



Le Processus unifié de développement logiciel

- Ivar Jacobson
- Grady Booch
- James Rumbaugh

05-336-1

2-005-336-1

Le processus unifié de développement logiciel



- Ivar Jacobson
- Grady Booch
- James Rumbaugh

Traduit de l'anglais par Virginie Zaïm



Eyrolles

Table des matières

Avant propos	1
Qu'est-ce qu'un processus de développement logiciel ?	2
Objectifs du livre	3
À qui s'adresse ce livre ?	3
Approche du livre	4
Historique du Processus unifié	4
L'approche Ericsson	5
SDL : Langage de spécification et de description	6
Objectory	7
L'approche de Rational	8
Processus Rational Objectory : 1995-1997	9
UML : Unified Modeling Language	9
Rational Unified Process	10
Remerciements	10
Pour leur contribution à cet ouvrage	10
Pour leur collaboration fidèle depuis des années	11
Nous voudrions particulièrement remercier	12
Un processus en marche	12

PARTIE I

Le Processus unifié de développement logiciel

CHAPITRE 1

Le Processus unifié : piloté par les cas d'utilisation, centré sur l'architecture, itératif et incrémental

1.1 Le Processus unifié en bref

1.2 Le Processus unifié est piloté par les cas d'utilisation

1.3 Le Processus unifié est centré sur l'architecture

1.4 Le Processus unifié est itératif et incrémental

1.5 La vie du Processus unifié

1.5.1 Le produit

1.5.2 Les phases d'un cycle

1.6 Un processus intégré

CHAPITRE 2

Les quatre « P » du développement logiciel : personnes, projet, produit et processus

2.1 Les personnes jouent un rôle crucial

2.1.1 Les processus de développement ont un impact sur les personnes

2.1.2 Les rôles changent

2.1.3 Des « ressources » aux « travailleurs »

2.2 Tel projet, tel produit

2.3 Un produit ne se résume pas à du code

2.3.1 Qu'est-ce qu'un système logiciel ?

2.3.2 Les artefacts

2.3.3 Un système dispose de plusieurs modèles

2.3.4 Qu'est-ce qu'un modèle ?

2.3.5 Chaque modèle est une vue autonome du système

2.3.6 Exploration d'un modèle

2.3.7 Relations entre modèles

2.4 Le processus dirige les projets	3
2.4.1 Le processus : un cadre générique	3
2.4.2 Les relations entre activités constituent les enchaînement d'activités	3
2.4.3 Spécialisation du processus	3
2.4.4 Les mérites d'un processus	4
2.5 Les outils font partie intégrante du processus	4
2.5.1 Les outils influent sur le processus	4
2.5.2 Le processus conditionne les outils	4
2.5.3 Équilibre entre processus et outils	4
2.5.4 La modélisation visuelle au service d'UML	4
2.5.5 Les outils prennent en charge tout le cycle de vie	4
2.6 Références	4

CHAPITRE 3

Un processus piloté par les cas d'utilisation	4
3.1 Le développement piloté par les cas d'utilisation en bref	4
3.2 Pourquoi les cas d'utilisation ?	5
3.2.1 Pour appréhender les véritables besoins	5
3.2.2 Pour diriger le processus	5
3.2.3 Pour mettre au point l'architecture et plus encore...	5
3.3 « Capture » des cas d'utilisation	5
3.3.1 Le modèle des cas d'utilisation représente les besoins fonctionnels	5
3.3.2 Les acteurs constituent l'environnement du système	5
3.3.3 Les cas d'utilisation spécifient le système	5
3.4 L'analyse, la conception et l'implémentation pour la réalisation des cas d'utilisation	5
3.4.1 Création du modèle d'analyse à partir des cas d'utilisation	5
3.4.2 Chaque classe doit assumer tous ses rôles de collaboration	6
3.4.3 Création du modèle de conception à partir du modèle d'analyse	6
3.4.4 Les sous-systèmes regroupent les classes	6
3.4.5 Création du modèle d'implémentation à partir du modèle de conception	6
3.5 Tests des cas d'utilisation	6
3.6 Résumé	7
3.7 Références	7

CHAPITRE 4

Un processus centré sur l'architecture	71
4.1 L'architecture en bref	72
4.2 Pourquoi il nous faut une architecture	74
4.2.1 Comprendre le système	75
4.2.2 Organiser le développement	75
4.2.3 Favoriser la réutilisation	76
4.2.4 Faire évoluer le système	76
4.3 Cas d'utilisation et architecture	78
4.4 Étapes d'élaboration de l'architecture	81
4.4.1 L'architecture de référence est un système « petit et maigre »	82
4.4.2 Utilisation des patterns d'architecture	84
4.4.3 Description de l'architecture	86
4.4.4 C'est l'architecte qui crée l'architecture	89
4.5 Enfin, une description de l'architecture !	90
4.5.1 Vue architecturale du modèle des cas d'utilisation	91
4.5.2 Vue architecturale du modèle de conception	91
4.5.3 Vue architecturale du modèle de déploiement	94
4.5.4 Vue architecturale du modèle d'implémentation	95
4.6 Trois concepts intéressants	96
4.6.1 Qu'est-ce que l'architecture ?	96
4.6.2. Comment l'obtient-on ?	96
4.6.3 Comment la décrit-on ?	96
4.7 Références	97

CHAPITRE 5

Un processus itératif et incrémental	99
5.1 Le développement itératif et incrémental en bref	100
5.1.1 Procéder par étapes modestes	101
5.1.2 Ce que n'est pas une itération	102
5.2 Pourquoi un développement itératif et incrémental ?	103
5.2.1 Réduire les risques	103
5.2.2 Obtenir une architecture robuste	105
5.2.3 Gérer les exigences de changement	106
5.2.4 Permettre des changements tactiques	106
5.2.5 Parvenir à une intégration continue	107
5.2.6 Accéder très tôt à la connaissance	108

5.3 L'approche itérative est guidée par la réduction des risques	109
5.3.1 Les itérations atténuent les risques techniques	110
5.3.2 Les responsables ont la charge des risques non techniques	112
5.3.3 Traiter les risques	112
5.4 L'itération générique	113
5.4.1 Qu'est-ce qu'une itération ?	113
5.4.2 Planifier les itérations	115
5.4.3 Séquencer les itérations	116
5.5 Une itération se traduit par un incrément	116
5.6 Itérations dans le cycle de vie	117
5.7 Les modèles évoluent à partir des itérations	120
5.8 Les itérations remettent en question l'organisation	121
5.9 Références	122

PARTIE II

Les enchaînements d'activités principaux

CHAPITRE 6

Expression des besoins : de la vision aux besoins	125
6.1 Pourquoi l'expression des besoins est-elle délicate ?	126
6.2 Objectif de l'enchaînement d'activités des besoins	127
6.3 Présentation générale de l'expression des besoins	127
6.4 Rôle des besoins dans le cycle de vie du logiciel	132
6.5 Compréhension du contexte du système à l'aide d'un modèle du domaine	133
6.5.1 Qu'est-ce qu'un modèle du domaine ?	133
6.5.2 Élaboration d'un modèle du domaine	135
6.5.3 Utilisation du modèle du domaine	136
6.6 Compréhension du contexte du système à l'aide d'un modèle du métier	136
6.6.1 Qu'est-ce qu'un modèle du métier ?	137
6.6.2 Élaboration d'un modèle du métier	139
6.6.3 Identification des cas d'utilisation à partir d'un modèle du métier	141

6.7 Exigences supplémentaires	143
6.8 Résumé	144
6.9 Références	144
CHAPITRE 7	
Expression des besoins sous forme de cas d'utilisation ...	147
7.1 Introduction	147
7.2 Artefacts	149
7.2.1 Artefact : modèle des cas d'utilisation	149
7.2.2 Artefact : acteur	150
7.2.3 Cas d'utilisation	151
7.2.4 Artefact : description de l'architecture (vue du modèle des cas d'utilisation)	155
7.2.5 Artefact : glossaire	155
7.2.6 Artefact : prototype d'interface utilisateur	156
7.3 Les travailleurs	156
7.3.1 Travailleur : analyste système	156
7.3.2 Travailleur : spécificateur de cas d'utilisation	157
7.3.3 Travailleur : concepteur d'interface utilisateur	158
7.3.4 Travailleur : architecte	158
7.4 Enchaînement d'activités	159
7.4.1 Activité : identifier les acteurs et les cas d'utilisation	160
7.4.2 Activité : définir un ordre de priorité pour les cas d'utilisation	169
7.4.3 Activité : détailler un cas d'utilisation	170
7.4.4 Activité : prototyper l'interface utilisateur	176
7.4.5 Activité : structurer le modèle des cas d'utilisation	182
7.5 Résumé de l'enchaînement d'activités des besoins	186
7.6 Références	187
CHAPITRE 8	
Analyse	189
8.1 Introduction	189
8.2 L'analyse en bref	192
8.2.1 Pourquoi l'analyse n'est ni la conception, ni l'implémentation...	193
8.2.2 Objectif de l'analyse : résumé	194
8.2.3 Quand employer l'analyse : exemples concrets	194

8.3 Rôle de l'analyse dans le cycle de vie du logiciel	195
8.4 Artefacts	197
8.4.1 Artefact : modèle d'analyse	197
8.4.2 Artefact : classe d'analyse	198
8.4.3 Artefact : réalisation–analyse de cas d'utilisation	202
8.4.4 Artefact : paquetage d'analyse	206
8.4.5 Artefact : description de l'architecture (vue du modèle d'analyse)	209
8.5 Travailleurs	210
8.5.1 Travailleur : architecte	210
8.5.2 Travailleur : ingénieur de cas d'utilisation	210
8.5.3 Travailleur : ingénieur de composants	211
8.6 Enchaînement d'activités	212
8.6.1 Activité : analyse architecturale	213
8.6.2 Activité : analyser un cas d'utilisation	219
8.6.3 Activité : analyser une classe	223
8.6.4 Activité : analyser un paquetage	227
8.7 Résumé de l'analyse	228
8.8 Références	230

CHAPITRE 9

Conception	231
9.1 Introduction	231
9.2 Rôle de la conception dans le cycle de vie du logiciel	232
9.3 Artefacts	233
9.3.1 Artefact : modèle de conception	233
9.3.2 Artefact : classe de conception	234
9.3.3 Artefact : réalisation–conception de cas d'utilisation	237
9.3.4 Artefact : sous-système de conception	240
9.3.5 Artefact : interface	242
9.3.6 Artefact : description de l'architecture (vue du modèle de conception)	243
9.3.7 Artefact : modèle de déploiement	244
9.3.8 Artefact : description de l'architecture (vue du modèle de déploiement)	245

9.4 Travailleurs	245
9.4.1 Travailleur : architecte	245
9.4.2 Travailleur : ingénieur de cas d'utilisation	246
9.4.3 Travailleur : ingénieur de composants	247
9.5 Enchaînement d'activités	248
9.5.1 Activité : conception architecturale	248
9.5.2 Activité : concevoir un cas d'utilisation	264
9.5.3 Activité : concevoir une classe	271
9.5.4 Activité : concevoir un sous-système	278
9.6 Résumé de la conception	280
9.7 Références	281
 CHAPITRE 10	
Implémentation	283
10.1 Introduction	283
10.2 Rôle de l'implémentation dans le cycle de vie du logiciel	284
10.3 Artefacts	285
10.3.1 Artefact : modèle d'implémentation	285
10.3.2 Artefact : composant	285
10.3.3 Artefact : sous-système d'implémentation	288
10.3.4 Artefact : interface	289
10.3.5 Artefact : description de l'architecture (vue du modèle d'implémentation)	291
10.3.6 Artefact : plan de construction des intégrations	291
10.4 Travailleurs	293
10.4.1 Travailleur : architecte	293
10.4.2 Travailleur : ingénieur de composants	294
10.4.3 Travailleur : intégrateur système	294
10.5 Enchaînement d'activités	295
10.5.1 Activité : implémenter l'architecture	296
10.5.2 Activité : intégrer le système	298
10.5.3 Activité : implémenter un sous-système	301
10.5.4 Activité : implémenter une classe	304
10.5.5 Activité : effectuer les tests unitaires	305
10.6 Résumé de l'implémentation	309
10.7 Références	309

CHAPITRE 11

Test	311
11.1 Introduction	311
11.2 Rôle des tests dans le cycle de vie du logiciel	312
11.3 Artefacts	313
11.3.1 Artefact : modèle des tests	313
11.3.2 Artefact : cas de test	314
11.3.3 Artefact : procédure de test	316
11.3.4 Artefact : composant de test	317
11.3.5 Artefact : plan des tests	318
11.3.6 Artefact : anomalie	318
11.3.7 Artefact : évaluation des tests	318
11.4 Travailleurs	318
11.4.1 Travailleur : concepteur de tests	318
11.4.2 Travailleur : ingénieur de composants	319
11.4.3 Travailleur : testeur d'intégration	319
11.4.4 Travailleur : testeur système	319
11.5 Enchaînement d'activités	320
11.5.1 Activité : planifier les tests	321
11.5.2 Activité : concevoir les tests	322
11.5.3 Activité : implémenter les tests	325
11.5.4 Activité : effectuer les tests d'intégration	326
11.5.5 Activité : effectuer les tests système	327
11.5.6 Activité : évaluer les tests	328
11.6 Résumé des tests	329
11.7 Références	329

PARTIE III

Un développement itératif et incrémental 331

CHAPITRE 12

Enchaînement d'activités de l'itération générique	333
12.1 Le besoin d'équilibre	334
12.2 Les phases constituent la première division du travail	335
12.2.1 La phase de création établit la faisabilité	335

12.2.2 La phase d'élaboration s'intéresse à la « capacité de réalisation »	336
12.2.3 La phase de construction « construit » le système	337
12.2.4 La phase de transition aborde l'environnement utilisateur	337
12.3 L'itération générique revisitée	338
12.3.1 Les enchaînements d'activités principaux se répètent à chaque itération	338
12.3.2 Les travailleurs prennent part aux enchaînements d'activités	339
12.4 La planification précède l'action	341
12.4.1 Planifier les quatre phases	341
12.4.2 Planifier les itérations	342
12.4.3 Penser à long terme	343
12.4.4 Planifier les critères d'évaluation	343
12.5 Les risques affectent la planification du projet	345
12.5.1 Gérer une liste des risques	345
12.5.2 Les risques affectent le plan des itérations	346
12.5.3 Planifier les actions à entreprendre face aux risques	346
12.6 Définition d'un ordre de priorité pour les cas d'utilisation	347
12.6.1 Risques spécifiques à un produit particulier	348
12.6.2 Risques de créer une architecture inadaptée	348
12.6.3 Risque de ne pas bien comprendre les besoins	349
12.7 Ressources nécessaires	350
12.7.1 Les projets diffèrent énormément	351
12.7.2 Voici à quoi ressemble généralement un projet... ..	352
12.7.3 À projets complexes, besoins supérieurs	352
12.7.4 Une nouvelle ligne de produits exige de l'expérience	353
12.7.5 Il faut payer le prix des ressources utilisées	354
12.8 Évaluer les itérations et les phases	355
12.8.1 Les critères ne sont pas remplis	356
12.8.2 Les critères eux-mêmes	356
12.8.3 L'itération suivante	356
12.8.4 Évolution de l'ensemble des modèles	357

CHAPITRE 13

La phase de création lance le projet	359
13.1 La phase de création en bref	359
13.2 Premiers stades de la phase de création	360
13.2.1 Avant le début de la phase de création	360
13.2.2 Planifier la phase de création	361
13.2.3 Élargir la vision du système	362
13.2.4 Définir les critères d'évaluation	362
13.3 Enchaînement d'activités de l'itération typique de création	364
13.3.1 Introduction aux cinq enchaînements d'activités principaux ...	365
13.3.2 Adapter le projet à l'environnement de développement	366
13.3.3 Identifier les risques critiques	367
13.4 Exécuter les enchaînements d'activités principaux : des besoins aux tests	367
13.4.1 Formuler les besoins	368
13.4.2 Analyse	371
13.4.3 Conception	372
13.4.4 Implémentation	373
13.4.5 Tests	373
13.5 Élaborer l'étude de rentabilité initiale	373
13.5.1 Tracer les grandes lignes de l'offre	374
13.5.2 Estimer le retour sur investissement	375
13.6 Évaluer les itérations dans la phase de création	375
13.7 Planifier la phase d'élaboration	376
13.8 Éléments à livrer à l'issue de la phase de création	378

CHAPITRE 14

La phase d'élaboration fabrique l'architecture de référence	379
14.1 La phase d'élaboration en bref	379
14.2 Premiers stades de la phase d'élaboration	380
14.2.1 Planifier la phase d'élaboration	380
14.2.2 Constituer l'équipe	381
14.2.3 Modifier l'environnement de développement	381
14.2.4 Définir les critères d'évaluation	381

14.3 Enchaînements d'activités de l'itération typique d'élaboration	382
14.3.1 Formuler et affiner la plupart des besoins	383
14.3.2 Développer l'architecture de référence	384
14.3.3 Procéder à des itérations tant que l'équipe est réduite	384
14.4 Exécuter les enchaînements d'activités principaux :	
des besoins aux tests	384
14.4.1 Formuler les besoins	386
14.4.2 Analyse	388
14.4.3 Conception	392
14.4.4 Implémentation	395
14.4.5 Tests	397
14.5 Élaborer l'étude de rentabilité	398
14.5.1 Préparer l'offre commerciale	398
14.5.2 Actualiser le retour sur investissement	399
14.6 Évaluer les itérations de la phase d'élaboration	399
14.7 Planifier la phase de construction	400
14.8 Principaux éléments à livrer	401

CHAPITRE 15

La construction aboutit à la capacité opérationnelle initiale	403
15.1 La phase de construction en bref	403
15.2 Premiers stades de la phase de construction	404
15.2.1 Constituer l'équipe de la phase	405
15.2.2 Définir les critères d'évaluation	405
15.3 Enchaînements d'activités de l'itération typique de construction	406
15.4 Exécuter les enchaînements d'activités principaux :	
des besoins aux tests	408
15.4.1 Besoins	409
15.4.2 Analyse	410
15.4.3 Conception	411
15.4.4 Implémentation	413
15.4.5 Tests	414
15.5 Maîtriser l'étude de rentabilité	416
15.6 Évaluer les itérations et la phase de construction	416
15.7 Planifier la phase de transition	417
15.8 Principaux éléments à livrer	417

CHAPITRE 16

La phase de transition finalise le produit	419
16.1 La phase de transition en bref	420
16.2 Premiers stades de la phase de transition	421
16.2.1 Planifier la phase de transition	421
16.2.2 Constituer l'équipe de la phase de transition	423
16.2.3 Définir les critères d'évaluation	423
16.3 Les enchaînements d'activités principaux jouent un rôle mineur dans cette phase	424
16.4 Activités de la phase de transition	425
16.4.1 Livrer la version bêta	425
16.4.2 Installer la version bêta	426
16.4.3 Répondre aux résultats des tests	426
16.4.4 Adapter le produit aux divers environnements utilisateur	427
16.4.5 Finaliser les artefacts	429
16.4.6 Quand le projet se termine-t-il ?	429
16.5 Achever l'étude de rentabilité	429
16.5.1 Maîtriser la progression	430
16.5.2 Réviser le plan stratégique	430
16.6 Évaluer la phase de transition	430
16.6.1 Évaluer les itérations et la phase	431
16.6.2 Post mortem du projet	431
16.7 Planifier la version ou la génération suivante	432
16.8 Principaux éléments à livrer	432

CHAPITRE 17

Application optimale du Processus unifié	433
17.1 Le Processus unifié aide à gérer la complexité	433
17.1.1 Les objectifs du cycle de vie	434
17.1.2 L'architecture du cycle de vie	434
17.1.3 Capacité opérationnelle initiale	435
17.1.4 Livraison du produit	435
17.2 Les principaux thèmes	435
17.3 Les responsables dirigent la transition vers le Processus unifié	436
17.3.1 Les raisons d'agir	437
17.3.2 La directive de réingénierie achève de convaincre	438
17.3.3 Mettre en œuvre la transition	439

17.4 Spécialiser le Processus unifié	441
17.4.1 Adapter le Processus	441
17.4.2 Étoffer l'infrastructure du Processus	442
17.5 Élargir le cercle de ses interlocuteurs	442
17.6 Profiter des avantages qu'offre le Processus unifié	443
17.7 Références	444
 ANNEXE A	
Présentation générale du langage UML	445
A.1 Introduction	445
A.1.1 Vocabulaire	446
A.1.2 Mécanismes d'extensibilité	446
A.2 Notation graphique	447
A.2.1 Éléments structurels	447
A.2.2 Éléments comportementaux	448
A.2.3 Éléments de regroupement	449
A.2.4 Éléments d'annotation	449
A.2.5 Relations de dépendance	450
A.2.6 Relations d'association	450
A.2.7 Relations de généralisation	450
A.2.8 Mécanismes d'extensibilité	451
A.3 Glossaire	451
A.4 Références	459
 ANNEXE B	
Extensions du langage UML spécifiques au Processus unifié	461
B.1 Introduction	461
B.2 Stéréotypes	461
B.3 Valeurs marquées	464
B.4 Notation graphique	464
B.5 Références	465
 ANNEXE C	
Glossaire général	467
C.1 Introduction	467
C.2 Termes	467