



UML: Unified modeling Language

Leçon 5/6 - 9-16/10/2008

Les packages - Modèle d'implémentation - Méthodologie (RUP)

Luigi DILILLO - LIRMM / UM2

dilillo@lirmm.fr

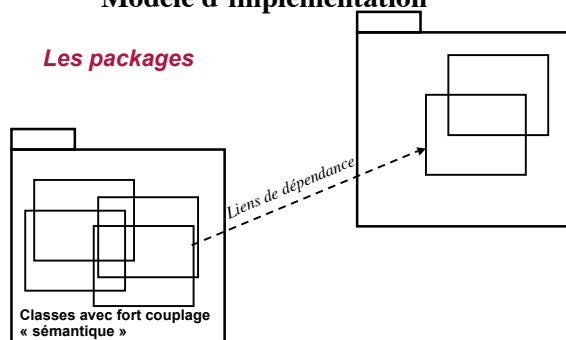
Modèle d'implémentation

Les packages

- Un package ou sous-système est un regroupement logique de classes, associations, contraintes ..
- Un sous-système est généralement défini par les services qu'il rend
- Les liens entre sous-systèmes sont des liens de dépendance
- Les « packages » servent à structurer une application
- Ils sont utilisés dans certains LPO (Java) ce qui assure une bonne « traçabilité » de l'analyse à l'implémentation

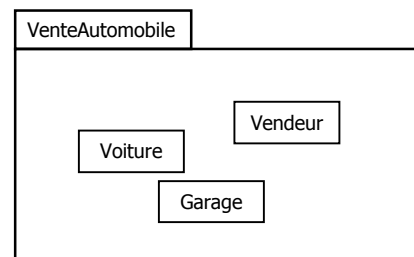
Modèle d'implémentation

Les packages



Classes avec fort couplage
« sémantique »

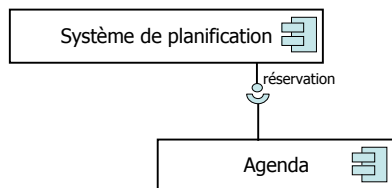
Modèle d'implémentation



Modèle d'implémentation (UML 2.0)

Diagramme de composants

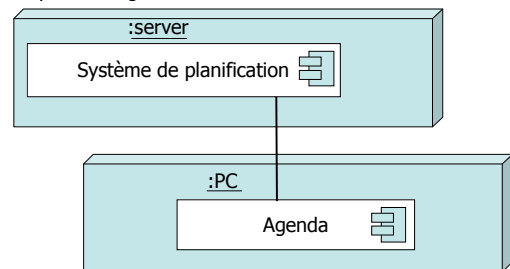
dépendances entre composants logiciels
(sources, binaires, exécutables, etc.)



Modèle d'implémentation (UML 2.0)

Diagramme de déploiement

organisation matérielle des éléments de calcul et des composants logiciels



Plan

- ANNEXES

Méthodologie (RUP - Rational Unified Process)

Méthodologie

Méthode ou Processus de développement

- ✓ acteurs nécessaires (qui)
- ✓ grands types d'activités (comment)
- ✓ artefacts (quoi)
- ✓ organisation du travail (quand)

Historique

- Méthodes cartésiennes
Jackson, SADT, Yourdon
- Méthodes systémiques
Merise, Axial, Information Engineering
- Méthodes objet
OOD, HOOD, OMT, OOSE, OOA/OOD,
unifiées dans le *RUP (Rational Unified Process)*

Concepts généraux

- *Conceptualiser*
obtenir un énoncé du problème (utilisateurs)
- *Analyser*
spécifier le problème
- *Concevoir*
une solution informatique
- *Implémenter*
réaliser la solution informatique

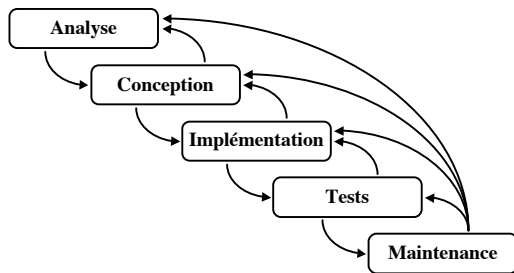
Concepts généraux

Étapes	Résultats
<i>Planification</i>	<i>Schéma directeur</i>
<i>Analyse des besoins</i>	<i>Modèle des besoins</i>
<i>Spécification formelle ou analyse</i>	<i>Modèle conceptuel</i>
<i>Spécification technique ou conception</i>	<i>Modèle logique</i>
<i>Implémentation</i>	<i>Modèle physique</i>
<i>Intégration et Tests</i>	<i>Rapport de cohérence logique</i>
<i>Validation du système</i>	<i>Rapport de conformité</i>
<i>Maintenance et évolution</i>	<i>Documentation et trace</i>

Concepts généraux

- *Cycles de développement*
 - en cascade
 - en V
 - en spirale
 - tridimensionnel

Modèle de cycle en cascade



Modèle de cycle en cascade

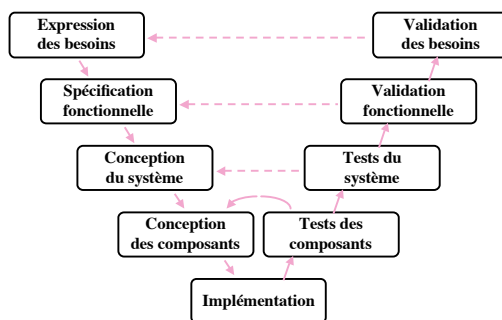
✓ *Contre* :

- ✓ Retours sur les phases précédentes difficiles (rupture entre les phases)
- ✓ Visualisation et validation tardive

✓ *Pour* :

- ✓ Organisation simple et directe

Modèle de cycle en V



Modèle de cycle en V

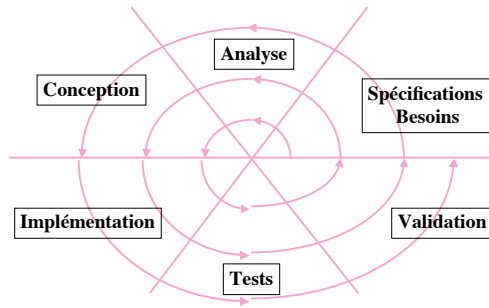
✓ *Pour* :

- ✓ Décomposition du système en sous-systèmes
- ✓ Hiérarchie de tests et retours facilités
- ✓ Vérification ascendante

✓ *Contre* :

- ✓ Validation en fin de cycle (erreurs d'analyse coûteuses)

Modèle de cycle en spirale



Modèle de cycle en spirale

✓ Pour :

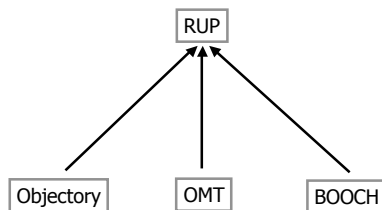
- ✓ Réalisation de plusieurs prototypes (versions) avant la réalisation du système réel (définitif)
- ✓ Validation progressive et précoce
- ✓ Souplesse dans le choix des prototypes

✓ Contre :

- ✓ Mise en œuvre parfois coûteuse
- ✓ Possibilité de divergence, nombre de prototypes difficile à déterminer

➤ *Le RUP est une réponse à ces critiques*

RUP (Rational Unified Process)



RUP (Rational Unified Process)

Mots-clefs :

- ☞ développement itératif
- ☞ développement incrémental
- ☞ pilotage par les cas d'utilisation
- ☞ centré sur l'architecture
- ☞ configurable

RUP (Rational Unified Process)

- 1- Les grandes activités
- 2- La notion d'architecture logicielle
- 3- L'organisation itérative des activités

Bibliographie

- « The RUP, an Introduction »
P. Kruchten, Addison-Wesley 2000
- « The unified Software Development Process »
I. Jacobson, G. Booch, J. Rumbaugh, Addison-Wesley 1999
- « Modélisation Objet avec UML »
P.-A. Muller, N. Gaertner, Eyrolles, 2002

1- Les grandes activités

RUP (Rational Unified Process)

1- Les grandes activités

Le RUP distingue 9 activités,
à chacune correspond :

- 📎 des artefacts
- 👤 des métiers
- ✓ des outils (cf. site de Rational)

RUP (Rational Unified Process)

1- Les grandes activités

activité 1. GESTION DE PROJET

Planification
Allocation des tâches et des responsabilités
Allocation des ressources
Etude de faisabilité et des risques

- 📎 calendrier des tâches
- 👤 chef de projet

RUP (Rational Unified Process)

1- Les grandes activités

activité 2. MODELISATION DE L'ORGANISATION

modélisation

- de la structure
- du fonctionnement
- de l'organisation où le système sera déployé

@ cas d'utilisation de l'organisation
(avec scenarii)

👤 concepteur d'organisation

RUP (Rational Unified Process)

1- Les grandes activités

activité 3. ANALYSE DES BESOINS

détermination des besoins :

- fonctionnels (ce que l'on attend du système)
- non fonctionnels (fiabilité, temps de réponse, environnement distribué, etc.)

@ cas d'utilisation du système à construire
(avec scenarii)

@ documents descriptifs

@ conception de l'interface utilisateur

👤 analyste

RUP (Rational Unified Process)

1- Les grandes activités

activité 4. ANALYSE ET CONCEPTION

évoluer depuis la spécification des besoins jusqu'à une solution informatique

analyse~besoins fonctionnels

conception~intègre aussi les besoins non fonctionnels

@ diagrammes de classes, paquetages,
sous-systèmes

@ diagrammes de collaboration, d'états

@ diagrammes de composants

👤 architecte, concepteur

RUP (Rational Unified Process)

1- Les grandes activités

activité 5. IMPLEMENTATION

Transcription dans un langage de programmation
ou de base de données

Utilisation de composants existants

@ code

👤 implémenteur, développeur

RUP (Rational Unified Process)

1- Les grandes activités

activité 6. TEST

Estimer

si les besoins sont satisfaits
s'il y a des erreurs/défauts à corriger

Renforcer et stabiliser l'architecture

@ modèles de test, scripts

👤 concepteur de test, testeur

RUP (Rational Unified Process)

1- Les grandes activités

activité 6. TEST

Différents niveaux de tests

- *unitaires* (test d'une classe, d'un module isolément)
- *intégration* (plusieurs modules ensembles)
- *validation* (les fonctionnalités du système sont assurées)
- *recette* (souvent contractualisés, avec le client)

RUP (Rational Unified Process)

1- Les grandes activités

activité 6. TEST

Différents types de tests

- *Benchmark* (sur un ensemble de données type)
- *Configuration/installation*
- *Charge*
- *Fiabilité, stress*
- *Performance*

RUP (Rational Unified Process)

1- Les grandes activités

activité 7. DEPLOIEMENT

Distribuer le logiciel dans son environnement opérationnel

Installation, test

Formation des utilisateurs

Migration des données

@ diagrammes de déploiement

👤 formateur, graphiste, rédacteur de documentation, testeur, implémenteur (scripts d'installation)

RUP (Rational Unified Process)

1- Les grandes activités

activité 8. MAINTENANCE ET EVOLUTION

Gérer pendant l'avancement du projet l'évolution :
des besoins des utilisateurs,
du niveau des développeurs,
de la technologie, etc.

- 📄 plan de modification
- 👤 tous les métiers !

RUP (Rational Unified Process)

1- Les grandes activités

activité 9. ENVIRONNEMENT

Activité de support du développement
sélection des outils de travail,
administration système et réseau,
administration BD,
formation de l'équipe de travail, etc.

- 📄 doc outils, doc installation
- 👤 admin. S&R, formateur, admin. BD

2- L'architecture logicielle

RUP (Rational Unified Process)

2- L'architecture logicielle

- Analogie avec l'architecture dans le domaine du bâtiment
- Désigne un ensemble de descriptions de haut niveau (les « plans de construction »)

RUP (Rational Unified Process)

2- L'architecture logicielle

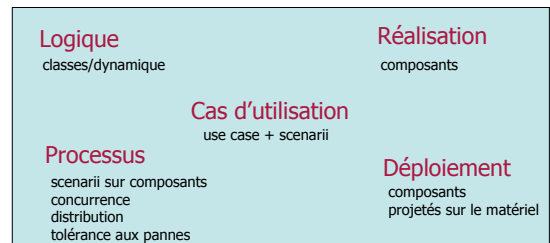
La vue de l'architecte est générale et sert à :

- contrôler l'intégrité du système
- identifier les éléments réutilisables
- baser le partage du travail

RUP (Rational Unified Process)

2- L'architecture logicielle

Plans de construction



Orientation des modèles par les cas d'utilisation

RUP (Rational Unified Process)

2- L'architecture logicielle

Une définition de la notion d'architecture

« *vue limitée du système permettant de comprendre :*

- *ce qu'il fait*
- *comment il fonctionne*
- *comment travailler sur une seule partie*
- *comment l'étendre*
- *comment réutiliser certaines parties »*

Seules les grandes lignes de chaque diagramme font partie de l'architecture

3- L'organisation itérative des activités

RUP (Rational Unified Process)

3- L'organisation itérative des activités

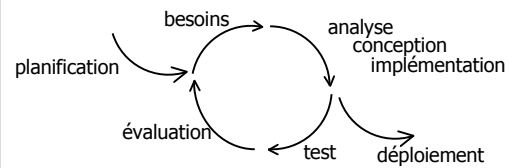
Pour répondre aux problèmes connus du développement en cascade :

- découverte tardive des défauts
- intégration difficile des modifications
- contrôle temps et coûts délicat

RUP (Rational Unified Process)

3- L'organisation itérative des activités

Cycle de base

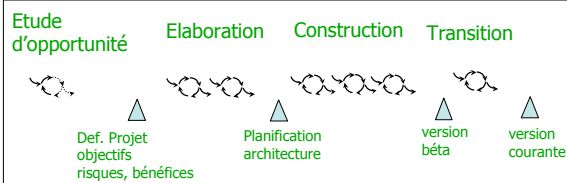


RUP (Rational Unified Process)

3- L'organisation itérative des activités

Contrôle de la convergence : instauration de 4 PHASES

- Chaque phase développe tout ou partie d'un ou plusieurs cycles
- Des points de contrôle entre les phases permettent de vérifier l'avancement



RUP (Rational Unified Process)

3- L'organisation itérative des activités

A chaque itération

- la planification est remise à jour et étendue
- les modèles sont approfondis
- un prototype est développé ou augmenté
- on teste (on re-teste parfois ...) l'existant