

Des signatures numériques pour améliorer la recherche structurelle de patrons

Yann-Gaël Guéhéneuc
Houari Sahraoui

{guehene,sahraouh}@iro.umontreal.ca

2005/10/03



Group of Open and Distributed Systems, Experimental Software Engineering
Department of Informatics and Operations Research
University of Montreal



Motivations

- Restructuration informée des programmes
 - *Refactorings*
- Motifs de conception (MC)
 - Implantations souvent adaptées
 - Opportunités de restructurations



Motivations

- MC définissent des rôles
- Occurrence d'un MC
 - Combinaison de classes qui correspondent aux rôles
 - Micro-architecture du programme
- Identifier des MC
 - Trouver toutes les combinaisons de classes qui satisfont les rôles
 - Tâche complexe, coûteuse



Recherche structurelle

- Programmation par contraintes avec explications
- Coûteuse
- ...

Exemple

- Identifier *Abstract Factory* dans un programme de taille moyenne
 - JREFACTORY (575 classes)
 - 5 rôles (*AbstractFactory*, *ConcreteFactory*, *AbstractProduct*, *Product* et *Client*)
 - $\sim 575^5 = 6.28549E+13$ combinaisons possibles

Idée

- Réduire l'espace de recherche par des heuristiques
- Une heuristiques efficaces
 - Caractérisation quantitative des rôles dans les MC
- ...



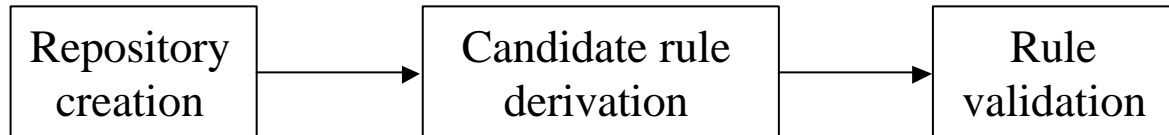
Idée

- Approche déjà expérimentée
 - Par exemple, [Antoniol *et al.*, 98]
- Limitations des approches existantes
 - Caractérisations quantitative manuelle
 - Hypothèse forte que l'implantation est conforme à la théorie
- Solution
 - Caractérisation automatique à partir d'occurrences de MC connues

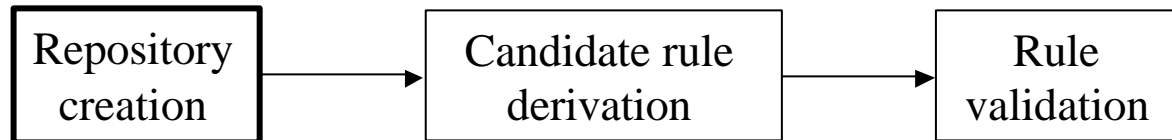
Principes des signatures numériques

- Utilisation d'attributs internes des classes
 - Taille, héritage, couplage, cohésion
- Les valeurs des attributs internes ne sont pas uniques
 - Une classe peut jouer plus d'un rôle
 - Un rôle peut être joué par plus d'une classe
- Les attributs internes sont utilisés pour réduire l'espace de recherche
 - Élimination de classes qui ne joue **évidemment** pas un rôle

Construction des signatures numériques



Construction des signatures numériques



■ Création du référentiel

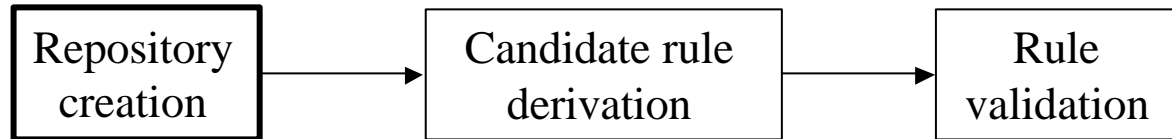
– Sources

- Études de la littérature (cf., [Bieman *et al.*, 03])
- Outils maison pour l'identification [Guéhéneuc, 01]
- Travaux dans des cours gradués

– Structure

- <program><designmotif><occurrence><role>

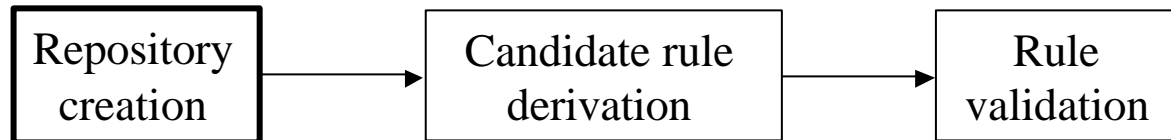
Construction des signatures numériques



■ Création du référentiel

	JHODRAW v5.1	JREFACTORY v2.6.24	JUNIT v3.7	LEXI v0.0.1 α	NETBEANS v1.0.x	QUICKUML 2001	Total	Nombre de rôles [GAM94]	Nombre de classes jouant un rôle par motif de conception	
Nombre de classes	173	575	157	127	5812	224	7 068			
Motif de conception	Nombre de micro-architectures similaires au motif de conception par programme									
Usine Abstraite					12	1	13	5	217	
Adaptateur	1	17			8		26	4	230	
Constructeur		2		1		1	4	4	24	
Commande	1				1	1	3	5	67	
Composite	1		1			2	4	4	107	
Décorateur	1		1				2	4	64	
Méthode Usine	3	1					4	4	67	
Itérateur			1		5		6	5	30	
Observateur	2		3	2		1	8	4	93	
Prototype	2						2	3	32	
Singleton	2	2	2	2		1	9	1	9	
État	2	2					4	3	32	
Stratégie	4						4	3	36	
Méthode Patron	2						2	2	36	
Visiteur		2					2	4	138	
							Total	93	55	1 182

Construction des signatures numériques

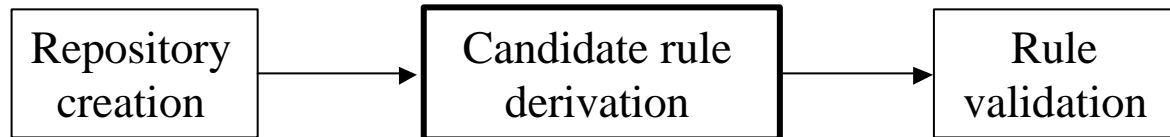


■ Création du référentiel

- Extraction des métriques pour chaque classes

	Acronymes	Descriptions	Références
Size/complexité	NM	Nombre de méthodes	[LOR 94]
	NMA	Nombre de nouvelles méthodes	[LOR 94]
	NMI	Nombre de méthodes héritées	[LOR 94]
	NMO	Nombre de méthodes surchargées	[LOR 94]
	WMC	Poids du nombre de méthodes	[CHI 93]
Filiation	CLD	Profondeur de la classe	[TEG 95]
	DIT	Profondeur de l'arbre d'héritage	[CHI 93]
	NOC	Nombre d'enfants	[CHI 93]
Cohésion	C	Connectivité	[HIT 95]
	LCOM5	Manque de cohésion des méthodes	[BRI 97b]
Couplage	ACMIC	Imports classes-méthodes des ancêtres	[BRI 97a]
	CBO	Couplage entre objets	[CHI 93]
	DCMEC	Imports classes-méthodes des fils	[BRI 97a]

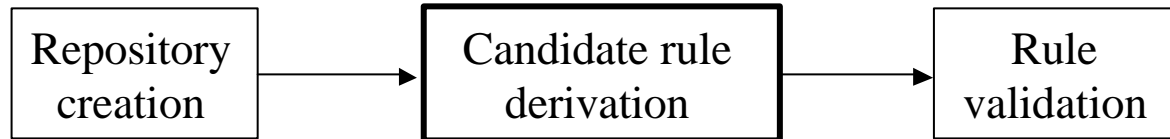
Construction des signatures numériques



■ Inférence des règles candidates

- *Rule learner*, JRIP (outil Weka)
- Ensemble de données d'apprentissage (pour chaque rôle): $4 \sim n$ cas
 - n classes jouant le rôle
 - $3 \sim n$ classes choisies aléatoirement
- Chaque cas
 - $\langle \text{metrique}_1, \text{metrique}_2, \dots, \text{metrique}_m, \text{rôle} \rangle$

Construction des signatures numériques



■ Inférence des règles candidates

– Forme des signatures

Rules for <Role>:

Signature₁, confidence

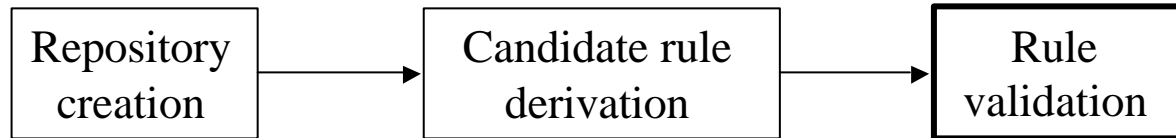
...

Signature_k, confidence

où

Signature₁: {*metrique*₁ \hat{I} *D*₁₁; ... ; *metric*_m \hat{I} *D*_{m1}}

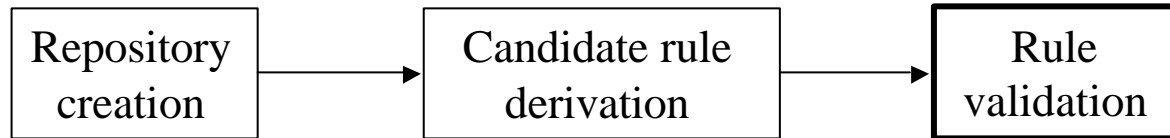
Construction des signatures numériques



■ Inférence des règles candidates

- Élimination des rôles sans un nombre significatif d'occurrences (20 sur 55)
- Validation croisée
 - Méthode *leave-one-out*
 - Élimination des rôles avec un rappel inférieur à 75% (15 sur 35)
 - Erreur pour les 20 rôles restant < 10%

Construction des signatures numériques



■ Validation des règles

Motif de conception	Rôles	Erreur (en %)	Rappel (en %)
Itérateur	Client	0,00	100,00
Observateur	Sujet	0,00	100,00
Observateur	Observateur	2,38	100,00
Méthode patron	Classe Concrète	0,00	97,60
Prototype	Prototype Concret	0,00	96,30
Décorateur	Composant Concret	4,17	89,58
Visiteur	Visiteur Concret	0,00	88,89
Stratégie	Contexte	3,70	88,89
Visiteur	Élément Concret	2,04	88,78
Singleton	Singleton	8,33	87,50
Méthode Usine	Créateur Concret	4,30	87,10
Méthode Usine	Produit Concret	3,45	86,21
Adaptateur	Cible	4,00	84,00
Composite	Feuille	6,47	82,09
Décorateur	Décorateur Concret	0,00	80,00
Itérateur	Itérateur	0,00	80,00
Commande	Receveur	6,67	80,00
État	État Concret	6,67	80,00
Stratégie	Stratégie Concrète	2,38	78,57
Commande	Commande Concrète	3,23	77,42

Utilisation des signatures numériques

- Identification structurelle par contraintes
 - $\{V, C, D\}$
 - Variables représentent les rôles d'un MC
 - Domaines contiennent toutes les classes d'un programme
- Signatures numériques réduisent les domaines
- Signatures numériques réduisent le nombre de combinaisons possibles

MC Composite

■ Sans signatures

- S = Toutes les classes
- Variables
 - *client*, $D_{cl} = S$
 - *component*, $D_{cn} = S$
 - *composite*, $D_{co} = S$
 - *leaf*, $D_{le} = S$

■ Avec signatures

- S_{cl} , S_{cn} , S_{co} et $S_{le} \subseteq S$
- Variables
 - *client*, $D_{cl} = S_{cl}$
 - *component*, $D_{cn} = S_{cn}$
 - *composite*, $D_{co} = S_{co}$
 - *leaf*, $D_{le} = S_{le}$

S_{cl} , S_{cn} , S_{co} , and S_{le} = Classes avec des signatures similaires à celles des quatre rôles

– Contraintes

- *association(client, component)*, *héritage(component, composite)*, *héritage(component, leaf)*, *composition(composite, component)*

Composite DM

■ Règle pour le rôle *Leaf*

- $NMI = 26 \wedge DIT = 5, 23/67$
- $NMI = 25 \wedge NMO = 2, 45/67$
- $DIT = 3 \wedge NM = 12, 9/67$

■ Réduction de l'espace de recherche

Signatures numériques	Nombre de classes correspondant à cette signature numérique	Réduction de l'espace de recherche (en %)
<i>NMI grand et DIT grand</i>	20	69,00
<i>NIM grand et NMO petit</i>	7	89,15
<i>DIT moyen et NM moyen</i>	10	84,50

Types de règles

- Règles de « bon sens » qui confirment l'intuition théorique des rôles
- Règles contraires au « bon sens » qui vont à l'opposé de l'intuition, résultats d'un manque de données
- Règles invérifiables par l'intuition et qui demandent une étude plus poussée



Dernières nouvelles

- Études plus complètes des signatures numériques
- Application à l'identification d'occurrences approchées de MC
- ...

Conclusion

- Caractérisation quantitative expérimentale des rôles dans les motifs de conception
- Signatures numériques pour les rôles définies par des métriques
- Signatures numériques utilisée pour améliorer l'identification de motifs
- Réduction significative du temps et des ressources nécessaires à l'identification
- Applicable à l'identification d'occurrences approchées de motifs de conception

Travail futur (une p'tite blague)



Travail futur (une p'tite blague)

- Signatures numériques de composants

Travail futur (une p'tite blague)

- Signatures numériques de composants
- Signatures numériques de frameworks

Travail futur (une p'tite blague)

- Signatures numériques de composants
- Signatures numériques de frameworks
- Signatures numériques d'aspects



Travail futur (une p'tite blague)

- Signatures numériques de composants
- Signatures numériques de frameworks
- Signatures numériques d'aspects
- Signatures numériques de refactorings

Travail futur (une p'tite blague)

- Signatures numériques de composants
- Signatures numériques de frameworks
- Signatures numériques d'aspects
- Signatures numériques de refactorings
- Signatures numériques de services



Travail futur (une p'tite blague)

- Signatures numériques de composants
- Signatures numériques de frameworks
- Signatures numériques d'aspects
- Signatures numériques de refactorings
- Signatures numériques de services
- Signatures numériques de métriques

Travail futur (une p'tite blague)

- Signatures numériques de composants
- Signatures numériques de frameworks
- Signatures numériques d'aspects
- Signatures numériques de refactorings
- Signatures numériques de services
- Signatures numériques de métriques
- ...



Données

- Précision dépend du référentiel

- P-MARt (*Pattern-like Micro-Architecture Repository*) est un référentiel de micro-architectures similaires à des motifs de conception
- P-MARt et les outils associés sont disponibles gratuitement sous licence LPGL

S'il vous plaît, contribuez !

- Information, téléchargement et contributions à <http://www.iro.umontreal.ca/~labgelo/p-mart/>