



# **Rappel sur les cycles de vie et les méthodes de conception**



**Partie 1**



**Mr Sellami K  
Informatiqueii9@gmail.com**

# Intitulé de la matière : Atelier de génie logiciel

- **Contenu de la matière :**
  - **Chapitre 1** : Rappel sur les cycles de vie et les méthodes de conception
  - **Chapitre 2** : Ateliers de Génie Logiciel (AGL)
  - **Chapitre 2** : Analyse de la qualité du logiciel
  - **Chapitre 4** : Ingénierie des Besoins
  - **Chapitre 5** : Test et maintenance des logiciels

# Comparaison avec le génie civil

D'après C. Chartier-Kastler "Précis de la conduite de projet informatique"

- **Génie** fait directement référence à celui de **Génie Civil** : l'art de la construction
- Pour construire un ouvrage architecturale, le seul fait de poser brique et ciment ne suffit pas

# Comparaison avec le génie civil

La construction d'un bâtiment est un ensemble d'activités :  
Conception architecturale,  
maçonnerie, plomberie,  
électricité, . . .  
le tout coordonné



# Comparaison avec le génie civil

## Quelques questions ...

- – Quelles tâches accomplir ? Dans quel ordre ?
- – Qui faire intervenir ? Comment les synchroniser ?
- – Quels outils utiliser ? Quand ? Avec des experts ?
- – Quels plans réaliser ? Ont-ils des liens entre eux ?
- – Existe-t-il des contraintes administratives ?
- – Comment suivre les opérations ?
- – Comment effectuer la recette ?
- – Quand payer ?
- – ...

## De nombreuses réponses !

- – Plan d'occupation des sols
- – Permis de construire
- – Architecte
- – Plan d'ensembles, puis techniques (eau, gaz, électricité)
- – Nommer un maître d'oeuvre
- – Avoir un plan de financement
- – Prévoir un point hebdomadaire
- – Avoir des engins de chantiers
- – ...

# Génie Logiciel : c'est quoi ?

- **Génie Logiciel**
- Le terme génie logiciel désigne l'ensemble des méthodes, des techniques et outils concourant à la production d'un logiciel de qualité avec maîtrise des coûts et délais.
- **Logiciel**
- Le logiciel est l'ensemble des programmes et de documents nécessaires à leur installation, utilisation, développement et maintenance.

# Génie Logiciel : Solution à la crise du logiciel

- Le GL est apparu à la fin des années 60 pour répondre à la "**la crise du logiciel**".
- **Symptôme de cette crise :**
  - Le logiciel livré ne satisfait pas les besoins de l'utilisateur,
  - il consomme plus de ressources que prévu et il est à l'origine de pannes.
  - Temps de réponse trop lents.
  - Non respect des délais et des coûts
  - Maintenance difficile, coûteuse et souvent à l'origine de nouvelles erreurs.

# Génie Logiciel : Solution à la crise du logiciel

- Le GL est apparu à la fin des années 60 pour répondre à la "**la crise du logiciel**".
- **Symptôme de cette crise :**
  - Le logiciel livré ne satisfait pas les besoins de l'utilisateur,
  - il consomme plus de ressources que prévu et il est à l'origine de pannes.
  - Temps de réponse trop lents.
  - Non respect des délais et des coûts
  - Maintenance difficile, coûteuse et souvent à l'origine de nouvelles erreurs.

# Quelques erreurs célèbres

- Entre 1985 et 1987, le problème du Therac-25 (un appareil d'irradiation thérapeutique) : 2 morts, 4 irradiés.
- En 1994, Le bug du Pentium. Coût : 500 millions de dollars.
- le 4 juin 1996, l'explosion du premier vol d'Ariane 5, le bug informatique le plus coûteux de l'histoire.
- En 1962, la mission Mariner 1 a connu un incident similaire
- TAURUS, un projet d'informatisation de la bourse londonienne : définitivement abandonné après 4 années de travail et 100 millions de £ de pertes
- Le bug de l'an 2000 : dysfonctionnements lorsque les dates sont postérieures au 31 décembre 1999.

# Génie Logiciel : Difficultés

- **Difficultés liées à la nature du logiciel**
  - un logiciel ne s'use pas, sa fiabilité ne dépend que de sa conception
  - pour rester utilisé un logiciel doit évoluer (changements fréquents)
  - ...
- **Les difficultés liées aux personnes**
  - ne savent pas toujours ce qu'elles veulent, ou ne savent pas bien l'exprimer
  - communication difficile entre personnes de métiers différents (jargons)
  - beaucoup d'autodidactes qui croient savoir,
  - ...
- **Les difficultés technologiques**
  - courte durée de vie du matériel,
  - beaucoup de méthodes et de langages
  - évolution des outils de développement.

# Génie Logiciel : le pourquoi?

- **Systemes informatiques**
  - 80 % de logiciel
  - 20 % de matériel
- **Les problèmes liés à l'informatique sont essentiellement des problèmes de Logiciel**
  - Le matériel est relativement fiable
  - Le marché est standardisé

# Réussite/échec

- **Les besoins du client sont imprécis et changeants :**
  - Faut-il faire un effort pour préciser et figer le besoin du client en début de projet ?
  - Faut-il développer de manière à être tolérant aux imprécisions et aux changements de besoins ?
- 2 pratiques différentes du génie logiciel s'opposent sur la manière de traiter ces 2 problèmes.

# Pourquoi avons-nous besoin d'une méthode de développement?

- Pour **l'organisation** : intégrer à son système de gestion un logiciel (acheté ou développé sur mesure)= investissement important, **rentable**???
- Beaucoup de **risques**: techniques , fonctionnels, (des fois organisationnels).

• **→ Une méthode de développement de logiciel** est une façon de faire , convenue définie en activités et tâches , adaptée au contexte du travail pour produire, entretenir un logiciel de qualité.

• **→ Objectifs** : produire un logiciel qui répond aux exigences (exprimées par le client).

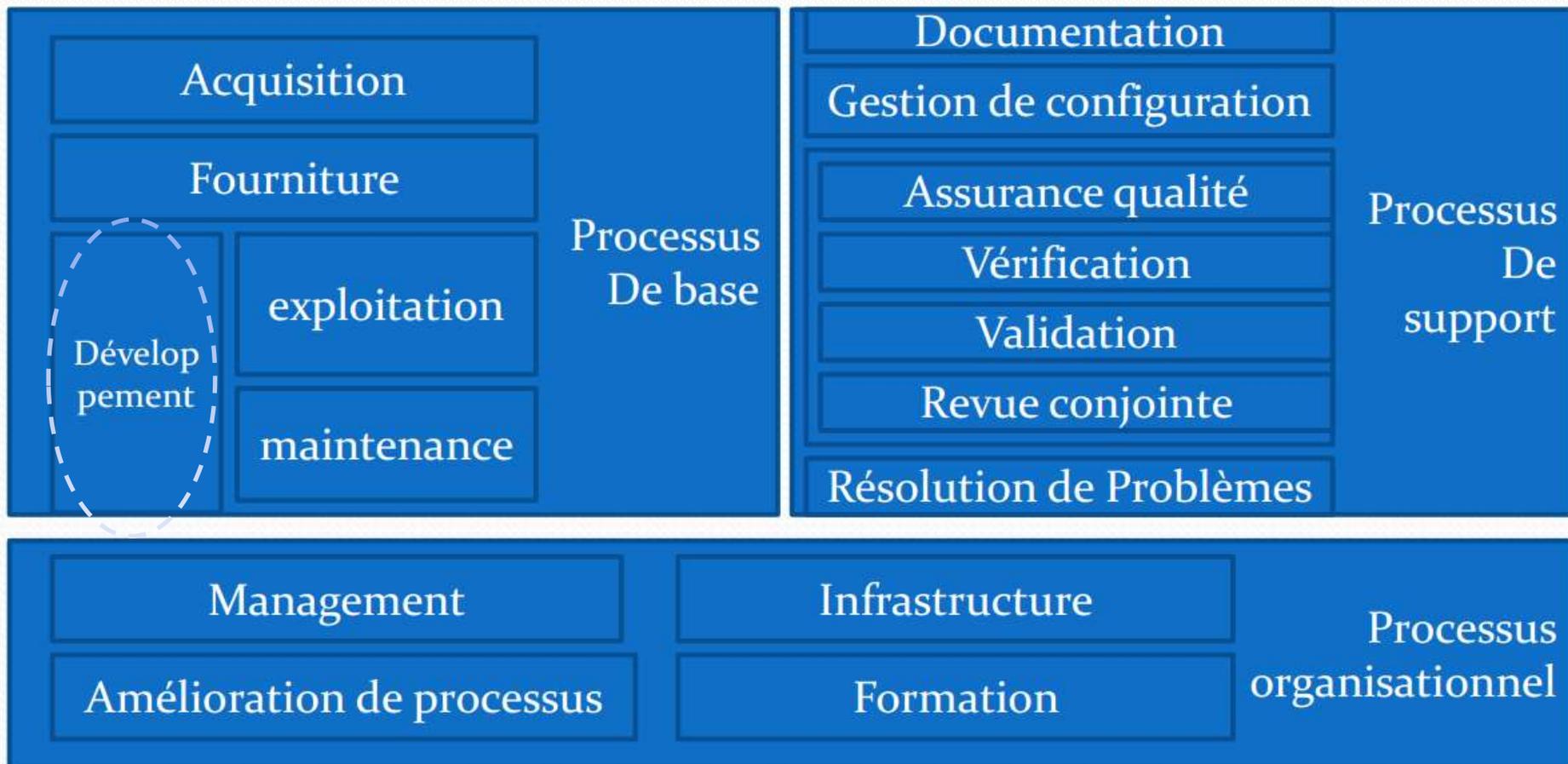
# Pourquoi avons-nous besoin d'une méthode de développement?

- Dans la plupart des temps , les projets dépassent à la fois
  - Délais
  - Budgets
- Erreurs de fonctionnement lors de l'installation initiale.

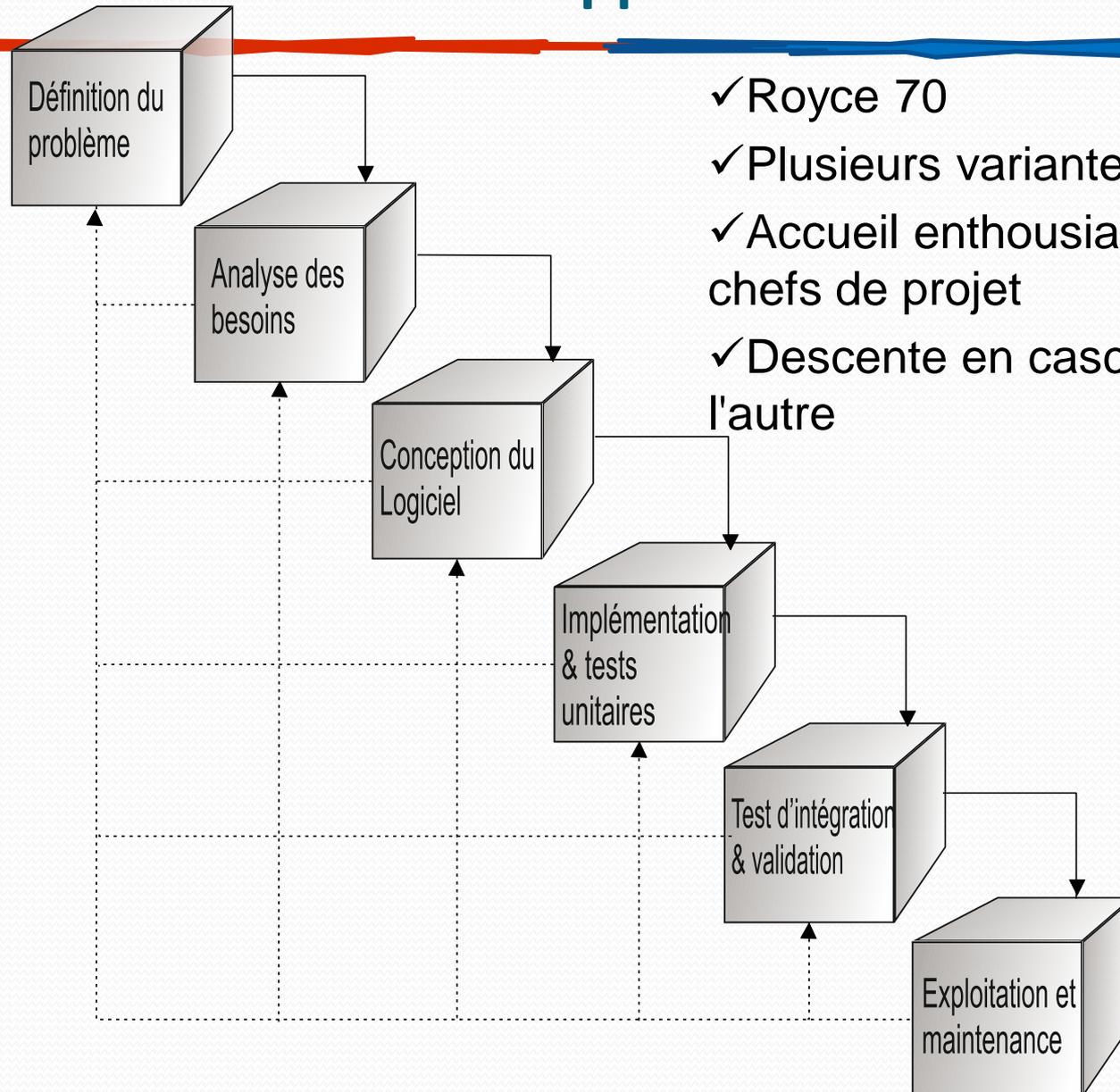
 **Cause: manque d'une bonne méthode de développement de logiciel.**

# Processus de développement de logiciel

- Une méthode complète couvre tout le cycle de vie du logiciel



# Les modèles de développement: modèle de la cascade



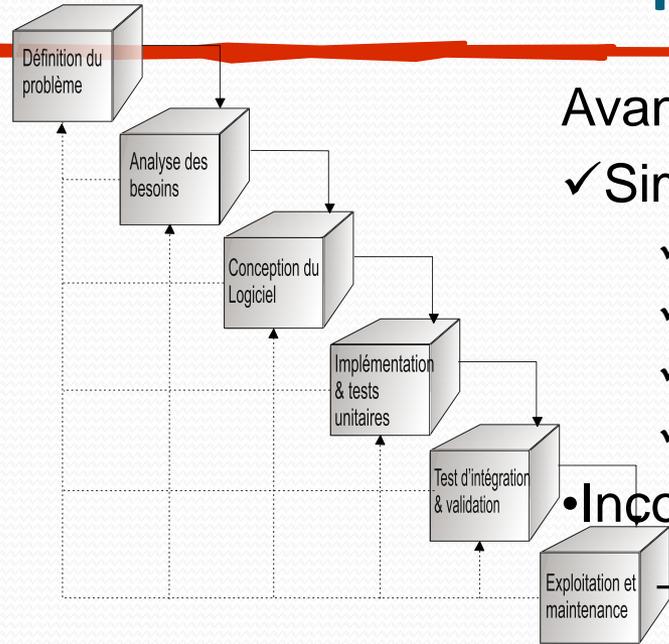
✓ Royce 70

✓ Plusieurs variantes

✓ Accueil enthousiaste de la part des chefs de projet

✓ Descente en cascade d'une étape à l'autre

# Les modèles de développement: modèle de la cascade



## Avantages :

### ✓ Simplicité

- ✓ Def & planif plus précises
- ✓ Danger détectés à l'avance
- ✓ Responsabilité claires
- ✓ Respect Échéances

### • Inconvénients

- Flexibilité ✗
- Retour en arrière ✗
- Risque de déception du client
- Cout des Changements ↑

# Les modèles de développement: modèle de la cascade

## ÉTAPES TYPES DU CYCLE DE VIE TRADITIONNEL

### •a. Définition du problème

#### •Actions :

- On identifie le problème (quel est-il ? pour qui ? pourquoi veut-on le résoudre ?)
- Et on décide ou non de la mise en place d'un projet logiciel.

#### •Moyens:

- Traitements de textes, tableurs

#### •Délivrables en sortie:

- Ouverture d'un dossier projet
- Identification du maître de l'ouvrage et du maître de l'œuvre

#### •Intervenants:

Les intervenants sont fort peu nombreux, on distingue essentiellement:

- des consultants qui peuvent être
  - internes, c'est à dire appartenant à l'organisation considérée.
  - externes, c'est à dire n'appartenant pas à l'organisation mais plutôt à un cabinet de conseil extérieur. Et cette solution est souvent meilleure surtout si une prise de recul est nécessaire.
- Aucune structure de type comité directeur, groupe de projet, groupe d'utilisateurs n'est mise en place car il n'y a pas encore de projet logiciel.

#### •Remarque:

- Le projet décidé peut faire partie d'un ensemble de projets planifiés. Il est alors inscrit dans le plan informatique de l'E (entreprise ou organisation). Il est bon de rapprocher les objectifs du projet décidé des objectifs stratégiques de l'E et de voir s'il entre dans le cadre général du plan de développement de l'E.



# Les modèles de développement: modèle de la cascade

## ÉTAPES TYPES DU CYCLE DE VIE TRADITIONNEL

### •*b. Analyse des besoins*

#### •Actions :

- Le concepteur recense et étudie les besoins du client (maître de l'ouvrage) c'est à dire :
- les fonctions que le logiciel doit effectuer
- les conditions d'exploitation
- le contrat de service
- la qualité requise (performances, sûreté, fiabilité,...)

#### •Moyens:

- Les différents moyens d'enquête (interviews, sondages, entretiens, consultation de documents, brainstorming, etc.) et ceci en fonction du projet et de l'interlocuteur.
- Langages de description formelle (PDL: Program Design Language), techniques graphiques de modélisation (SADT, E/A,...)
- Outils de maquettage, et autres AGL

#### •Traitements de textes

#### •Délivrables en sortie:

•• Cahier des charges: document qui est un engagement et qui fait partie du contrat de réalisation du projet.

#### •Intervenants:

Les intervenants dans cette phase peuvent être nombreux, et on peut principalement distinguer:

- Le chef de projet qui doit animer, orienter et veiller au respect des contraintes: Délais imposés, Qualité requise, Coûts à minimiser.
- Les consultants qui apportent de l'aide et de l'expérience
- Les experts qui permettent de vérifier ou valider la pertinence d'une solution.
- Les utilisateurs qui expriment leurs besoins, leurs attentes et souhaits.
- Le comité directeur qui décide du choix de l'orientation à prendre,

# Les modèles de développement: modèle de la cascade

## ÉTAPES TYPES DU CYCLE DE VIE TRADITIONNEL

### •b. Analyse des besoins

#### •Actions :

- Le concepteur recense et étudie les besoins du client (maître de l'ouvrage) c'est à dire :
- les fonctions que le logiciel doit effectuer
- les conditions d'exploitation

#### **Remarque:**

L'analyse des besoins comprend généralement une analyse de l'existant.

•Mo La validation de cette étape est **EXTREMEMENT IMPORTANTE**. En effet, c'est lors de cette validation que la décision de poursuivre ou d'arrêter le projet sera prise.

De plus, c'est à ce moment que l'on décidera des types de solutions envisageables (achat de progiciel, développement effectué à l'intérieur de l'E ou sous-traités à des SSII, réorganisations, versions successives, etc.).

•Dé On vérifie la faisabilité (économique, technologique, organisationnelle, sociale) et on choisit la solution la plus appropriée.

•• C

#### •Intervenants:

Les intervenants dans cette phase peuvent être nombreux, et on peut principalement distinguer:

- Le chef de projet qui doit animer, orienter et veiller au respect des contraintes: Délais imposés, Qualité requise, Coûts à minimiser.
- Les consultants qui apportent de l'aide et de l'expérience
- Les experts qui permettent de vérifier ou valider la pertinence d'une solution.
- Les utilisateurs qui expriment leurs besoins, leurs attentes et souhaits.
- Le comité directeur qui décide du choix de l'orientation à prendre,

# Les modèles de développement: modèle de la cascade

## ÉTAPES TYPES DU CYCLE DE VIE TRADITIONNEL

### •c. Conception du logiciel

#### •Actions :

- Le concepteur étudie les moyens (informatiques, organisationnels, etc.) nécessaires aux fonctions définies dans le cahier des charges :
  - choix du matériel
  - la structure des données
  - l'architecture des traitements
  - éventuellement les nouvelles organisations du travail (définition des postes, formations à mettre en place, etc.) et les nouvelles procédures de travail.

#### •Moyens:

- Outils de conception
- Simulation
- Outils de prototypage
- Méthodes et/ou modèles (Organigrammes, algorithmes, E/A, Pétri, SADT,...)

#### •Délivrables en sortie:

- Le document des spécifications techniques (générales et détaillées) qui précise comment sera réalisé physiquement le futur logiciel.

#### •Intervenants:

Principalement, on peut distinguer:

- Le chef de projet qui gère, négocie, contrôle et coordonne.
- Les analystes qui détaillent les besoins et les solutions retenues
- Les analystes programmeurs qui décrivent d'une façon complète et détaillée les modules.
- Les techniciens qui sont garant de l'architecture technique à mettre en place: matériels, logiciels, et AGL.
- Les experts qui apportent leur compétence.
- Le responsable qualité qui veille à la qualité des travaux réalisés.
- Les utilisateurs pour consultation éventuelle.
- Ainsi que le comité directeur.

# Les modèles de développement: modèle de la cascade

## ÉTAPES TYPES DU CYCLE DE VIE TRADITIONNEL

---

### *d. Implémentation et test unitaire*

#### •Actions :

- Le concepteur élabore les différents programmes et crée les fichiers de données.
- Il procède aux tests unitaires (de premier niveau)

#### •Moyens:

- Environnements de programmation qui supportent un certain nombre de langages
- AGL
- Outils de vérification et de tests

#### •Délivrables en sortie:

- Les programmes, structure des fichiers de données.
- La documentation des programmes et la documentation utilisateur
- les rapports des tests unitaires
- le rapport de programmation (réalisation)

#### •Intervenants:

En plus des intervenants de l'étape précédente, on a:

- Les programmeurs qui procèdent au codage des différents modules.
- Configurateur du système d'exploitation
- Outils de métrologie
- Installateur de corrections et de versions

# Les modèles de développement: modèle de la cascade

## ÉTAPES TYPES DU CYCLE DE VIE TRADITIONNEL

---

### •e. *Exploitation et maintenance*

#### •Actions :

- Réaliser un essai complet sur un morceau majeur de logiciel

#### •Moyens:

- Environnements de programmation qui supportent un certain nombre de langages
- AGL
- Outils de maintenance

#### •Délivrables en sortie:

- Recette définitive
- Contrat de maintenance
- Le registre des incidents
- Le rapport de suivi de maintenance

#### •Intervenants:

- Responsable de projets si la maintenance demande des efforts importants.
- Les analystes
- Les programmeurs
- Les utilisateurs
- Le comité directeur, si la maintenance est importante et de grande ampleur.

# Les modèles de développement: modèle de la cascade

## ÉTAPES TYPES DU CYCLE DE VIE TRADITIONNEL

---

### •e. *Exploitation et maintenance*

#### •Actions :

- Réaliser un essai complet sur un morceau majeur de logiciel

#### •Moyens:

- Environnements de programmation qui supportent un certain nombre de langages
- AGL
- Outils de maintenance

#### •Délivrables en sortie:

- Recette définitive
- Contrat de maintenance
- Le registre des incidents
- Le rapport de suivi de maintenance

#### •Intervenants:

- Responsable de projets si la maintenance demande des efforts importants.
- Les analystes
- Les programmeurs
- Les utilisateurs
- Le comité directeur, si la maintenance est importante et de grande ampleur.