

# Gestion de projet en génie logiciel



La planification de projet et ses  
processus

# Actualité

- Le gouvernement veut engager 1000 personnes en TI pour remplacer les consultants dans les ministères et organismes gouvernementaux
  - <http://www.journaldemontreal.com/2015/09/02/quebec-en-mode-embauche>
  - En tant que chargé de projet, que pensez-vous de cette approche? Pourquoi ?

# Récapitulatif

# Cours 1

- Qu'est-ce qu'un projet?
- Qu'est-ce que la gestion de projet?
- Quelles sont les habiletés d'un chargé de projet?
- Quelles sont les 2 premiers critères de succès d'un projet?
  - Nommez-en d'autres?
- Décrire les 3P+P
- Pourquoi de la gestion de projet en TI

# Cours 2

- Qu'est-ce qu'une approche holistique
- Pourquoi un chargé de projet doit avoir une vue systémique?
  - Que doit-il considérer? Des exemples?
- Comment la culture de l'organisation peut influencer un projet
- Le chargé de projet...son rôle et ses responsabilités

# Cours 3

- Le plan directeur
  - Qu'est-ce que c'est?
  - Qu'est-ce que cela contient?
  - À quoi ça sert?
  - Quel est le lien avec les projets?

# Cours 3

- Le démarrage du projet
  - Pourquoi est-ce important?
  - Quel est le principal livrable?
  - Que devez-vous savoir à la fin de cette phase?
- La facilitation
  - À quoi ça sert?
  - Donner quelques exemples de tâches qu'un facilitateur va faire

# Retour sur la lecture (1 de 3)

- Qu'est-ce que le Scope Management?
  - Planning
  - Collecting...
  - Defining
  - WBS
  - Validating
  - Controlling
- Qui fait ça?
- Explicitation des exigences
  - Combien de temps faut-il consacrer à l'explicitation des exigences
  - Coûts d'une exigences si détecté...
- Qui fait ça?



# Retour sur la lecture (2 de 3)

- WBS
  - Level 1 → Projet entier
  - Level 2 → Grands livrables
  - Level 3 → Tâches
    - Ce n'est pas la seule façon de voir un WBS
- Découpage
  - Par livrable
  - Par phase (Initiating/Planning/Executing/Monitoring&Controlling/Closing)
- WBS, ce n'est pas linéaire dans le temps
- WBS, ce n'est pas un calendrier, ni une « *schedule* »

# Retour sur la lecture (3 de 3)

- WBS
  - Différence avec un cahier des charge?
  - Faire la différence entre une tâche et une spécification
- Dictionnaire WBS
- Techniques
  - Guides / Guidelines
  - Analogie
  - Top-Down
  - Bottom-Up
  - Mind mappings

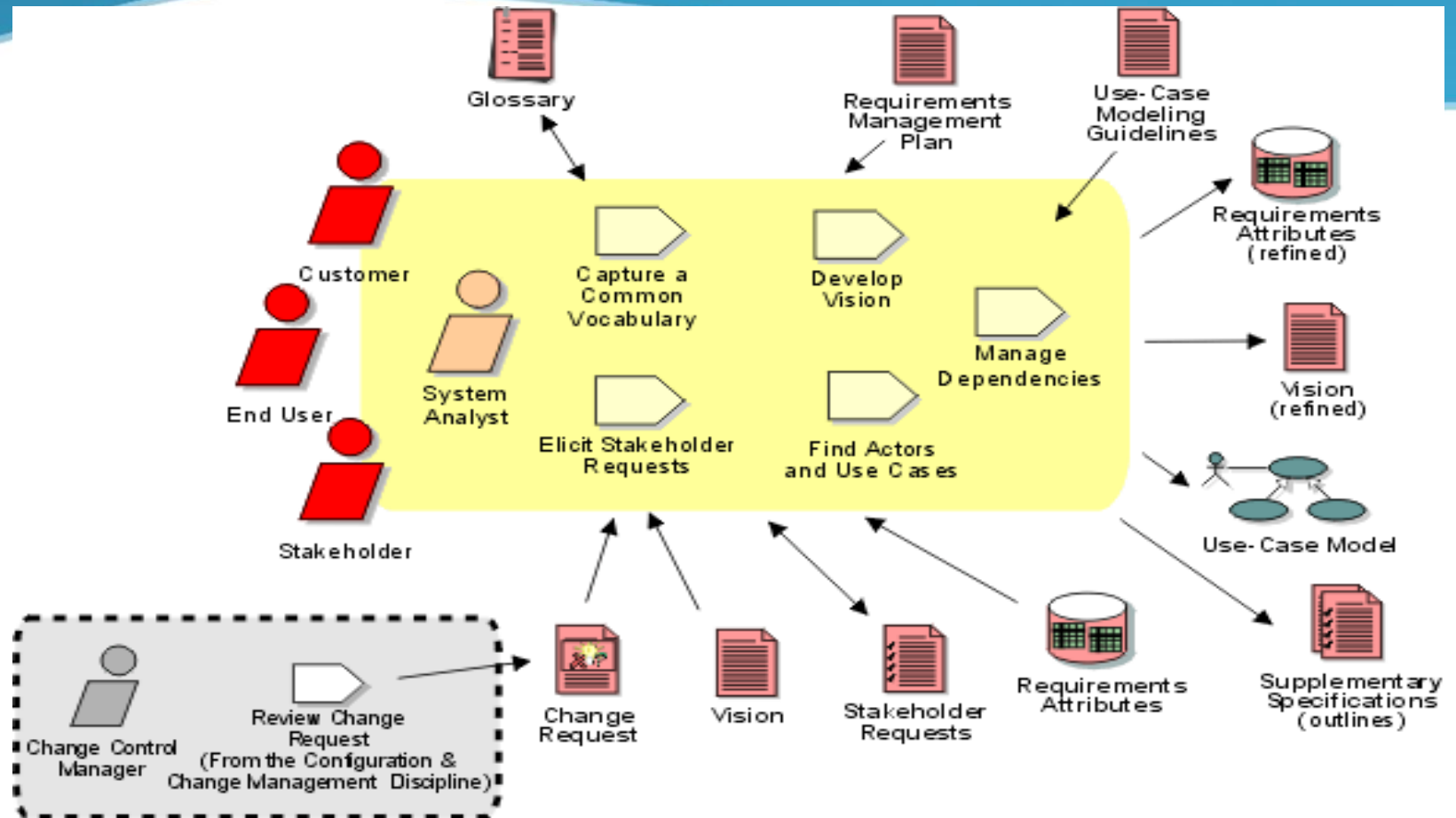
# Introduction - Constats

- Une grande proportion de projets TI échouent. Pourquoi?
  - Plusieurs facteurs
  - Les mauvaises exigences en font partie
- Si ça va si mal, c'est qu'il y a quelque chose qu'on ne fait pas bien. Il faut se remettre en question!

# Agenda

- **Gestion des besoins**
- **Planification de projet**
- **Cas:** Exercice de planification en classe

# Comprendre les besoins



# Objectifs

- Rappelez-vous les raisons pour lesquelles les projets échouent...
- L'objectif de ce processus est de collecter et expliciter l'information provenant des intervenants (stakeholders) du projet pour comprendre leurs besoins.
- Les requêtes peuvent être vues comme une “Wish list” qui va être utilisé comme premier intrant pour définir les caractéristiques (features) de haut niveau du système.
- Par la suite, ces caractéristiques alimenteront les exigences logicielles, telles qu'elles seront décrites dans le modèle des cas d'utilisation, les cas d'utilisation et les spécifications supplémentaires.

# Expliciter les requêtes des intervenants

- **Objectifs**

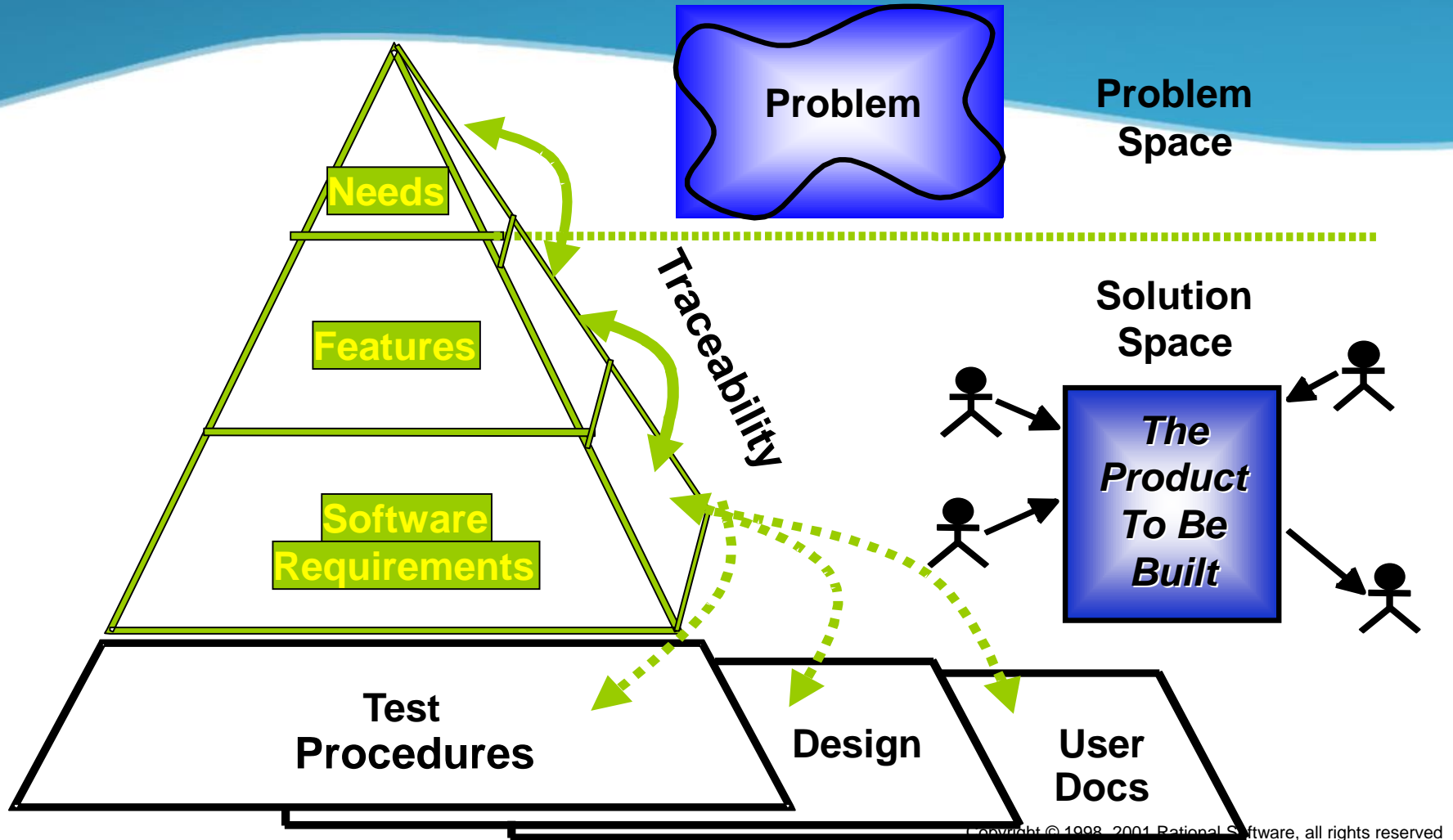
- Qui sont les intervenants du projet?
- Quels sont les besoins auxquels le système doit répondre?
- Prioriser les requêtes des intervenants.

# Expliciter les requêtes des intervenants

- Étapes
  - Identifier les sources d'information des exigences
  - Récupérer l'information
  - Réaliser des ateliers d'explicitation des exigences
  - Évaluer les résultats



# La bonne vieille pyramide



# Les problèmes de l'explicitation

- Le syndrome du *oui..mais*
- Le syndrome des *ruines non découvertes*
- Le syndrome des *utilisateurs et des développeurs*

# Le syndrome du *oui...mais*

- Lorsque l'utilisateur voit l'implémentation, il y a 2 réactions possibles
  - Wow, c'est cool
  - Oui, mais...hummmm, maintenant que je vois ça, pourrions-nous...et que pensez-vous de....
- Il faut vivre avec...
  - Permet de découvrir de nouvelles exigences
  - Minimiser ce syndrome en explicitant les exigences plus tôt

# Le syndrome des *ruines non découvertes*

- Combien de ruines reste-t-il à découvrir...
  - Plus vous en trouvez....moins il y en a.
- Même chose pour les exigences...
  - Vous ne saurez jamais si vous les avez tous trouvées
  - Vous espérez en trouver assez.

# Le syndrome des utilisateurs et des développeurs

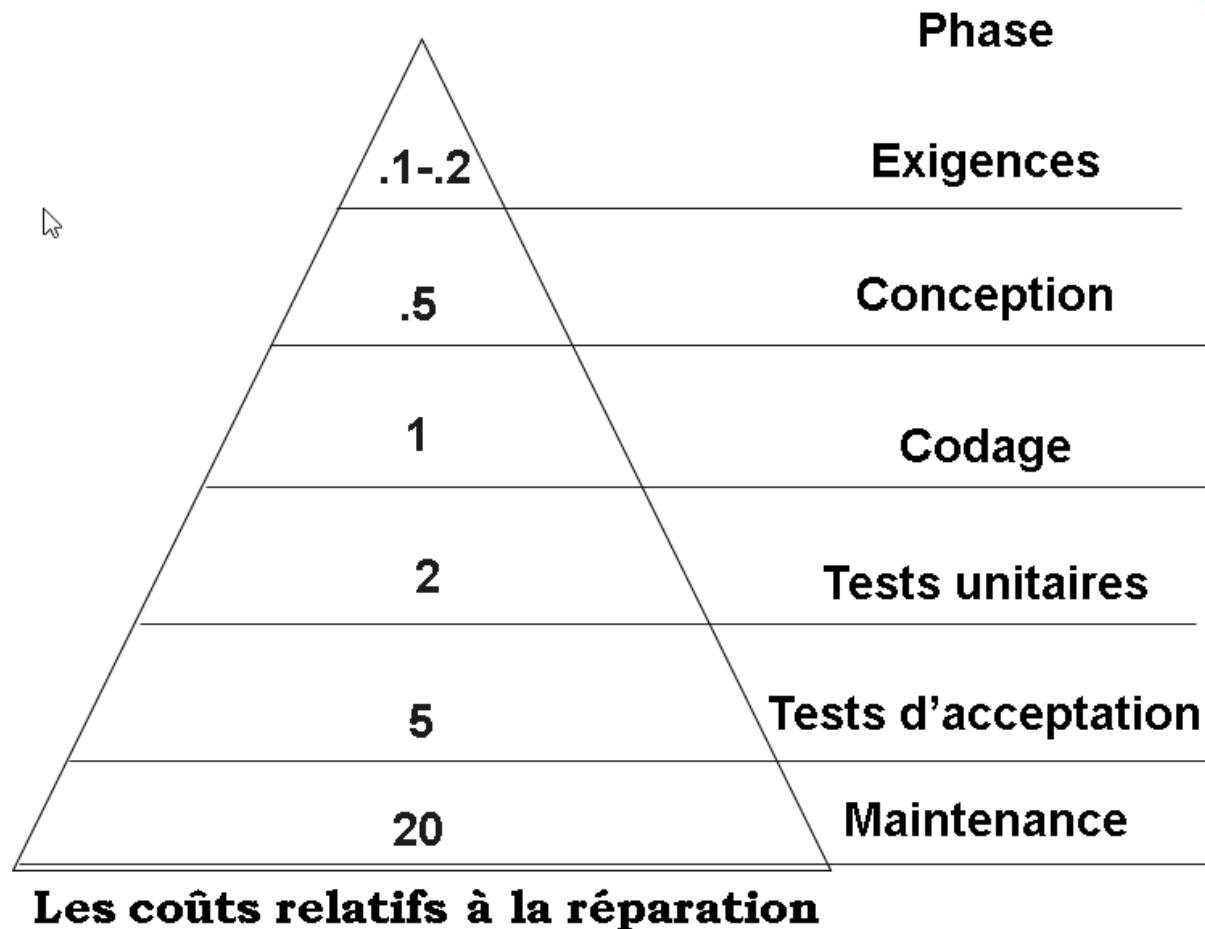
L'utilisateur :

- Ne sait pas ce qu'il veut
- Le sait, mais ne peut l'expliquer
- Pense savoir ce qu'il veut – avant de se le faire dire par le développeur
- L'analyste pense savoir + que l'utilisateur
- Tout le monde pense que tous les autres font de la politique

# Le syndrome des utilisateurs et des développeurs

- Les solutions...
  - Reconnaître et apprécier l'utilisateur comme un expert du domaine.
  - Essayer différentes techniques d'explicitation des exigences
  - Mettez-vous à la place de l'utilisateur, faites son travail pendant 1 heure ou 2.
  - La politique fait partie de la nature humaine...

# Les coûts relatifs à la correction des défauts à travers les différentes phases du cycle de vie



# A la fin de cette séance nous connaîtrons:

- Les méthodes pour recueillir et documenter les besoins
- La notion de bien livrable
- Le processus de planification de projet
  - Le processus pour créer une charte et un WBS



# Figure 5-1. Project Scope Management Summary

## Planning

Process: **Collect requirements**

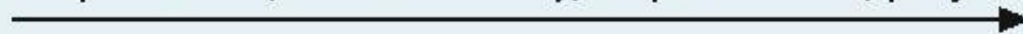
Outputs: Requirements documentation, requirements management plan, requirements traceability matrix

Process: **Define scope**

Outputs: Project scope statement, project document updates

Process: **Create WBS**

Outputs: WBS, WBS dictionary, scope baseline, project document update



## Monitoring and Controlling

Process: **Verify scope**

Outputs: Accepted deliverables, change requests, project document updates

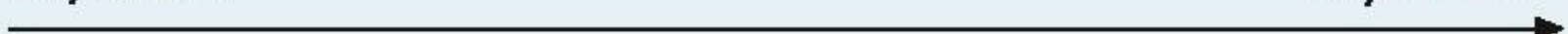
Process: **Control Scope**

Outputs: Work performance measurements, organizational process assets updates, change requests, project management plan updates, project document updates



Project Start

Project Finish





# Les Besoins

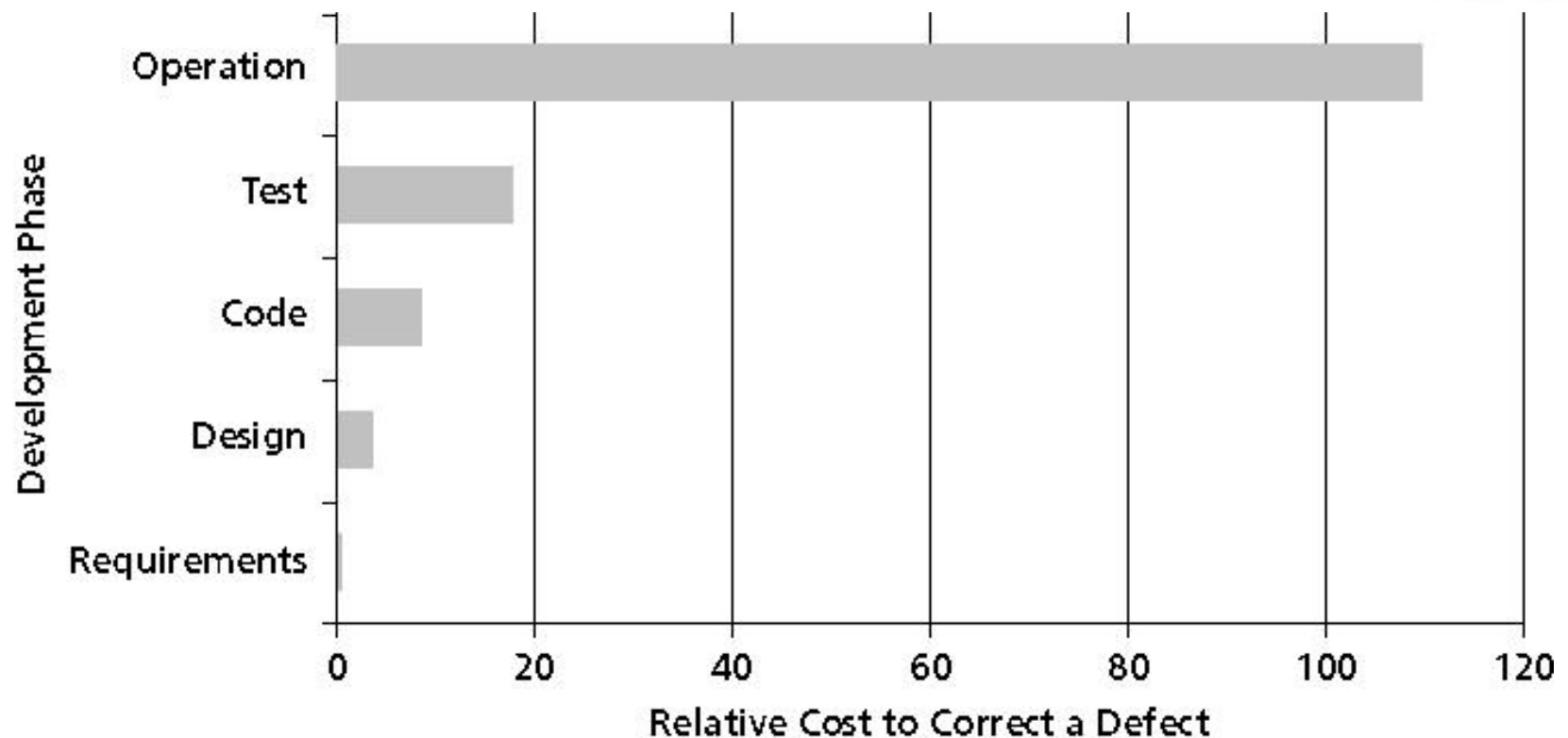
# Les besoins

- A **requirement** is “a condition or capability that must be met or possessed by a system, product, service, result, or component to satisfy a contract, standard, specification, or other formal document” (PMBOK® Guide, 2008)
- Dans plusieurs projets TI, la gestion des besoins comporte les étapes **identification, analyse, définition détaillée et validation**

# Les besoins

- Une approche **itérative** est souhaitable car les besoins sont souvent **flous** ou **incomplets** en début de projet
- Probablement l'étape la plus **complexe** et aussi la plus **importante** d'un projet
- Une erreur classique: Besoins **incomplets**
- Débute souvent par un "Kick off"
- Se termine souvent avec une **revue formelle** des besoins (Software Requirements Review (SRR) )
- **Approbation** requise des clients ou demandeurs

# Pourquoi les besoins sont-ils si importants ?



Source: Robert B. Grady, "An Economic Release Decision Model: Insights into Software Project Management." *Proceedings of the Applications of Software Measurement Conference* (Orange Park, FL: Software Quality Engineering, 1999), pp.227-239.

# Souvent 2 types de besoins

- **Fonctionnel**

- Fonctions et capacité

- **Non fonctionnel** (a.k.a. “technique”) (tout le reste)

- Utilisabilité
    - » Facteurs humains, aide, documentation
  - Capacité
    - » Disponibilité, relève, périodes de maintenance
  - Performance
    - » Temps réponse, performance machine, utilisation de ressources
  - Support
    - » Langues, ressources de support requises, etc...
  - Operations: gestion du système, installation, etc...
  - Intégration: intégration avec d'autres services et systèmes
  - Autres: matériel requis, aspects légaux, vente, etc...

# Les besoins

- Doivent être priorisés
  - Requis
  - Importants mais non indispensables
  - Nice to have (le superflu ou bonus)
- Doivent être approuvés

# Les Besoins

- Différents points de vue
- Équipe de projet
  - Gestionnaires: Oui, c'est ce que j'ai payé (commandé...)
  - Utilisateurs: Oui, c'est ce que j'ai besoin
  - Développeur: Oui, c'est ce que j'ai programmé



# Méthodes de cueillette des besoins

- Entrevues
  - Ateliers de travail ou groupes d'utilisateurs
  - Sondages et questionnaires
  - Observation
  - Prototype
  - Outils logiciels
- 
- Nous les étudierons en détail lors d'une autre séance

# Documentation des besoins & outils

- La documentation de besoins se compose souvent de texte, images, diagrammes, vidéos, ou autres; Ils sont souvent divisés en catégories tel que: Fonctionnel, Technologique, Performance, Support, Qualité, Formation, etc...
- Un **plan de gestion des besoins (requirements management plan)** défini comment les besoins seront analysés, documentés et gérés
- Une matrice de traçabilité des besoins (**requirements traceability matrix (RTM)** ) défini les attributs des différents besoins, le statut, la source des besoins, etc...

# Table 5-1. Sample Requirements Traceability Matrix

Requirement No.	Name	Category	Source	Status
R32	Laptop memory	Hardware	Project charter and corporate laptop specifications	Complete. Laptops ordered meet requirement by having 4GB of memory.

# Attributs des caractéristiques

Attribut	Description
Statut	Suit le progrès. Exemples : Proposé, approuvé, incorporé, etc.
Priorité/ Bénéfice	Les caractéristiques ne sont pas à priorités égales ou génèrent des bénéfices égaux. Exemple : critiques, importantes, optionnelle
Effort	Estimation du nombre de personnes/semaines, lignes de code, points de fonction, COSMIC_FFP
Risque	La probabilité que la caractéristique aille causer un événement indésirable, tel que le dépassement du budget ou des délais ou même l'annulation du projet

# Attributs des caractéristiques

Attribut	Description
<b>Stabilité</b>	La probabilité que la caractéristique elle-même ou sa compréhension par l'équipe de développement changera
<b>Mise en application</b>	Documenter la version prévue pour laquelle la caractéristique sera incorporée
<b>Assignée à</b>	Désigner l'équipe responsable de la caractéristique
<b>Raison</b>	L'origine de la requête de la caractéristique

# Le problème avec les besoins...

- Comment s'assurer de **bien comprendre** les besoins du client?
- Il faut bien différencier **BESOINS** vs **SOLUTION**
- **Impacts très importants** sur le projet si le besoin est mal défini

# Bonnes pratiques gestion des besoins

- Avoir une méthode de gestion des besoins pour bien **distinguer les besoins des solutions**
- Travailler **conjointement** avec les utilisateurs dans ce processus
- Utilisation de techniques reconnues (Maquettes, Use Case Modeling, Joint App Development - JAD)
- Exemples de UC: <http://alistair.cockburn.us/Use+cases>
- S'assurer que les **besoins sont documentés**

# Bonnes pratiques gestion des besoins (suite)

- Bien planifier les **essais** et **l'arrimage** aux besoins
- Utiliser les critères **SMART** pour les besoins, les critères de **succès** et les **contraintes**
  - Date d'implantation
  - 2 Langues à supporter
  - Inclusions & exclusions





# Les Biens Livrables

# Clarification biens livrables - Définition

- Dans un projet, nous avons des tâches et des livrables
- Biens livrables
  - Document devant être **produit** et **vérifié/approuvé** lors de la réalisation d'un projet. Ce type de document peut-être encadré par un Cadre de gestion de projet, Une Méthodologie (du domaine) de développement ou simplement requis pour la réalisation du projet mais non encadré par un programme spécifique.

# Catégories et Exemples

- **Biens livrables de gestion:** Document produit dans le cadre du projet à des fins de gestion ou coordination du projet et non des objectifs de celui-ci. Généralement, ce type de bien livrable est encadré par un Cadre de gestion de projet.
- **Exemples:**
  - Manuel Organisation de Projet/Charte de projet
  - Plan de communication
  - Plan de gestion des risques
  - Planification détaillée des activités
  - Post mortem

# Catégories et Exemples...suite

- **Biens livrable du projet:** Document produit dans le cadre du projet pour les objectifs du projet. Généralement ce type de document est encadré par une Méthodologie de développement.
- **Exemples:**
  - Le Code...
  - Documents d'architecture (sécurité, fonctionnelle,...)
  - Documents d'analyse (préliminaire, sécurité, fonctionnelle...)
  - Certains documents sont des documents de projet mais ne sont pas encadrés :
    - Configuration détaillée, Guide de post implantation, Plan de relève

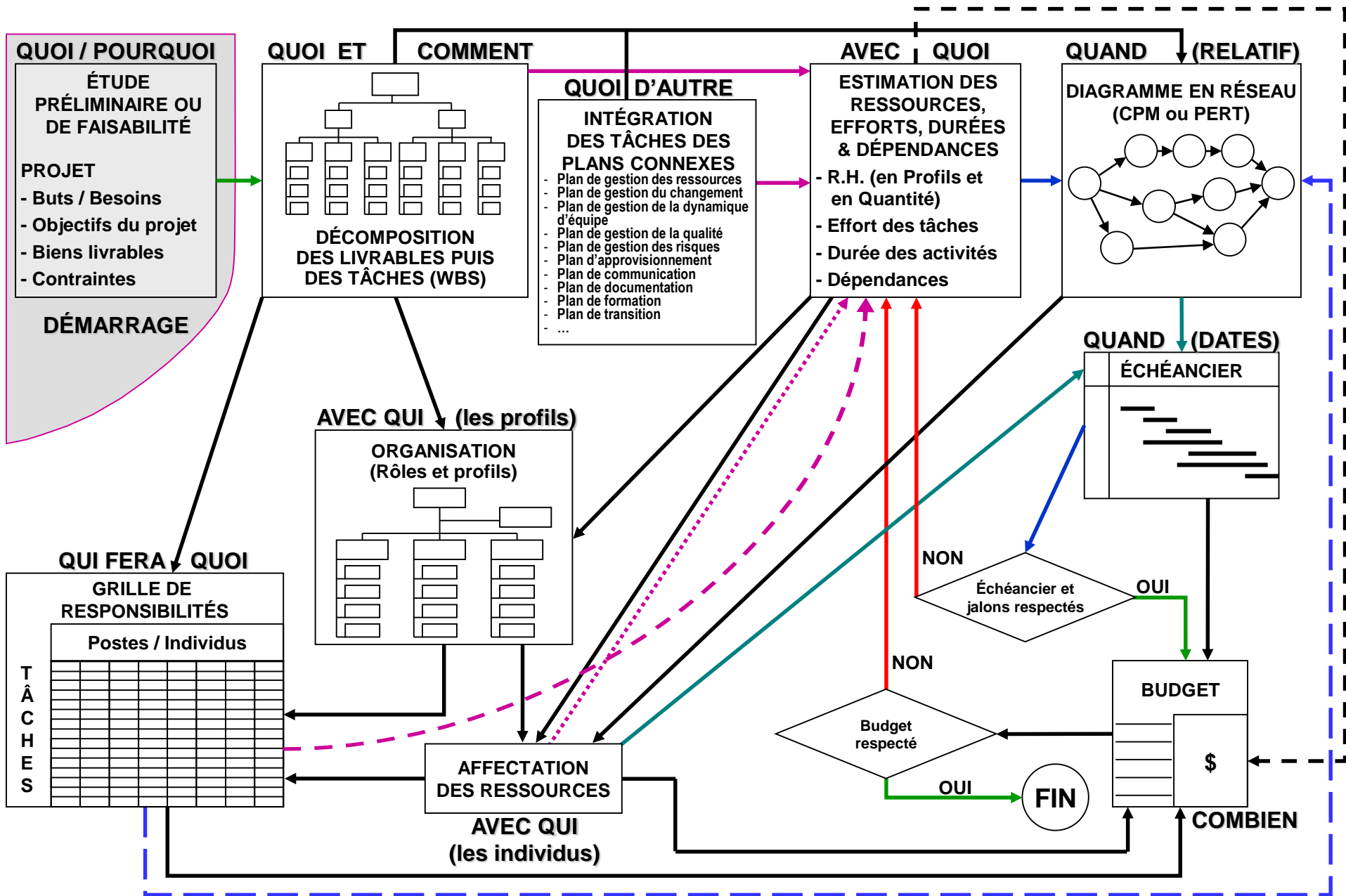


# Processus de planification

# Introduction

- La planification de projet comprend les **étapes** et les **activités** nécessaires pour s'assurer que le projet contient tout le travail requis, **et uniquement celui-ci**, pour assurer la bonne fin du projet. Il se concentre avant tout sur la **définition** et la **maîtrise** de ce qui fait ou non partie du projet. De plus, **l'estimation** et **l'ordonnement** dans le temps des activités en font aussi partie pour **planifier** le déroulement du projet.
  - Voir « Groupe des processus de planification » dans le PMBOK

# SURVOL D'UN PROCESSUS DE PLANIFICATION (HEC, mtl)



# Planification – Un préambule (HEC, MTL)

- Tout d'abord, s'assurer de résoudre le vrai problème – Quoi et Pourquoi (s'assurer que le démarrage a bien couvert tout cela)
  - **Bien comprendre le projet**
    - Le positionner par rapport à tout ce qui se vit dans l'organisation
    - Comprendre les besoins réels que l'on veut combler
    - En cerner les objectifs tangibles et moins intangibles
  - **Définir les biens livrables** (extrants)
  - **Préciser ce qui sera fait**, les produits et services qui seront inclus ...**et ce qui ne le sera pas** (exclusions)
  - **Comprendre les enjeux et les risques**, ainsi que notre capacité réelle à réaliser le projet selon les critères du client, puis préciser les mesures préventives à mettre en place
  - **Au besoin, réaliser une étude de faisabilité** (pour un projet interne) ou un équivalent en mode accéléré lorsqu'il s'agit d'une réponse à un appel d'offres
- Puis procéder aux étapes usuelles de planification



## 4 Concepts: *Planification, Estimation, Ordonnancement, Capacité*

- Quelles sont les différences ?
- **Planification:** Identification des tâches et des livrables...pas de date ni efforts.
- **Estimation:** Déterminer les efforts et la durée
- **Ordonnancement:** Définir les dates de début et fin, prédécesseurs/successeurs
- **Capacité:** Assigner les ressources et résoudre les conflits

# Comment faire ?

- 1. Identifier ce qui doit être fait (***Planifier***)
  - *Work Breakdown Structure (WBS)*
- 2. Identifier les efforts & coûts (***Estimation***)
  - Il existe plusieurs techniques
- 3. Identifier les dépendances (***Ordonnancement***)
  - Graphique de dépendance, ordre logique, Méthodologie
- 4. Assigner les ressources (***Capacité***)
  - Le résultat final donnera le pan détaillé !

# La planification – Un programme en 12 étapes !!!

(HEC, mtl)

1. **Décomposer les livrables.**
2. **Décomposer les tâches pour réaliser les livrables du projet (WBS)**
3. **Intégrer les tâches découlant de tous les plans connexes.**
4. **Décrire les ressources requises et en estimer les quantités.**
5. **Évaluer l'effort et la durée de chaque tâche.**
6. **Ordonnancer (faire le « scheduling »).**
  - Identifier les dépendances
  - Mettre en réseau
7. **Établir la structure d'équipe.**
  - Décrire les compétences requises et les profils visés
  - Clarifier les rôles et les interdépendances
  - Préparer la grille des responsabilités
  - Élaborer la structure de l'équipe
8. **Affecter les ressources et/ou en amorcer l'appropriation.**
9. **Estimer les coûts.**
  - Obtenir les coûts unitaires
  - Évaluer les coûts par livrable, par activité, par phase ou par ...
  - Intégrer la contingence d'estimation et la contingence de réalisation
  - Préparer le budget total et le rythme des dépenses (et revenus?)
10. **Élaborer l'échéancier.**
  - Intégrer les jalons et dates critiques
  - Transposer le réseau en échéancier
  - Nivelier les ressources au besoin
11. **Vérifier qu'on ne dépasse pas le budget... sinon revoir le tout.**
12. **Vérifier qu'on ne dépasse pas la date cible... sinon revoir le tout.**

# Concept: Planification du projet

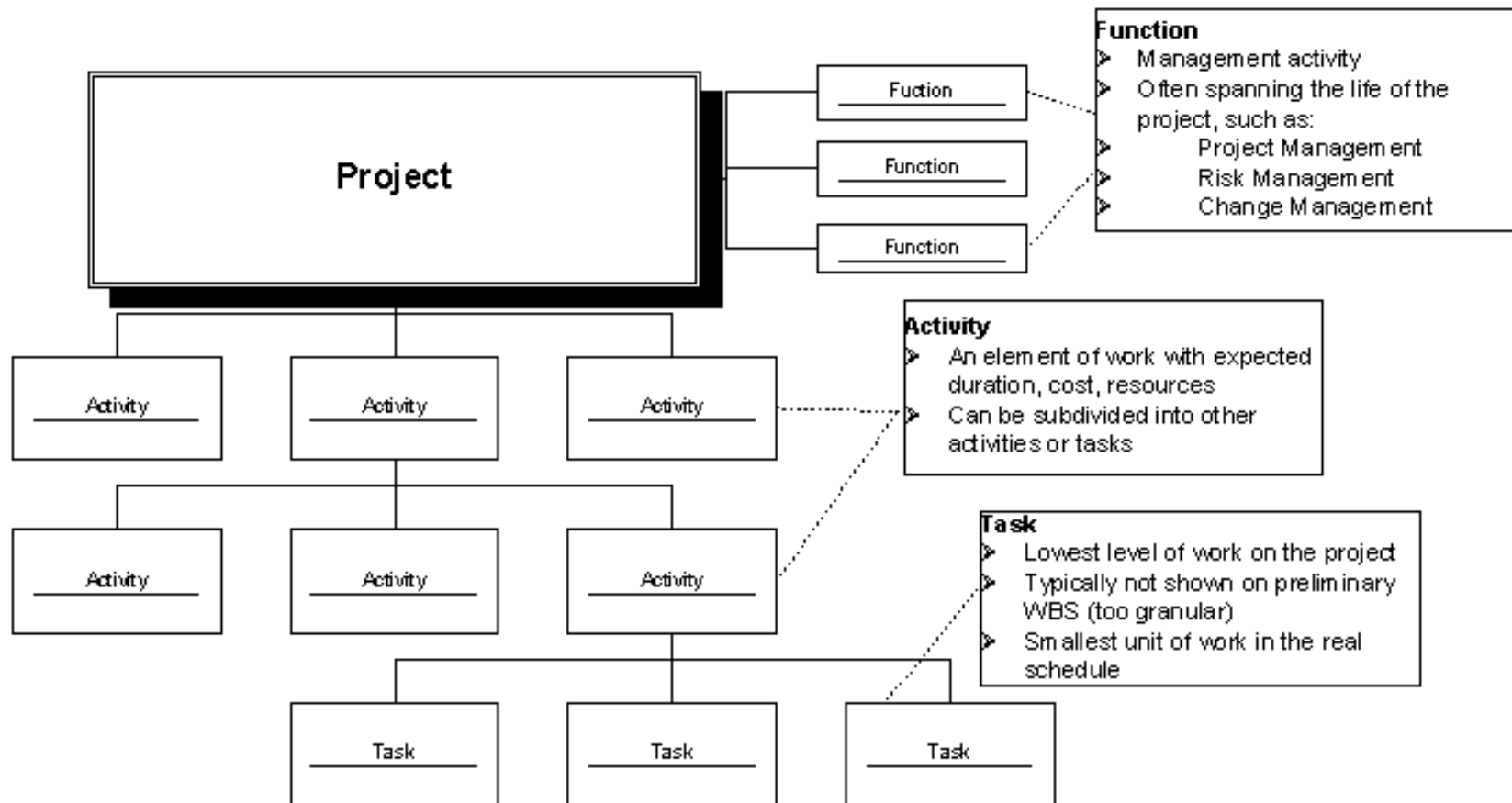
- Cas: Projet type
- Comment réagir à la question ?
  - “Combien de temps prendra le projet ?
- Difficile de répondre...
- Comment approcher le problème?

# Il faut décomposer le projet

- Il faut décomposer les tâches et livrables en activités d'une taille suffisamment détaillée pour être **capable d'estimer** les efforts !
- TOUS les projets requièrent ce travail
- Approcher “Diviser pour régner” (« *Divide and conquer* »)
- Diverses causes d'échec et dépassement de budget
  - Oubli d'une activité critique
  - Des estimés approximatifs deviennent les estimés finaux
  - Architecture détaillée incomplète ou étape carrément oubliée (vite il faut commencer à coder...)

# Décomposition d'un projet (Source: Columbia university)

- Projet: Processus, Activités, Tâches, Livrables





# La Planification

## WBS (Structure de découpage)

# Creating the Work Breakdown Structure (WBS)

- A **WBS** is a deliverable-oriented grouping of the work involved in a project that defines the total scope of the project
- WBS is a foundation document that provides the basis for planning and managing project schedules, costs, resources, and changes
- **Decomposition** is subdividing project deliverables into smaller pieces
- A **work package** is a task at the lowest level of the WBS



# Work Breakdown Structure: WBS

- Décomposition hiérarchique d'un projet
- **2 Formats**
  - **Structuré** (Outline *indented format*)
  - **Arbre Graphique** (Ex: Organigramme)
- Utilise souvent une nomenclature numérique
  - Ex: 3.1.5
  - 0 est le "Top Level" du projet
- Concept maître
  - **Inclus 100% du travail à faire en tout temps**
- Présente la hiérarchie du projet
- Ne présente pas les dépendances ou les durée (pas un plan de ressources)

# Types de WBS

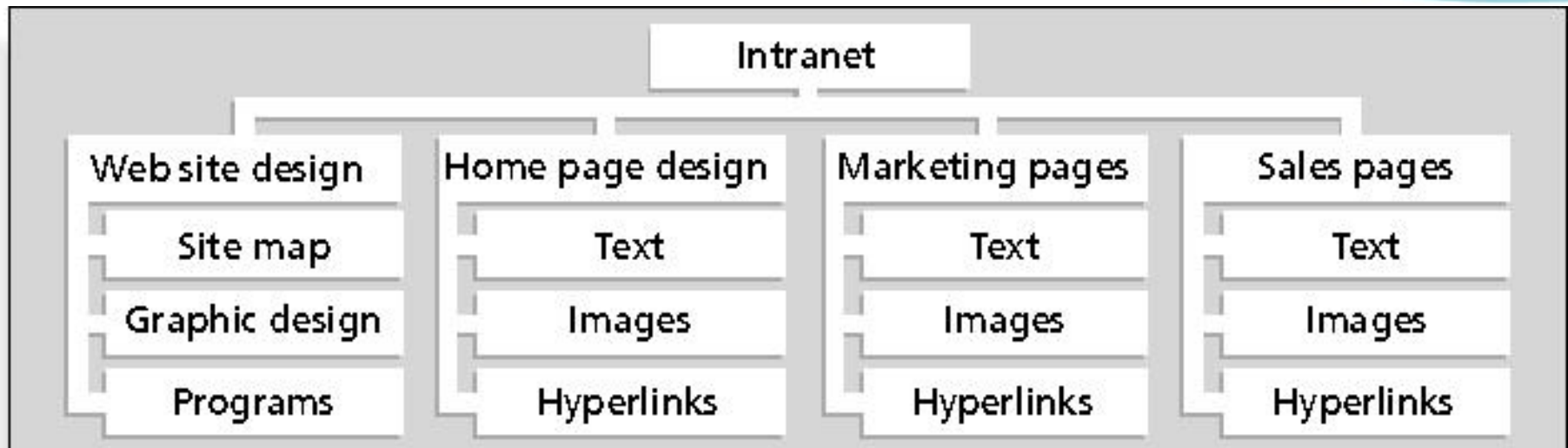
- **Processus**

- Orienté sur les activités
- Ex: Besoins, Architecture, Analyse, ...

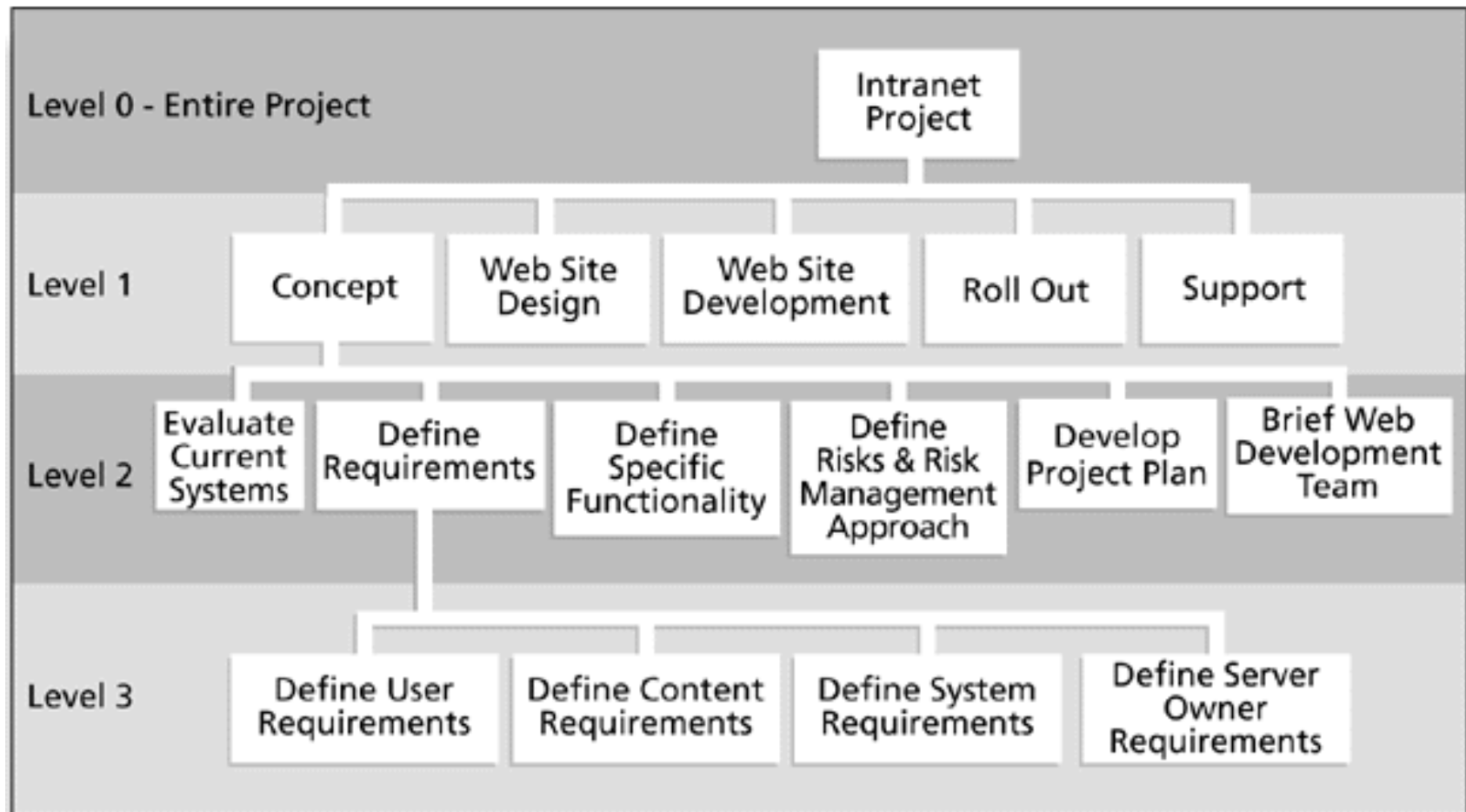
- **Produit**

- Orienté sur le produit (objet)
- Ex: Interface à un système externe

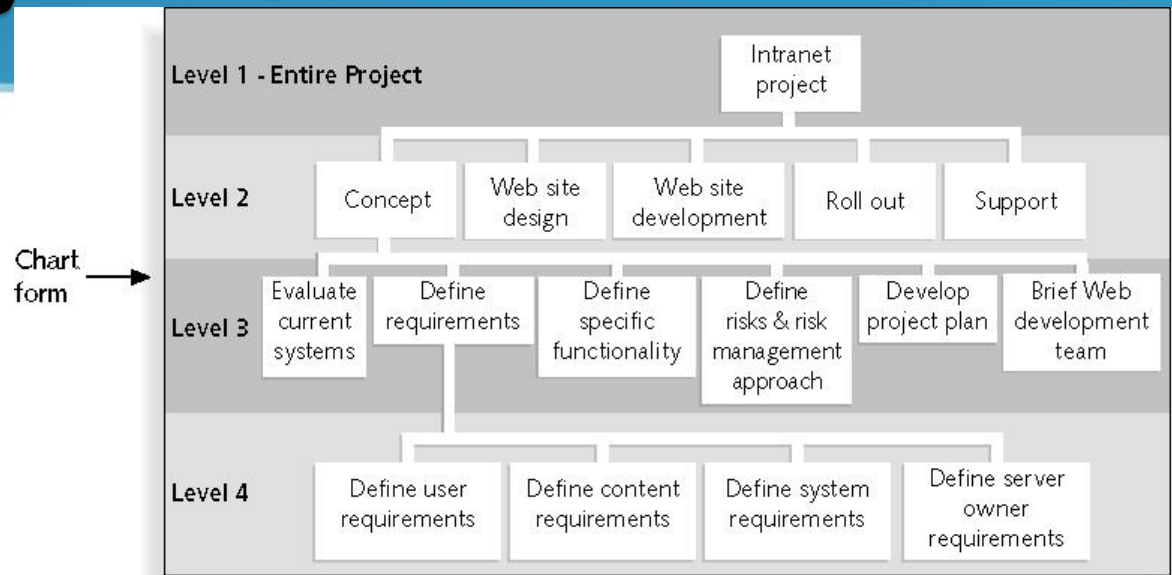
# Figure 5-3. Sample Intranet WBS Organized by Product



# WBS de processus



# Figure 5-4. Sample Intranet WBS Organized by Phase



Tabular form with Microsoft Project numbering

- 1.0 Concept
  - 1.1 Evaluate current systems
  - 1.2 Define requirements
    - 1.2.1 Define user requirements
    - 1.2.2 Define content requirements
    - 1.2.3 Define system requirements
    - 1.2.4 Define server owner requirements
  - 1.3 Define specific functionality
  - 1.4 Define risks and risk management approach
  - 1.5 Develop project plan
  - 1.6 Brief Web development team
- 2.0 Web site design
- 3.0 Web site development
- 4.0 Roll out
- 5.0 Support

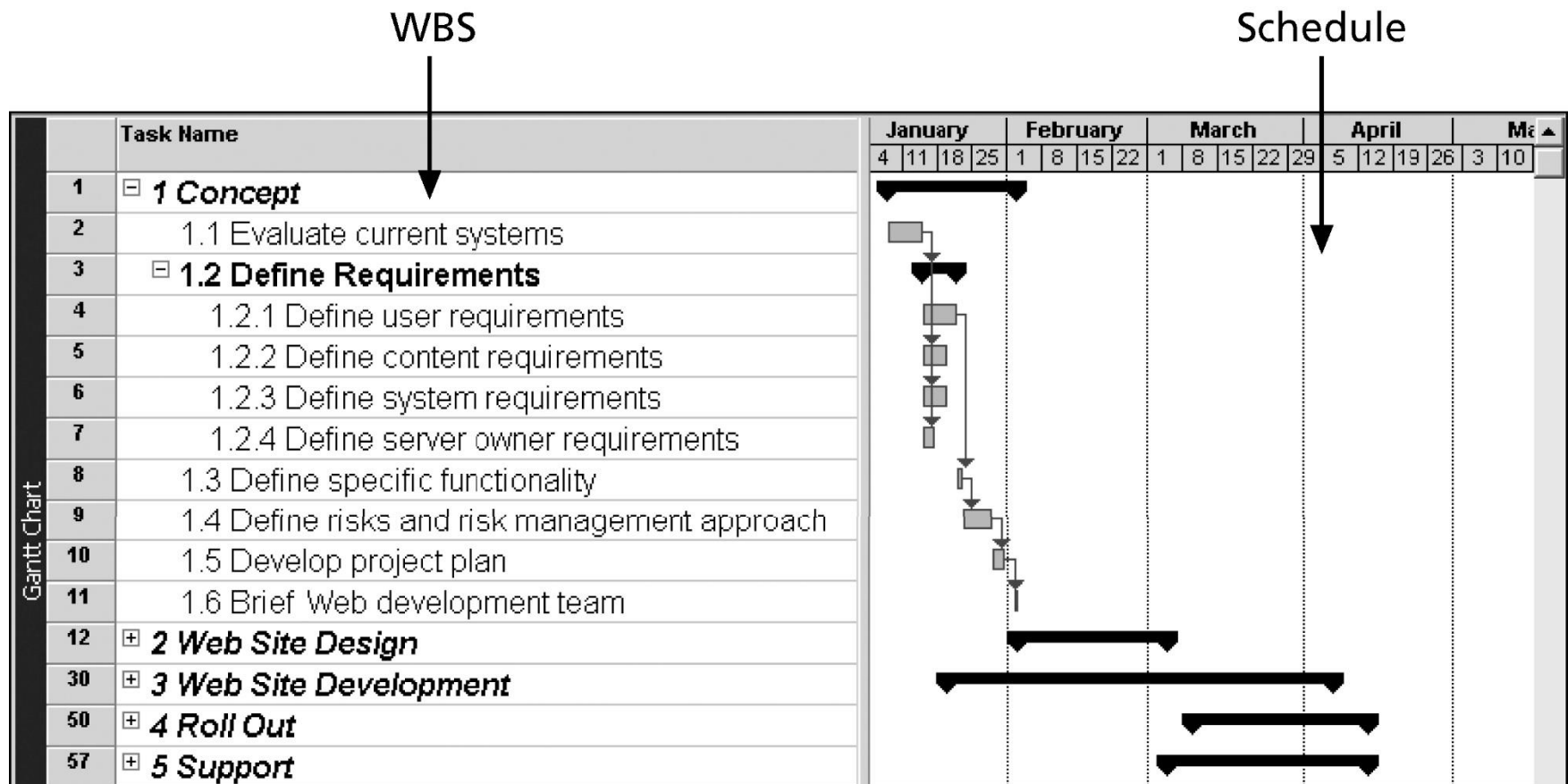
Tabular form with PMI numbering

- 1.1 Concept
  - 1.1.1 Evaluate current systems
  - 1.1.2 Define requirements
    - 1.1.2.1 Define user requirements
    - 1.1.2.2 Define content requirements
    - 1.1.2.3 Define system requirements
    - 1.1.2.4 Define server owner requirements
  - 1.1.3 Define specific functionality
  - 1.1.4 Define risks and risk management approach
  - 1.1.5 Develop project plan
  - 1.1.6 Brief Web development team
- 1.2 Web site design
- 1.3 Web site development
- 1.4 Roll out
- 1.5 Support

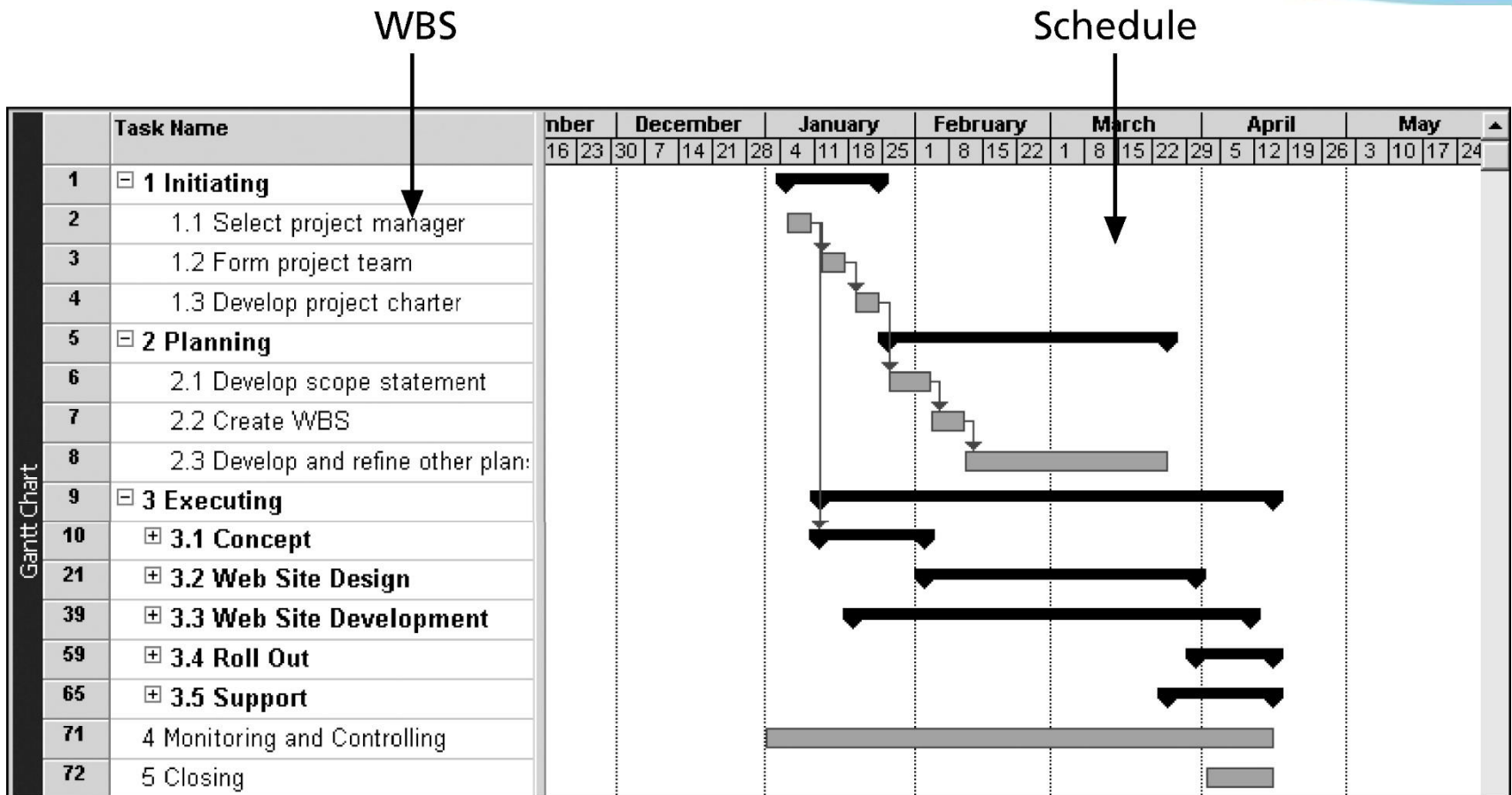
# Les processus du PMP

- Initiating
  - Démarrage, charte de projet,
- Planning
  - Portée, préciser les objectifs et les besoins, 1<sup>er</sup> version des documents de projet
- Executing
  - Réalisation du projet (architecture, conception), livrables, déploiement
- Monitoring & controlling
  - Suivre l'avancement, gérer le changement, traiter les requêtes, travaux des comités
- Closing
  - Fermer le projet, rapport bilan

# Figure 5-5. Intranet WBS and Gantt Chart in Microsoft Project



## Figure 5-6. Intranet Gantt Chart Organized by Project Management Process Groups





# Table 5-4. Executing Tasks for JWD Consulting's WBS

## 3.0 Executing

### 3.1 Survey

### 3.2 User inputs

### 3.3 Intranet site content

#### 3.3.1 Templates and Tools

#### 3.3.2 Articles

#### 3.3.3 Links

#### 3.3.4 Ask the Expert

#### 3.3.5 User requests feature

### 3.4 Intranet site design

### 3.5 Intranet site construction


### 3.6 Site testing

### 3.7 Site promotion

### 3.8 Site roll out

### 3.9 Project benefits measurement

---



# Décomposition des livrables et des tâches

# Décomposition des livrables (HEC, MTL)

- **Vise à décomposer les biens livrables en éléments de moindre envergure, reflétant soit des biens livrables intermédiaires, soit des parties d'un tout pouvant par contre être produites séparément**
- **Une autre approche de décomposition vise à définir les fonctionnalités plutôt que les composantes...**

# Décomposition des livrables (HEC, MTL)

- **La décomposition des tâches :**
  - il faut savoir *s'arrêter à un niveau de détail qui est proportionnel à la complexité et à la taille du livrable.*
  - on peut ensuite effectuer une décomposition des tâches pour chacune des composantes et/ou chaque unité, selon le niveau de décomposition atteint.
- **Quelques avantages**
  - Complète l'approche usuelle et minimise les risques d'oublier un livrable ou une composante clé.
  - Supporte le processus de gestion des approvisionnements.
  - Supporte aussi le processus de gestion de la qualité en permettant de préciser les critères de qualité et de performance de chaque livrable/unité/composante à produire.
  - Facilite un regroupement des activités en unités de travail qui fourniront des livrables pertinents aux yeux du client.

# Décomposition des livrables (exemple)

(HEC, MTL)

- Confirmer le(s) livrables du projet
  - Véhicule de transport
- Décomposer chaque livrable en unités pouvant être livrées de façon distincte, même s'il s'agit d'une livraison « interne » et non au client
  - Véhicule de transport
    - Chassis
    - Propulsion
    - Mobilité
    - Support
    - Etc.

# Décomposition des livrables (suite)

(HEC, MTL)

- Décomposer chaque unité en ses composantes
  - Véhicule de transport
    - Châssis
      - » Armature du véhicule
      - » Mécanisme d'attache au module de support
    - Propulsion
      - » Sources d'énergie
      - » Moteur
    - Mobilité
      - » Roues
      - » Essieux
      - » Mécanisme d'alignement
    - Support
      - » Armature de support du module
      - » Mécanisme d'attache
    - Etc.
  - Décomposer chaque composante en ses diverses sous-composantes, pièces ou modules

# Décomposition des tâches (HEC, Mtl)

- Identique en approche à la décomposition des livrables
- Vise à identifier toutes les tâches nécessaires afin:
  - d’atteindre les objectifs du projet
  - de produire tous les biens livrables convenus
- Cette décomposition peut être effectuée :
  - de façon intuitive
  - pour chacun des livrables
  - de façon structurée sans passer par les livrables...  
si on a l’expérience requise du domaine d’activité

# WBS...quelques conseils

- Une liste d'activités et non de petits items (To do list...)
- Source des WBS
  - Cahier de charges, Proposition, brainstorming, clients, équipe de projet
- Utiliser un court texte pour les items du WBS
  - “Bullet language”
  - Direct et concis
- Ne planifier pas plus d'items que vous êtes capable d'en gérer...



# Techniques WBS

- Top-Down
- Bottom-Up
- Analogy
- Rolling Wave
  - 1<sup>ere</sup> passe: de 1 à 3 niveaux dans un 1er temps
  - Accumuler plus de besoins
  - Ajouter des détails plus tard...
- Post-it sur le mur...

# Techniques WBS

- Top-down
  - Débute au plus haut niveau
  - On développe ensuite le niveau de détail
  - Avantages
    - Problématique connue
    - La méthodologie et la technologie connues
    - Possible de relier à un projet ou problème antérieur
  - S'applique à la majorité des situations...

# Techniques WBS

- Bottom-up
  - Démarre au plus bas niveau de tâches
  - Cumuler ensuite vers de niveaux supérieurs
  - Désavantages
    - Prends du temps...car dans le détail...
    - Requier des besoins complets...et détaillés
  - Avantage
    - Justement...le DÉTAIL

# Techniques WBS

- Analogie
  - WBS qui s'inspire d'un projet antérieur
  - Un gabarit en quelque sorte
  - Peut aussi servir de base à l'estimation
  - Avantages
    - S'inspire des succès du passé (expérience antérieure)
  - Désavantages
    - Il faut trouver des projets comparables...difficiles parfois...

# Cas A&D Tech

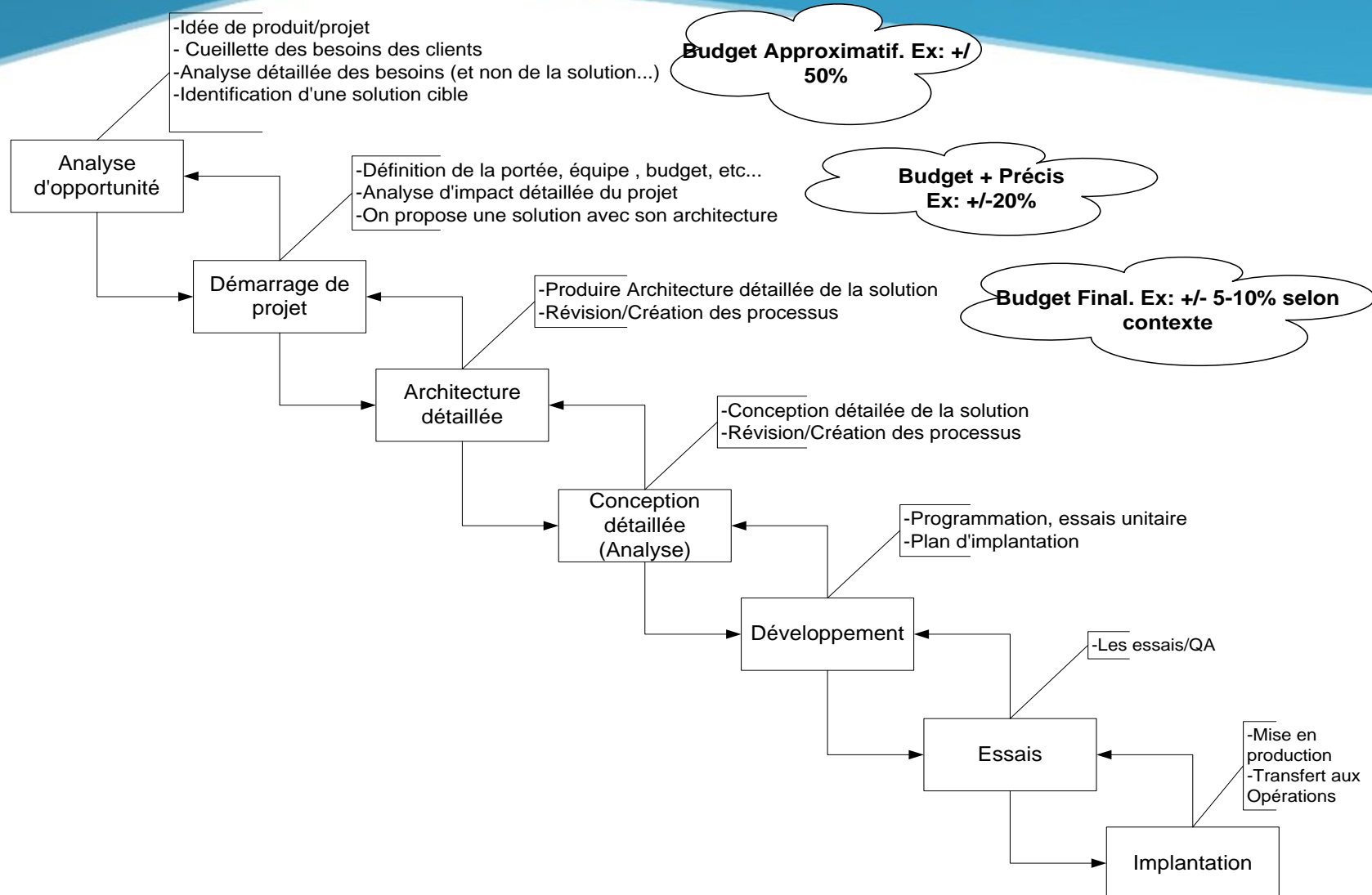
- Travail à faire
  - En utilisant le gabarit de **Software Engineering**, découper le projet et produire un WBS par PROCESSUS
  - En équipe de Maximum 4
  - Découper les 3 premiers niveaux:
  - Exemple: Projet...Architecture...Architecture fonctionnelle...
  - Nous reviendrons en plénière pour revoir les résultats

# Départ

N°	N° hiérarchique	Task Name
	1	1 Overall Project
	2 1.1	Project Management
	4 1.2	System Requirements
	10 1.3	Software Requirements
	21 1.4	Detailed Design
	30 1.5	Test Planning
	34 1.6	Technical Infrastructure
	40 1.7	Development & Unit Test
	50 1.8	Testing
	56 1.9	Deployment

# Solution

# Les activités...





# La QUESTION....

- ***“I know I may be asking a lot here, but I really need you to get the plan together in the next week. Walter really wants to know if we can get this thing done by Christmas.”***

# Les grands blocs

1. Démarrage et Gestion
2. Prise des besoins
3. Architecture Solution
4. Architecture technologique & Infrastructure
5. Analyse détaillée
6. Développement et essais unitaire
7. Les Essais
8. Implantation

# 1-Démarrage et gestion

- Scope (Charte de projet)
- Plan communication
- Comités de gestion
- Gestion de projet
  - Risques
  - Suivi d'avancement
- Planifier le projet
- Rencontre de démarrage

## 2-Besoins détaillés

- Besoins fonctionnels en détail
- Priorisation\*\*\*
- Besoins techno
- Besoin intégration
- Besoins en ressources
- Processus d'affaire
- Besoins de formation
- Marketing (Besoins)

# 3-Architecture solution

- Fonctionnel
  - Maquettes
  - Use Case détaillés (UT)
- Architecture Intégration. ERP
- Arch. Organique
- Architecture des Données

# 4-Architecture techno et Infra

- Environnements – design et installation
- Framework détaillé & Normes
- Architecture sécurité
- Stratégie de capacité
- Relève

# 5-Analyse

- Analyse fonctionnelle détaillée
  - 1-Profil Client
  - 2-Recherche de produit au catalogue
  - 3-Mise à jour du panier d'achat
  - 4-Paiement
  - 5-Soumettre la commande
  - 6-Voir commande et statut
- Structure de données
- Analyse organique & Sécurité
- Analyse intégration système ERP

# 6-Développement et essais unit.

- Accueil développ. Et formation
- Développement
  - 1-Profil Client
  - 2-Recherche de produit au catalogue
  - 3-Mise à jour du panier d'achat
  - 4-Paiement
  - 5-Soumettre la commande
  - 6-Voir commande et statut
- Gestion erreur
- Strat. Essais unit
- Déve. Interface
- Documentation
- Stratégie de support & Guides de support



# 7-Les Essais

- Stratégie d'essais & Relève
- Sécurité (Essais)
- Documentation et Exécution des essais
  - Fonctionnel et performance
  - Assemblage et Fonctionnel (Client)
- Corrections
  - Optimi. & bug fix

# 8-Implantation et fermeture

- Stratégie de déploiement et maintenance
- Formations
- Déploiement
- Post implantation (support 60 jours)
- Transfert en exploitation (autre équipe)
- Bilan de projet et leçons acquises

# L'équation de planification du Chargé de projet

- Qu'est-ce que nous allons planifier pour les ressources en informatique ?
- Typiquement des jours-personne
- Mesure du travail à accomplir dans le cadre d'un projet
- 5 Jours-personnes=Une semaine de travail pour une personne (+/-220 dans une année...)

# Relation Durée-efforts-unités

- $\text{Durée} = \text{Efforts} / \text{unités}$
- Mesure de la durée d'une tâche (en jours....)
- Efforts: Jours personnes requis pour effectuer une tâche
- Unités: Disponibilité de la ressource
- Exemple: Une tâche d'analyse fonctionnelle requiert 10 J/p d'efforts mais l'analyste n'est disponible qu'à 25% de son temps...Quelle sera la durée ?
- **$\text{Durée} = E / U = 10 / 0.25 = 40 \text{ Jours ! (calendrier)}$**

# Conclusion

- Vous êtes le chien de garde de la portée
- Si la portée change, vous devez le soulever rapidement
- Les besoins doivent être expliciter et prioriser
  - Facteur clef pour le succès du projet
- La création du WBS est une activité difficile, organisez-vous pour avoir de l'aide!

# Prochain cours

- Ordonnancement
- TP2
- Lecture : Chapitre 7 – Cost management
- Trouver un logiciel pour faire un gantt
  - MS Project
  - Gantt project