

**REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE  
UNIVERSITE DES SCIENCES ET DE LA TECHNOLOGIE  
HOUARI BOUMEDIENE**

**FACULTE DE GENIE ELECTRIQUE**

**DEPARTEMENT D'INFORMATIQUE**

**THESE  
Pour l'obtention du grade de  
MAGISTER EN INFORMATIQUE**

# **Intégration de modèles de Procédés Logiciels**

***Réalisé par***

Melle Nabila LARDJANE

***Dirigé par :***

Mr M. AHMED-NACER

***Jury :***

Mme Z. ALIMAZIGHI (présidente)

Mme A. AISSANI (examineur)

Mme BOUKALA (examineur)

# SOMMAIRE

## INTRODUCTION GENERALE

### CHAPITRE I :

#### PARTIE 1 : Les Ateliers de Génie Logiciel et les procédés logiciels

1. ATELIERS DE GENIE LOGICIEL	6
2. LES OUTILS CASES	6
3. INTEGRATION DES OUTILS CASES DANS UN AGL	7
3.1. Intégration par la présentation	8
3.2. Intégration au niveau des données	8
3.3. Intégration au niveau du contrôle	8
3.4 Intégration au niveau des procédés	9
4. LES AGLS DIRIGES PAR LES MODELES DE PROCEDES LOGICIELS	9
5. LES MODELES DE CYCLE DE VIE	10
5.1 Modèle en cascade	11
5.2 Modèle en V	11
5.3 Modèle en spirale	12
5.4 Modèle par incréments	12
6. MODELE DE PROCEDE LOGICIEL ET LANGAGE DE DESCRIPTION	14
7. POINTS DE VUES DES FORMALISMES DE MODELISATION DES PROCEDES	14
7.1 L'approche procédurale	15
7.2 L'approche déclarative	16
7.3 L'approche basée sur les réseaux de Petri	16
7.4 Approche basée sur les Bases de données Actives	17

7.5 L'approche multi-paradigmes	18
8. ARCHITECTURE D'UN AGL CENTRE PROCEDE LOGICIEL	19
9. COMPLEXITE DES ENVIRONNEMENTS CENTRES PROCEDES LOGICIELS	21
10. CONCLUSION	21

## **PARTIE 2 : Les environnements de développement et l'ingénierie coopérative**

1. CARACTERISTIQUE D'UN ENVIRONNEMENT DE DEVELOPPEMENT COOPERATIF	22
1.1 Multiples emplacements	23
1.2 Multiples copies	23
1.3 Multiples représentations	23
1.4 Contrôle de la collaboration	23
2. LES AGLS CENTRES PROCEDES LOGICIELS ET LE TRAVAIL COOPERATIF	24
2.1 La coordination	24
2.2 La collaboration	25
2.3 La communication	26
3. RELATION ENTRE UN ESPACE DE TRAVAIL ET ESPACE DE STOCKAGE	27
4. QUELQUES SUPPORTS A L'IC	28
4.1 Corba (Common Object Request Broker Architecture)	28
4.2 WebDav (Web Distributed Authoring and Versioning)	29
4.3 Adèle - Système de gestion de configuration	29
4.4 Orion – Système de gestion de BD	30
5. DISCUSSION	31
6. LES ENVIRONNEMENTS DE DEVELOPPEMENT DE LOGICIELS MULTI-VUES	31
7. CONCLUSION	33

## **CHAPITRE II : Architecture distribuée à base de composants Intéropérables**

1. OBJECTIF DE LA DECOMPOSITION D'UN PROCEDE	35
2. DEFINITION D'UN COMPOSANT	36
3. ARCHITECTURE	36
4. AUTONOMIE ET HETEROGENEITE DES COMPOSANTS	37
4.1 Autonomie des composants	37
4.2 Hétérogénéité des composants	37
5. TECHNIQUES D'INTEROPERABILITE :	38
5.1 Interopérabilité sémantique	38
5.2 Interopérabilité syntaxique	38
5.3 Interopérabilité au niveau exécution	40
6. FEDERATION DES COMPOSANTS LOGICIELS	40
6.1 Le Modèle commun	41
6.2 Le méta-modèle commun et le langage commun	41
6.3 La fondation	42
6.4 Les composants	42
7. QUELQUES ENVIRONNEMENTS BASES SUR DES COMPOSANTS INTEROPERABLES	42
7.1 OPC .Open Process Component Framework	42
7.2 OZ	43
7.3 Endeavors	43
7.4 APEL	43
8. CONCLUSION	44

## CHAPITRE III : Intégration de schémas de Bases de données réparties et hétérogènes

1. LES BASES DE DONNEES REPARTIES (BDR)	45
1.1 Architecture du schéma d'une BDR	46
1.2 Caractéristiques des BDs Distribuées et hétérogènes	47
2. INTEGRATION DE SCHEMAS : CONCEPTS ET METHODOLOGIE	48
2.1 Concepts d'intégration des schémas	50
2.2 Causes de diversité de schémas	50
2.3 Les concepts communs	53
2.4 Propriétés interschémas	53
3. METHODOLOGIE D'INTEGRATION DES SCHEMAS	54
3.1 Sources de connaissances	55
3.2 Etapes d'intégration	56
3.3 Caractéristique d'un schéma global	56
3.4 Les conflits lors de l'intégration	57
4. QUELQUES METHODES D'INTEGRATION	59
4.1 Méthode décrite par C. Batini, M. Lenzerini & S.B Navathe [Batini & al 86]	59
4.2 Méthode décrite par Johan Eder & Heinz Frank [Eder & Frank 94]	59
4.3 Méthode décrite par W.Gothard, A. Newfeld & P.C.Lockemann [Gothard & al 92] <b>Erreur ! Signet non défini.</b> 60	60
4.4 Méthode décrite par P Martin & W.Powley [Martin & Powley 95]	60
4.5 Méthode de K.Jia Ling & L.P. Chen [ Jia-ling Koh & al 93]	61
5. DISCUSSION	61
6. CONCLUSION	62
<b>CHAPITRE IV : Intégration des fragments de procédés logiciels</b>	
1. INTRODUCTION	63

2. NOTRE APPROCHE D'INTEGRATION DANS LA MODELISATION DISTRIBUEE DE PROCEDES LOGICIELS	64
3. MODELISATION DES FRAGMENTS	65
3.1 Modèle statique	66
3.1 Modèle dynamique	67
4. METHODOLOGIE D'INTEGRATION	70
5. INTEGRATION DU MODELE STATIQUE	72
5.1. Correspondances et conflits entre concepts du modèle statique	72
Correspondances entre Classes	72
Correspondances entre attributs	73
Correspondances entre méthodes	74
5.2 Comparaison entre modèles statiques et résolution des conflits	74
5.3 Conformité des fragments	76
5.4 Fusion et restructuration du modèle global	78
6. INTEGRATION DES DIAGRAMMES ETATS/TRANSITIONS	80
6.1mDéfinitions	80
6.1.1mDéfinitions et correspondances entre etats :	80
6.1.2mDéfinitions et correspondances entre transitions :	80
6.2 Comparaison des diagrammes	81
6.3 Conformité entre diagrammes	81
6.4 Fusion et restructuration du schéma global	82
7. INTEGRATION DES DIAGRAMMES D'ACTIVITES	84
7.1mDéfinitions	84
7.2mComparaison entre les diagrammes d'activité	84
7.3mConformité entre les diagrammes d'activités	88
8. INTEGRATION DES DIAGRAMMES DE SEQUENCES	88
Comparaison et conformité entre diagrammes de séquences	88
9. CONCLUSION	89
<b>CONCLUSION GENERALE</b>	

## Résumé

Cette thèse présente une approche d'intégration de modèles de procédés logiciels dans un contexte distribué. Elle consiste à voir un procédé logiciel comme étant un ensemble de sous procédés selon des points de vues dirigés par les besoins. Ces sous-procédés sont modélisés selon la notation UML (*Unified Modeling Language*) à travers quatre perspectives : schéma statique, schéma flot de données, schéma temporel et flot de contrôle et enfin le schéma comportemental.

La méthodologie d'intégration présentée permet d'unifier les différents fragments de modèles de processus tant au niveau statique qu'au niveau dynamique (comportemental). Pour cela, et afin de distinguer clairement la sémantique des différents conflits possibles, des définitions formelles des propriétés inter-fragments pour les différents diagrammes ont été établies et des exemples de solutions à ces conflits ont été proposés.

Ainsi, la méthode donne alors la possibilité d'améliorer et de concevoir de nouveaux modèles de procédés logiciels par simple fusion de fragments de processus réutilisables.

**Mots clés :** fragment de modèle de procédé logiciel, intégration de fragments, modèle statique, modèle dynamique, correspondances entre concepts, conflits d'intégration et méthodologie d'intégration.

## Abstract

This thesis presents an integration approach of software processes models in a distributed context. It defines a software process as a set of sub-processes according to different points of view related to different needs.

These sub-processes are modeled according to the UML notation (Unified Modeling language) and through four perspectives: static model, data-flow model, temporal and control-flow model and finally the behavioral model.

The integration methodology presented allows unifying the various fragments of process as at the static level as at the dynamic level (behavioral). For that purpose, and to distinguish clearly semantics of the various possible conflicts, Formal definitions of the inter-fragments properties for the various diagrams were established and examples of solutions of these conflicts were proposed.

So, the method gives the possibility of improving and of conceiving new models of processes software by simple fusion of fragments of reusable processes.

**Keywords:** process fragment, integration of fragments, static model, dynamic model, concepts correspondences, integration conflicts and integration methodology.

resume