

# Processus de développement ingénierie

## PDL

### 1 - Processus Développement Logiciel

## Abstract

Définition

## Table des matières

<b>1. Introduction au Processus Logiciel .....</b>	<b>2</b>
1.1. Activités Fondamentales du Processus Logiciel .....	3
1.2. Modèles de Processus Logiciel .....	3
1.2.1. Types de Modèles .....	3
1.3. Gestion des Changements .....	4
1.4. Amélioration du Processus Logiciel .....	4
1.4.1. Approches d'amélioration .....	4



# 1. Introduction au Processus Logiciel

Le processus logiciel est une décomposition en activités structurées qui mènent à la production d'un logiciel. La décomposition de problèmes complexes est au cœur de l'ingénierie logicielle, comme le souligne John Ousterhout, qui la considère comme "le problème fondamental en informatique".

## 1.1. Activités Fondamentales du Processus Logiciel

Tout processus logiciel implique cinq activités fondamentales :

1. **Spécification** : Définir ce que le logiciel doit faire
  - Formulation précise des besoins et exigences
  - Élicitation et analyse du besoin
  - Spécification détaillée
  - Validation du besoin (réalisme, cohérence, complétude)
2. **Conception** : Établir le design du logiciel
  - Conversion de la spécification en description structurée
  - Conception architecturale (composants, modules, relations)
  - Conception de base de données
  - Conception d'interfaces
  - Sélection et conception de composants
3. **Implémentation** : Écrire le code
  - Activité individuelle mais non individualiste
  - Deux approches principales :
    - **Programmation tactique** : Focus sur le fonctionnement immédiat
    - **Programmation stratégique** : Focus sur la structure à long terme
  - Recommandation : investir 10-20% du temps en améliorations stratégiques
4. **Vérification et Validation (V&V)** :
  - **Vérification** : Le système est-il construit conformément aux spécifications?
  - **Validation** : Le système répond-il aux besoins de l'utilisateur?
  - Types de tests :
    - Tests de composants (unitaires, intégration)
    - Tests système
    - Tests utilisateur
  - Cycle en V : chaque phase de développement correspond à une phase de test
5. **Évolution** : Changer le système en réponse aux nouveaux besoins
  - Distinction entre développement et maintenance qui tend à s'estomper

## 1.2. Modèles de Processus Logiciel

Un modèle de processus logiciel est une représentation abstraite qui précise les activités, leur ordre, leurs produits, et les rôles impliqués.

### 1.2.1. Types de Modèles

#### 1. Modèles planifiés (en cascade)

- Processus piloté avec planification préalable
- Phases distinctes et séquentielles
- Avantages : adapté aux grands projets, besoins stables, coordination multi-équipes
- Inconvénients : manque de flexibilité face aux changements

#### 2. Modèles incrémentaux (agiles)

- Planification incrémentale, processus adaptable
- Activités entrelacées plutôt que séparées
- Avantages : validation continue, adaptation aux besoins, valeur livrée plus tôt
- Inconvénients : requiert une excellente communication, dégradation potentielle de l'architecture

### 3. Modèles d'intégration et de configuration

- Réutilisation et configuration de composants existants
- Types de logiciels réutilisables :
  - Applications standalone (COTS)
  - Librairies et packages
  - Services web standardisés
- Avantages : rapidité, coûts réduits
- Inconvénients : perte de contrôle, compromis sur les fonctionnalités

### 1.3. Gestion des Changements

Les changements sont inévitables et peuvent être :

- Anticipés via des prototypes
  - Un prototype se concentre sur les inconnues du système
  - Ne devrait jamais être mis en production
  - Bénéfices : meilleure compréhension des besoins, test de variantes
- Tolérés via le développement incrémental
  - Logiciel développé et livré par incréments utilisables
  - Bénéfices : création de valeur progressive, réduction des risques

### 1.4. Amélioration du Processus Logiciel

Les entreprises cherchent à améliorer leurs processus pour :

- Réduire les coûts
- Améliorer la qualité
- Accélérer le développement

#### 1.4.1. Approches d'amélioration

##### 1. Niveaux de maturité

- Initial : processus décrits et communiqués
- Managed : processus mis en œuvre
- Defined : données de mise en œuvre collectées
- Quantitatively managed : performance mesurée
- Optimizing : processus améliorés sur base d'indicateurs

##### 2. Cycles d'amélioration

- Measure : collecte de données quantitatives
- Analyze : identification des faiblesses
- Change : amélioration du processus