

Réflexivité, Aspects et Composants pour l'Ingénierie des Intergiciels et des Applications Réparties

Lionel Seinturier

Habilitation à diriger des recherches
Université Pierre & Marie Curie
13 décembre 2005

1. Introduction

Contexte scientifique

Complexification des intergiciels

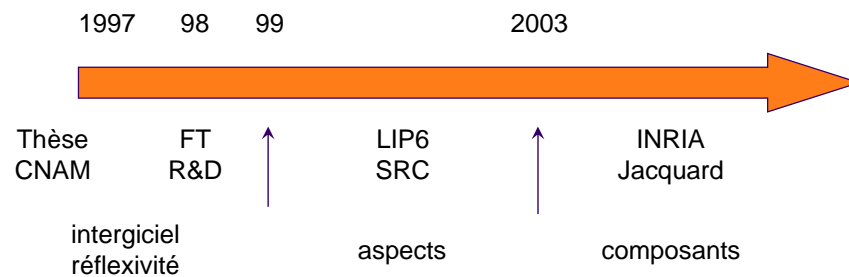
- message, RPC, objet, composant, ...
- ↗ du nombre de services offerts par les plates-formes
- comment spécifier/contractualiser/assembler services et applications
- comment gérer/administrer/déployer ces services
- clés pour de nombreuses problématiques
 - réutilisabilité, évolutivité, maintenabilité, adaptabilité, dynamicité

Lionel Seinturier

S2 - 13/12/2005

1. Introduction

Parcours



Lionel Seinturier

S3 - 13/12/2005

Plan

1. Introduction
2. Réflexivité
3. Aspects
4. Composants
5. Conclusion et perspectives

Lionel Seinturier

S4 - 13/12/2005

2. Réflexivité

Intergiciel & réflexivité

Contexte : intergiciel CORBA

Problèmes

- couplage fort entre services et applications
- « mélange » des préoccupations

Une solution : réflexivité

- au niveau système :
OpenCorba, OpenORB, Apertos, ...
- au niveau service :
migration, réplication, persistance,
tolérance aux pannes, coordination, ...

niveau méta



niveau de base

2. Réflexivité

Observation d'exécutions réparties

Contributions

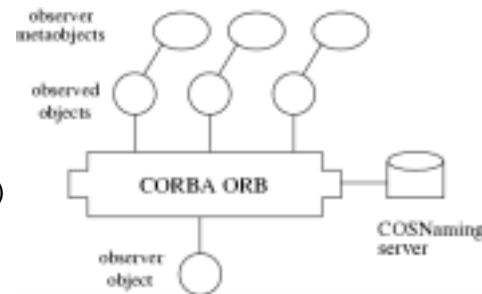
- Réflexivité pour un service intergiciel
 - service observation d'exécutions réparties CORBA
 - réflexivité (OpenJava) pour intégrer le service à l'application
 - intergiciel CORBA JacORB
- Relation d'ordre pour des événements distribués
 - extension relation d'ordre causale *Happened-before* [Lamport 78]
 - observer la causalité due à la sémantique de l'application

2. Réflexivité

Observation d'exécutions réparties

Événements réifiés (observés)

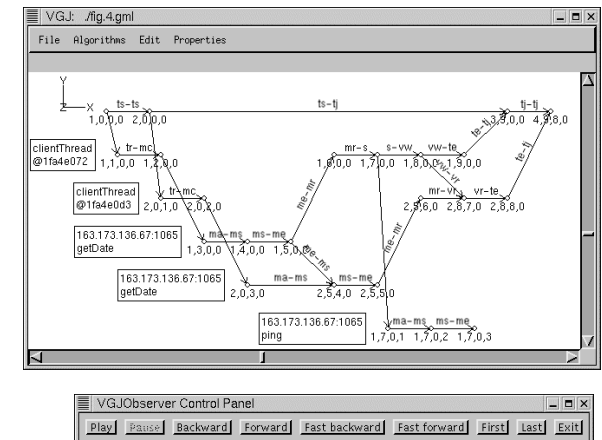
1. communications (asynchrone, synchrone, arrivée/début/fin)
2. activités (*thread*, création/début/fin/*join*)
3. synchronisations
4. lectures/écritures



2. Réflexivité

Reconstitution en ligne

- estampillage
- algorithme de calcul du graphe des dépendances
- outil de visualisation



2. Réflexivité

Conclusion et perspectives sur le service

- séparation effective entre applications et services
- limiter le volume des traces
- étendre l'observation : par ex. détection de prédicats, contrôle de rejeu

Problèmes ouverts

- configuration
- caractère multi-langages (Java, IDL) non pris en compte

Plan

1. Introduction
2. Réflexivité
3. Aspects
 - 3.1 Java Aspect Components (JAC)
 - 3.2 Aspects pour composants intergiciel
4. Composants
5. Conclusion et perspectives

3. Aspects

Aspect-Oriented Programming (AOP)

[Kiczales 96-97]

Constat : code entrelacé (*tangled*) et dispersé (*scattered*)

Analyse : influence de la décomposition dominante

Solution : un nouveau type d'entité logicielle (aspect)

Réponse AOP : une (+sieurs) nouvelle dimension de modularité

3. Aspects

Intérêt pour l'AOP

- Génie logiciel
 - structuration du code

Intergiciel

- architecture des plates-formes
- liaison dynamique pour reconfiguration [Kenens 98]
- gestion de la complexité
 - serveur servlet : Tomcat [Kiczales]
 - serveur web : JaWS [Kulesza 00]
 - ORB CORBA : ORBacus/JacORB/OpenORB [Zhang 03]
- familles de produits
 - serveur J2EE : Websphere [Colyer 04]

3. Aspects

2 contributions

- plate-forme JAC (thèse R. Pawlak)
- aspects pour plates-formes intergiciel à composants (thèse F. Legend)

3.1 JAC

JAC (Java Aspect Components)

- plate-forme pour le développement d'applications orientées aspect
- project *open-source* ObjectWeb
- une aventure collective (1999-2003)
 - la thèse de R. Pawlak [1999-2002]
 - équipe : L. Martelli, L. Seinturier, L. Duchien, G. Florin

Principaux domaines ciblés

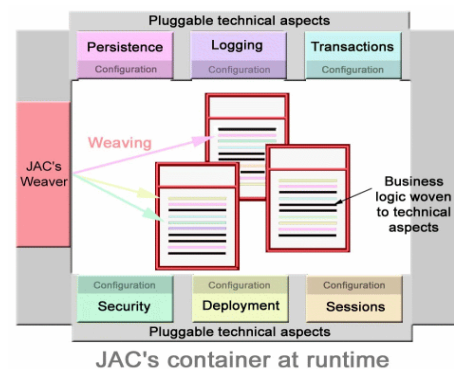
- applications client/serveur 3-tiers
- intergiciels adaptatifs
- algorithmique répartie

3.1 JAC

JAC (Java Aspect Components)

Conteneur ouvert

- services sous forme d'aspects
- API chargement
- API tissage



3.1 JAC

JAC (Java Aspect Components)

Modèle de programmation

- 100% Java
- *framework* de prise en charge des objets et des aspects
- dynamique tissage/détissage/retissage
- objets : préparés pour le tissage par instrumentation *bytecode*
- aspects : liés aux objets sélectionnés par les coupes
- granularité du tissage : objet

3.1 JAC

JAC (Java Aspect Components)

Configuration des aspects

- externaliser informations exogènes aux aspects
 - coupe
 - paramètres indépendants
 - paramètres dépendants du contexte
- DSL de configuration pour chaque aspect

3.1 JAC

JAC (Java Aspect Components)

Support pour la programmation distribuée

- instantiation distante (RMI ou CORBA) aspects, objets, souches
- aspects de réplication, cohérence, gestion de groupe, diffusion
- coupe répartie

3.1 JAC

JAC (Java Aspect Components)

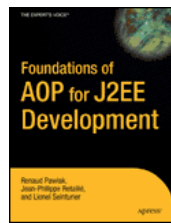
Résultats

- un *framework* de programmation AOP dynamique
- un serveur d'applications orienté aspect
- une notion de coupe répartie
- configuration d'aspects

Publications majeures : 1 livre (fr & en), 1 chap., Reflection'01, SPE 2004

Encadrements : thèse R. Pawlak, 3 DEA (G. Haïk, T. Garcia, L. Teboul)

Projet : IST NoE AOSD



Plan

1. Introduction
2. Réflexivité
3. Aspects
 - 3.1 Java Aspect Components (JAC)
 - 3.2 Aspects pour composants intergiciel
4. Composants
5. Conclusion et perspectives

3.2 Aspects pour composants intergiciel

Aspects pour intergiciels à composants

- thèse F. Legond (1 partie) [2001-2005] – co-encadrement avec G. Florin

Contexte

- applications composants EJB ou CCM

Constat

- similitudes importantes dans la programmation EJB ou CCM

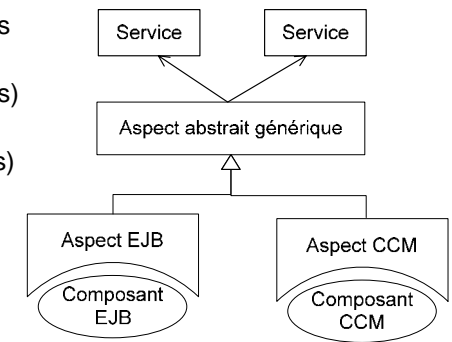
Objectifs

- abstraire ces similitudes dans un modèle commun
- capturer ce modèle à l'aide d'aspects
- pouvoir y greffer des services de niveau applicatif

3.2 Aspects pour composants intergiciel

Schéma de principe

- définition d'événements communs aux 2 modèles
- capture à l'aide d'aspects (coupes) génériques
- utilisation de ces aspects (coupes) pour introduire de nouveaux services au sein des applications EJB ou CCM



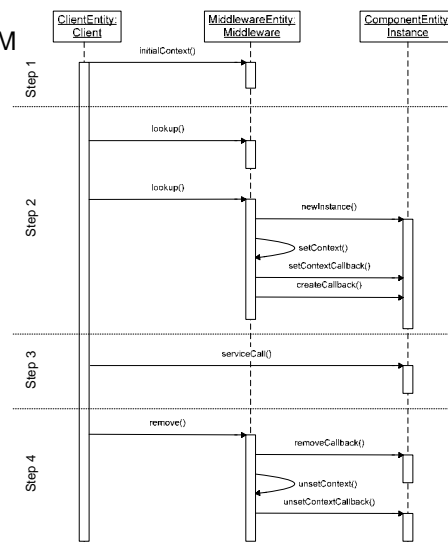
3.2 Aspects pour composants intergiciel

Similitudes des modèles EJB et CCM

1. Découverte de services
2. Obtention d'instances de composants
3. Invocation des composants
4. Relaxation des instances de composants

Services développés

- vérification de comportements (automates états/transitions)
- répartition de charge



3. Aspects

AOP – Perspectives

- interfaces d'aspect
- vers une AOP plus symétrique
- composition d'aspects
- aspects dans les phases amont
 - thèse en cours : D. Diaz
- coupes comportementales
- coupes architecturales
- élargir la couverture fonctionnelle des aspects
 - aspects sécurité dans serveur Web (projet LAFMI 2006)

Plan

1. Introduction
2. Réflexivité
3. Aspects
4. Composants
 - 4.1 Convergence aspect-composant
 - 4.2 Ingénierie du contrôle des composants
5. Conclusion et perspectives

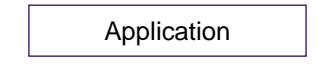
4. Composants

Component-based programming [McIlroy 68]

Composant et intergiciel

Au niveau application

- CCM, EJB, .Net/COM+, Comet
- SOFA, Artic Bean, Accord/UML
- ...



Au niveau intergiciel

- OSGi, Fractal, Kilim, OpenCOM, JMX, K-Component, ...



Aux 2 niveaux

- JOnAS à la carte [Abdellatif 05], JOnAS on OSGi [Desertot 05]

4. Composants

Complémentarité aspect-composant

- unité de structuration
 - fonctionnalités horizontales (transversales)
 - fonctionnalités verticales (métier)

Projet : unification des 2 concepts

Expériences préliminaires

- étude de cas comparative Fractal/Kilim/JAC (DEA F. Loiret 2003)
- extension de Fractal pour le support des aspects
- ingénierie du contrôle des composants à l'aide d'aspects

Plan

1. Introduction
2. Réflexivité
3. Aspects
4. Composants
 - 4.1 Convergence aspect-composant
 - 4.2 Ingénierie du contrôle des composants
5. Conclusion et perspectives

4.1 Convergence aspect-composant

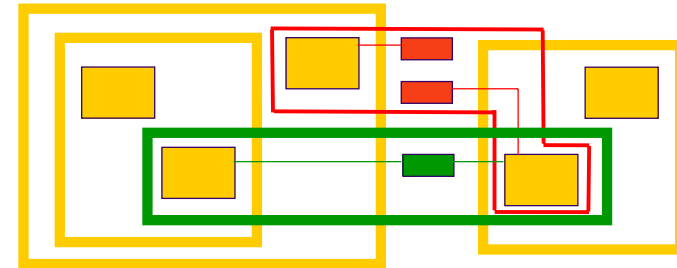
FAC (Fractal Aspect Component)

- thèse N. Pessemier [2004-...], contrat FT R&D
- implémentation dans Fractal [Bruneton 04]
- modèle « à la JAC » (JBoss AOP, AspectWerkz, ...)
- tissage d'aspect
 - = assemblage de composants
 - = liaison entre composants de base et composant d'aspect

4.1 Convergence aspect-composant

Mise en œuvre dans Fractal

- notion de domaine d'aspect (composite + partage)



- une aspectisation des architectures à base de composants

Plan

1. Introduction
2. Réflexivité
3. Aspects
4. Composants
 - 4.1 Convergence aspect-composant
 - 4.2 Ingénierie du contrôle des composants
5. Conclusion et perspectives

4.2 Ingénierie du contrôle des composants

2 dimensions \forall *frameworks* de composants

- fonctionnel
- contrôle (services)

Problèmes

- adapter le contrôle à des contextes variés
- ingénierie du contrôle

Objectifs

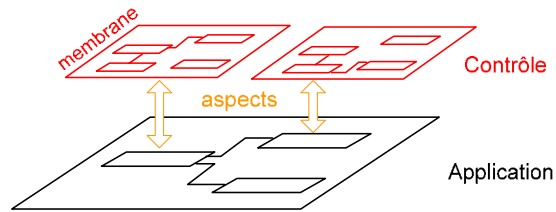
- avoir des composants sémantiquement riches (*self-**)
- pouvoir décliner une application dans différents contextes d'exécution

4.2 Ingénierie du contrôle des composants

Premiers éléments de solution

AOKell

- un modèle de composant réflexif
- implémentation des spécifications Fractal



Plan

1. Introduction
2. Réflexivité
3. Aspects
4. Composants
 - 4.1 Convergence aspect-composant
 - 4.2 Ingénierie du contrôle des composants
5. Conclusion et perspectives

5. Conclusion

Résultats

- réflexivité pour le développement de services
 - relation d'ordre « sémantique », lien méta
- aspects pour le développement de services
 - AOP dynamique
 - DSL de configuration
- rapprochement aspect-composant
 - unification du modèle de programmation
 - ingénierie du contrôle

2 thèses soutenues
R. Pawlak, F. Legend

1 thèse en cours
N. Pessemier

5. Perspectives

- aspects dans les phases amont
- composants pour l'embarqué
- ingénierie + formalisation de propriétés d'adaptabilité pour composants
 - ANR ARA REVE (2006-2008)

1 thèse en cours
D. Diaz

1 thèse en cours
F. Loiret

Projet de recherche

- composants sémantiquement riches (*self-**)
- co-évolution
 - applications (composants)
 - support d'exécution (services système, *middleware*, VM, ...)

Publications principales

- 1 livre : 2 éditions (1 en anglais chez APress, 1 en français chez Eyrolles)
- 1 chapitre de livre Addison-Wesley
- 5 revues internationales : KAIS, IJCIM, IJPDSN, IJTIDM, SPE
- 5 revues nationales : Calculateurs Parallèles, 3 x L'Objet, TSI
- 20 publications dans des congrès internationaux avec comité de lecture : AADEBUG, AOSD, DIMAS, DNIS, 2 x ERSADS, ETFA, ICMIT, IWCC, 2 x FMOODS, MIW, PDCS, Reflection, TOOLS Europe, 2 x TOOLS USA, WAOPSC, WAOSD, WFCS