HM : très grand projet

Effort ? **Durée** ? Taille ? ...

La **Durée** dépend du nombre de personnes

- 60 HM peut correspondre à :
- 1 personne pendant 5 ans ,
- 5 personnes pendant un an ,
- 10 personnes pendant 6 mois
- 60 personnes pendant 1 mois

## Définition [Fenton 1991]

# Effort ? Durée ? **Taille** ?...

## *Les lignes de code*

- Une ligne de code est toute ligne du texte d'un programme qui n'est pas une ligne de commentaires ou une ligne blanche, sans considération du nombre d'instructions ou de fragments d'instructions dans la ligne. Sont incluses toutes les lignes contenant des en-têtes de programmes, des déclarations et des instructions exécutables et non exécutables.



## EXERCICE

(FENTON 97, p 250)

# Problèmes avec les lignes de code ?

- Selon les définitions , le nombre de lignes d'un même programme peut varier de 1 à 5
- Que mesurer avec ce nombre ?
  - L'effort ?
    - Certaines lignes étant plus difficiles à coder que d'autres, il faudrait une pondération de ces lignes
  - Fonctionnalité du produit ?
  - Selon le langage, la même fonctionnalité prendra plus ou moins de lignes

# Problèmes avec les lignes de code ?

- Qu'est-ce qu'une ligne de code?
- Quels programmes doivent être considérés comme faisant partie du système?
- L'hypothèse selon laquelle il existe une relation linéaire entre le volume de la documentation et la taille du système
- Plus le langage est de bas niveau, plus le programmeur utilisant ce langage est productif
- Plus le style est verbeux, plus la productivité est élevée

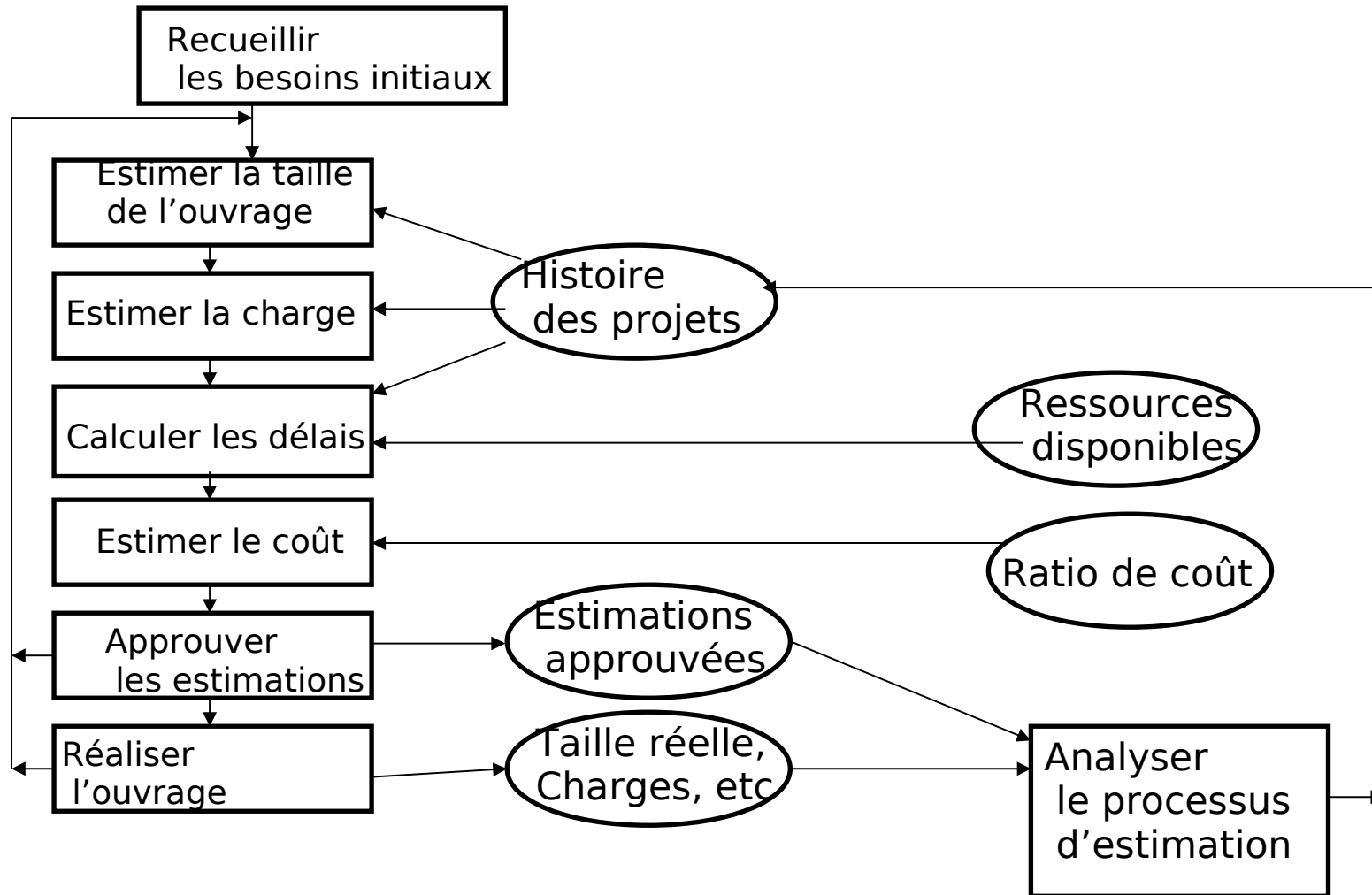
## Exemple : *Productivité comparée*

<b>Langage</b>	<b>Taille en lignes de code</b>	<b>Effort (en semaines)</b>	<b>Productivité</b>
<b>Assembleur</b>	5000	28	714 lignes/mois
<b>Langage de haut niveau</b>	1500	20	300 lignes/mois



# Processus d'estimation d'un projet

Ré estimer si nécessaire



# Conclusion

- Estimer le coût du projet pour le fournisseur puis décider du prix facturé au client
- *L 'estimation algorithmique du coût est difficile puisqu 'elle repose sur des estimations des attributs du logiciel final*
- Importance des modèles d 'estimation du coût du logiciel pour le management: moyen de comparaison entre plusieurs options de développement
- *Le temps requis pour développer un projet n 'est pas simplement proportionnel à la taille de l 'équipe de projet*
- Chaque organisation doit déterminer ses propres attributs et ses propres multiplicateurs en fonction de ses spécificités, ses priorités et ses contraintes