

# Intérêts des sous-(méta-)modèles pour un passage à l'échelle de l'IDM



Olivier Caron<sup>1</sup> Bernard Carré<sup>1</sup>

Gilles Vanwormhoudt<sup>1,2</sup>

<sup>1</sup>Univ. Lille, CRISAL/CAMEL

<sup>2</sup>Institut Mines Telecom Lille-Douai

10 mai 2021

- ① Les sous-modèles dans les pratiques IDM
- ② Le défi du GT sur le passage à l'échelle
- ③ Les sous-modèles pour traiter des grands modèles
- ④ Les sous-modèles pour gérer des dépôts de modèles
- ⑤ Nos travaux sur les sous-modèles
- ⑥ Conclusion
- ⑦ Bibliographie

- Depuis les années 2000, l'IDM est un domaine de recherche actif et couvre de multiples aspects :

- Depuis les années 2000, l'IDM est un domaine de recherche actif et couvre de multiples aspects :
  - ▶ Travaux sur la conception de modèles, la méta-modélisation (outils meta, DSM, ...), les approches descendantes (MDA), la vérification et validation de modèles, la transformation de modèles, les opérateurs, la comparaison de modèles...

- Depuis les années 2000, l'IDM est un domaine de recherche actif et couvre de multiples aspects :
  - ▶ Travaux sur la conception de modèles, la méta-modélisation (outils meta, DSM, ...), les approches descendantes (MDA), la vérification et validation de modèles, la transformation de modèles, les opérateurs, la comparaison de modèles...

## Constat

Implicitement ou explicitement, les sous-modèles sont concernés dans la plupart de ces travaux

# Sous-modèles

Exemple 1 : processus de construction de modèles

# Sous-modèles

Exemple 1 : processus de construction de modèles

- La construction de (méta-)modèles est un processus complexe et itératif



## Exemple 1 : processus de construction de modèles

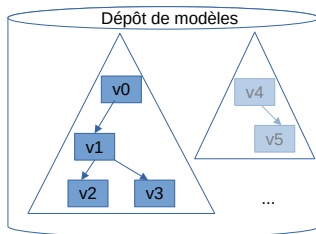
- La construction de (méta-)modèles est un processus complexe et itératif
- De nombreuses versions, branches

## Exemple 1 : processus de construction de modèles

- La construction de (méta-)modèles est un processus complexe et itératif
- De nombreuses versions, branches
- Peut aussi se construire par composition de modèles existants

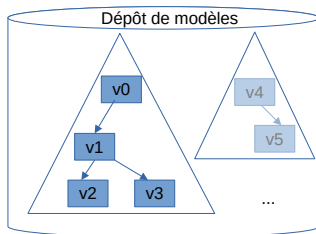
## Exemple 1 : processus de construction de modèles

- La construction de (méta-)modèles est un processus complexe et itératif
- De nombreuses versions, branches
- Peut aussi se construire par composition de modèles existants



## Exemple 1 : processus de construction de modèles

- La construction de (méta-)modèles est un processus complexe et itératif
- De nombreuses versions, branches
- Peut aussi se construire par composition de modèles existants



## Caractérisation des sous-modèles

multi-niveaux, structuration de l'espace des modèles, hiérarchie de sous-modèles, traçabilité

# Sous-modèles

## Exemple 2 : opérateurs de modèles

# Sous-modèles

## Exemple 2 : opérateurs de modèles

## Exemple 2 : opérateurs de modèles

- opérateurs pour les approches par réutilisation de sous-modèles (import, merge, ...) [6]

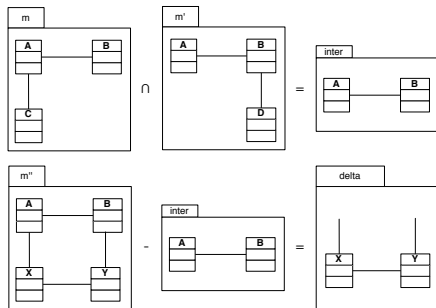
## Exemple 2 : opérateurs de modèles

- opérateurs pour les approches par réutilisation de sous-modèles (import, merge, ...) [6]
- opérateurs utiles pour certaines pratiques  
Exemple: opérateur de différence nécessaire pour la conception collaborative, versioning, ... [1, 11, 10]



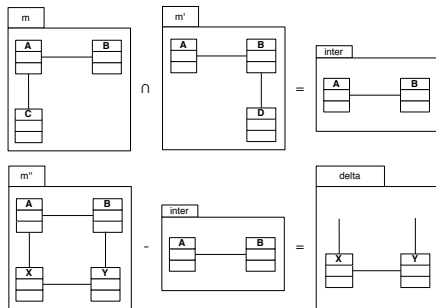
## Exemple 2 : opérateurs de modèles

- opérateurs pour les approches par réutilisation de sous-modèles (import, merge, ...) [6]
- opérateurs utiles pour certaines pratiques  
Exemple: opérateur de différence nécessaire pour la conception collaborative, versioning, ... [1, 11, 10]



## Exemple 2 : opérateurs de modèles

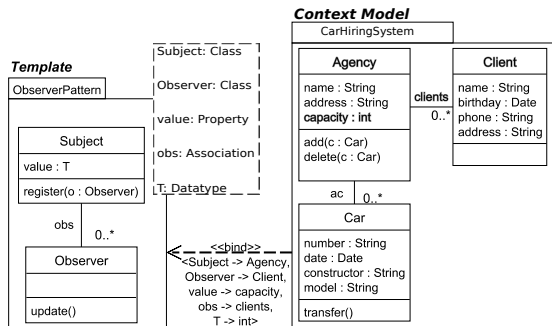
- opérateurs pour les approches par réutilisation de sous-modèles (import, merge, ...) [6]
- opérateurs utiles pour certaines pratiques  
Exemple: opérateur de différence nécessaire pour la conception collaborative, versioning, ... [1, 11, 10]



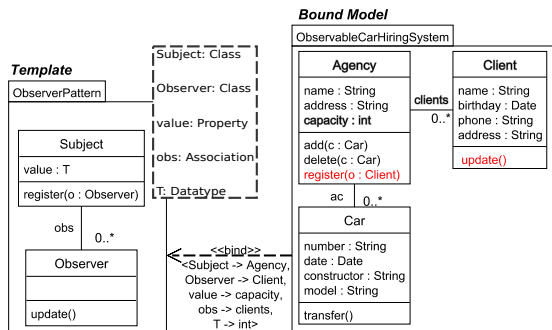
## Caractérisation des sous-modèles

fragments de modèles, modèles bien formés ou mal formés.

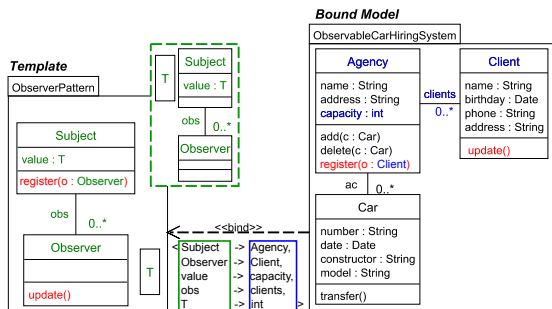
## Exemple 3 : Composition de modèles, templates aspectuels



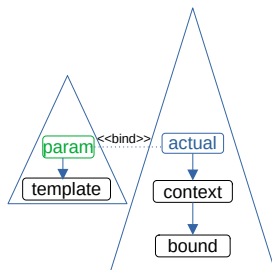
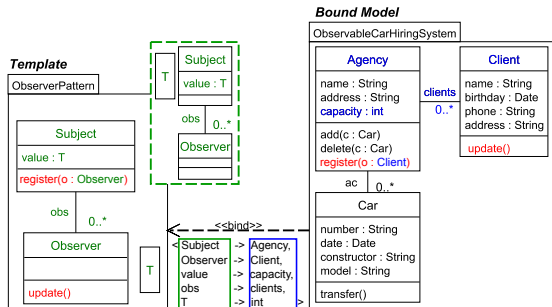
## Exemple 3 : Composition de modèles, templates aspectuels

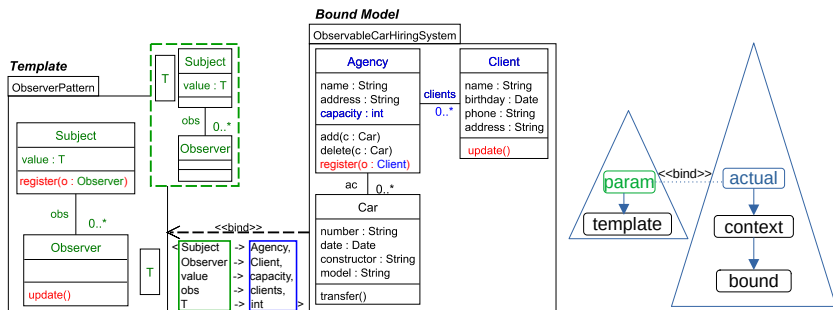


## Exemple 3 : Composition de modèles, templates aspectuels



## Exemple 3 : Composition de modèles, templates aspectuels





## Caractérisation des sous-modèles

Identifier les sous-modèles et leurs relations contribue à une meilleure compréhension et formalisation des processus associés [2, 12, 13]

- Les études et revues d'expériences régulières ([9, 7]) sur l'adoption de l'IDM dans l'industrie montrent une progression mais aussi une nécessaire prise en compte du passage à l'échelle.



- Les études et revues d'expériences régulières ([9, 7]) sur l'adoption de l'IDM dans l'industrie montrent une progression mais aussi une nécessaire prise en compte du passage à l'échelle.
- [8] présente une feuille de route sur 4 points :

- Les études et revues d'expériences régulières ([9, 7]) sur l'adoption de l'IDM dans l'industrie montrent une progression mais aussi une nécessaire prise en compte du passage à l'échelle.
- [8] présente une feuille de route sur 4 points :
  - ▶ *"being able to construct large models and domain specific languages in a systematic manner;*

- Les études et revues d'expériences régulières ([9, 7]) sur l'adoption de l'IDM dans l'industrie montrent une progression mais aussi une nécessaire prise en compte du passage à l'échelle.
- [8] présente une feuille de route sur 4 points :
  - ▶ *"being able to construct large models and domain specific languages in a systematic manner;*
  - ▶ *enabling large teams of modellers to construct and refine large models in a collaborative manner;*

- Les études et revues d'expériences régulières ([9, 7]) sur l'adoption de l'IDM dans l'industrie montrent une progression mais aussi une nécessaire prise en compte du passage à l'échelle.
- [8] présente une feuille de route sur 4 points :
  - ▶ *"being able to construct large models and domain specific languages in a systematic manner;*
  - ▶ *enabling large teams of modellers to construct and refine large models in a collaborative manner;*
  - ▶ *advancing the state of the art in model querying and transformations tools so that they can cope with large models (of the order millions of model elements);*

- Les études et revues d'expériences régulières ([9, 7]) sur l'adoption de l'IDM dans l'industrie montrent une progression mais aussi une nécessaire prise en compte du passage à l'échelle.
- [8] présente une feuille de route sur 4 points :
  - ▶ *"being able to construct large models and domain specific languages in a systematic manner;*
  - ▶ *enabling large teams of modellers to construct and refine large models in a collaborative manner;*
  - ▶ *advancing the state of the art in model querying and transformations tools so that they can cope with large models (of the order millions of model elements);*
  - ▶ *providing an infrastructure for efficient storage, indexing and retrieval of such models"*

- Les études et revues d'expériences régulières ([9, 7]) sur l'adoption de l'IDM dans l'industrie montrent une progression mais aussi une nécessaire prise en compte du passage à l'échelle.
- [8] présente une feuille de route sur 4 points :
  - ▶ *"being able to construct large models and domain specific languages in a systematic manner;*
  - ▶ *enabling large teams of modellers to construct and refine large models in a collaborative manner;*
  - ▶ *advancing the state of the art in model querying and transformations tools so that they can cope with large models (of the order millions of model elements);*
  - ▶ *providing an infrastructure for efficient storage, indexing and retrieval of such models"*
- Grands modèles Vs grands volumes de modèles

# Sous-modèles et grands modèles

(cf "related works" de [4] )

- De plus en plus de grands modèles :

- De plus en plus de grands modèles :
  - ▶ Des modèles complexes composés de plusieurs sous-modèles interconnectés tels que les vues UML (dont le méta-modèle est lui-même complexe).



- De plus en plus de grands modèles :
  - ▶ Des modèles complexes composés de plusieurs sous-modèles interconnectés tels que les vues UML (dont le méta-modèle est lui-même complexe).
  - ▶ Des modèles, résultats de processus de rétro-ingénierie [3]

(cf "related works" de [4] )

- De plus en plus de grands modèles :
  - ▶ Des modèles complexes composés de plusieurs sous-modèles interconnectés tels que les vues UML (dont le méta-modèle est lui-même complexe).
  - ▶ Des modèles, résultats de processus de rétro-ingénierie [3]
- Sous-modèles et **construction** de grands modèles:  
Toutes les nombreuses approches de construction par composition de sous-modèles existants : assemblage, vues, aspects, templates,...

(cf "related works" de [4] )

- De plus en plus de grands modèles :
  - ▶ Des modèles complexes composés de plusieurs sous-modèles interconnectés tels que les vues UML (dont le méta-modèle est lui-même complexe).
  - ▶ Des modèles, résultats de processus de rétro-ingénierie [3]
- Sous-modèles et **construction** de grands modèles:  
Toutes les nombreuses approches de construction par composition de sous-modèles existants : assemblage, vues, aspects, templates,...
- Sous-modèles et **utilisation** de grands modèles : de nombreuses approches de décomposition : extraction, slicing, pruning, ...

(cf "related works" de [4] )

- De plus en plus de grands modèles :
  - ▶ Des modèles complexes composés de plusieurs sous-modèles interconnectés tels que les vues UML (dont le méta-modèle est lui-même complexe).
  - ▶ Des modèles, résultats de processus de rétro-ingénierie [3]
- Sous-modèles et **construction** de grands modèles:  
Toutes les nombreuses approches de construction par composition de sous-modèles existants : assemblage, vues, aspects, templates,...
- Sous-modèles et **utilisation** de grands modèles : de nombreuses approches de décomposition : extraction, slicing, pruning, ...
  - ▶ réduction des échanges de données (édition collaborative)

(cf "related works" de [4] )

- De plus en plus de grands modèles :
  - ▶ Des modèles complexes composés de plusieurs sous-modèles interconnectés tels que les vues UML (dont le méta-modèle est lui-même complexe).
  - ▶ Des modèles, résultats de processus de rétro-ingénierie [3]
- Sous-modèles et **construction** de grands modèles:  
Toutes les nombreuses approches de construction par composition de sous-modèles existants : assemblage, vues, aspects, templates, ...
- Sous-modèles et **utilisation** de grands modèles : de nombreuses approches de décomposition : extraction, slicing, pruning, ...
  - ▶ réduction des échanges de données (édition collaborative)
  - ▶ optimisation des opérateurs coûteux sur des modèles plus simples (checking, transformation, ...)

- Un exemple concret : <https://repository.genymodel.com>  
plus d'1 million de (méta-)modèles, 12 types différents

- Un exemple concret : <https://repository.genymodel.com>  
plus d'1 million de (méta-)modèles, 12 types différents
- Les potentiels apports à étudier :

- Un exemple concret : <https://repository.genmymodel.com>  
plus d'1 million de (méta-)modèles, 12 types différents
- Les potentiels apports à étudier :
  - ▶ Structuration des dépôts grâce



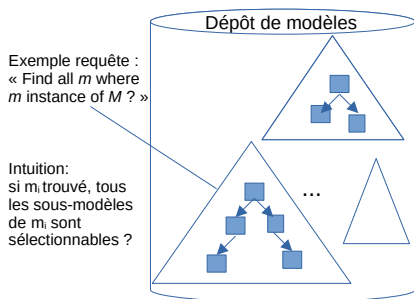
- Un exemple concret : <https://repository.genmymodel.com>  
plus d'1 million de (méta-)modèles, 12 types différents
- Les potentiels apports à étudier :
  - ▶ Structuration des dépôts grâce
    - Hiérarchies de sous-modèles

- Un exemple concret : <https://repository.genmymodel.com>  
plus d'1 million de (méta-)modèles, 12 types différents
- Les potentiels apports à étudier :
  - ▶ Structuration des dépôts grâce
    - Hiérarchies de sous-modèles
    - Traçabilité des modèles  
(construit par quel  
opérateur, ...)

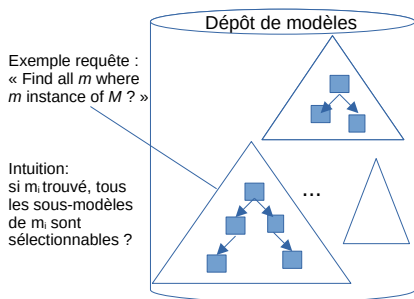
- Un exemple concret : <https://repository.genymodel.com>  
plus d'1 million de (méta-)modèles, 12 types différents
- Les potentiels apports à étudier :
  - ▶ Structuration des dépôts grâce
    - Hiérarchies de sous-modèles
    - Traçabilité des modèles  
(construit par quel  
opérateur, ...)
    - Exploiter les sous-modèles pour  
l'indexation, recherche, ...

- Un exemple concret : <https://repository.genymodel.com>  
plus d'1 million de (méta-)modèles, 12 types différents
- Les potentiels apports à étudier :
  - ▶ Structuration des dépôts grâce
    - Hiérarchies de sous-modèles
    - Traçabilité des modèles  
(construit par quel  
opérateur, ...)
    - Exploiter les sous-modèles pour  
l'indexation, recherche, ...
  - ▶ Établir des propriétés pour

- Un exemple concret : <https://repository.genymodel.com>  
plus d'1 million de (méta-)modèles, 12 types différents
- Les potentiels apports à étudier :
  - ▶ Structuration des dépôts grâce
    - Hiérarchies de sous-modèles
    - Traçabilité des modèles  
(construit par quel opérateur, ...)
    - Exploiter les sous-modèles pour l'indexation, recherche, ...
  - ▶ Établir des propriétés pour
    - Factoriser les recherches



- Un exemple concret : <https://repository.genymodel.com>  
plus d'1 million de (méta-)modèles, 12 types différents
- Les potentiels apports à étudier :
  - ▶ Structuration des dépôts grâce
    - Hiérarchies de sous-modèles
    - Traçabilité des modèles  
(construit par quel opérateur, ...)
    - Exploiter les sous-modèles pour l'indexation, recherche, ...
  - ▶ Établir des propriétés pour
    - Factoriser les recherches
    - Optimiser les temps de calcul des opérateurs.



- Dans [4, 5], une formalisation algébrique de sous-modèle avec ses relations et propriétés :

- Dans [4, 5], une formalisation algébrique de sous-modèle avec ses relations et propriétés :

## Definition



- Dans [4, 5], une formalisation algébrique de sous-modèle avec ses relations et propriétés :

## Definition

- ▶ A model  $m$  is a couple  $(\tilde{m}, \sqsubseteq_m)$  such as:

- Dans [4, 5], une formalisation algébrique de sous-modèle avec ses relations et propriétés :

## Definition

- ▶ A model  $m$  is a couple  $(\tilde{m}, \sqsubseteq_m)$  such as:
  - $\tilde{m} \subseteq ModelElements$  is the set of model elements of  $m$

- Dans [4, 5], une formalisation algébrique de sous-modèle avec ses relations et propriétés :

## Definition

- ▶ A model  $m$  is a couple  $(\tilde{m}, \sqsubseteq_m)$  such as:
  - $\tilde{m} \subseteq ModelElements$  is the set of model elements of  $m$
  - $\sqsubseteq_m: \tilde{m} \times \tilde{m}$  is a partial ordering relationship, local to  $\tilde{m}$ , which translates the dependency constraints asserted by  $m$  on its model elements.

- Dans [4, 5], une formalisation algébrique de sous-modèle avec ses relations et propriétés :

## Definition

- ▶ A model  $m$  is a couple  $(\tilde{m}, \sqsubseteq_m)$  such as:
  - $\tilde{m} \subseteq ModelElements$  is the set of model elements of  $m$
  - $\sqsubseteq_m: \tilde{m} \times \tilde{m}$  is a partial ordering relationship, local to  $\tilde{m}$ , which translates the dependency constraints asserted by  $m$  on its model elements.
- ▶ Fonction totale  $meta : ModelElements \rightarrow ModelElements$   
 $x \mapsto X$  such as  $X$  is the metaconstruct that  $x$  instantiates

- Dans [4, 5], une formalisation algébrique de sous-modèle avec ses relations et propriétés :

## Definition

- ▶ A model  $m$  is a couple  $(\tilde{m}, \sqsubseteq_m)$  such as:
  - $\tilde{m} \subseteq ModelElements$  is the set of model elements of  $m$
  - $\sqsubseteq_m: \tilde{m} \times \tilde{m}$  is a partial ordering relationship, local to  $\tilde{m}$ , which translates the dependency constraints asserted by  $m$  on its model elements.
- ▶ Fonction totale  $meta : ModelElements \rightarrow ModelElements$   
 $x \mapsto X$  such as  $X$  is the metaconstruct that  $x$  instantiates
- Représentation uniforme des modèles, méta-modèles bien formés ou pas.

- Dans [4, 5], une formalisation algébrique de sous-modèle avec ses relations et propriétés :

## Definition

- ▶ A model  $m$  is a couple  $(\tilde{m}, \sqsubseteq_m)$  such as:
  - $\tilde{m} \subseteq ModelElements$  is the set of model elements of  $m$
  - $\sqsubseteq_m: \tilde{m} \times \tilde{m}$  is a partial ordering relationship, local to  $\tilde{m}$ , which translates the dependency constraints asserted by  $m$  on its model elements.
- ▶ Fonction totale  $meta : ModelElements \rightarrow ModelElements$   
 $x \mapsto X$  such as  $X$  is the metaconstruct that  $x$  instantiates
- Représentation uniforme des modèles, méta-modèles bien formés ou pas.
- Relations imbriquées de sous-modèles :  
 $included \prec closed \prec covariant \prec invariant$

- Dans [4, 5], une formalisation algébrique de sous-modèle avec ses relations et propriétés :

## Definition

- ▶ A model  $m$  is a couple  $(\tilde{m}, \sqsubseteq_m)$  such as:
  - $\tilde{m} \subseteq ModelElements$  is the set of model elements of  $m$
  - $\sqsubseteq_m: \tilde{m} \times \tilde{m}$  is a partial ordering relationship, local to  $\tilde{m}$ , which translates the dependency constraints asserted by  $m$  on its model elements.
- ▶ Fonction totale  $meta : ModelElements \rightarrow ModelElements$   
 $x \mapsto X$  such as  $X$  is the metaconstruct that  $x$  instantiates
- Représentation uniforme des modèles, méta-modèles bien formés ou pas.
- Relations imbriquées de sous-modèles :  
 $included \prec closed \prec covariant \prec invariant$
- Tests de conformité :  $member \prec well\_formed$

- Dans [4, 5], une formalisation algébrique de sous-modèle avec ses relations et propriétés :

## Definition

- ▶ A model  $m$  is a couple  $(\tilde{m}, \sqsubseteq_m)$  such as:
  - $\tilde{m} \subseteq ModelElements$  is the set of model elements of  $m$
  - $\sqsubseteq_m: \tilde{m} \times \tilde{m}$  is a partial ordering relationship, local to  $\tilde{m}$ , which translates the dependency constraints asserted by  $m$  on its model elements.
- ▶ Fonction totale  $meta : ModelElements \rightarrow ModelElements$   
 $x \mapsto X$  such as  $X$  is the metaconstruct that  $x$  instantiates
- Représentation uniforme des modèles, méta-modèles bien formés ou pas.
- Relations imbriquées de sous-modèles :  
*included*  $\prec$  *closed*  $\prec$  *covariant*  $\prec$  *invariant*
- Tests de conformité : *member*  $\prec$  *well\_formed*
- Formalisation d'opérateurs (*closure*, *extract*, *proper*)

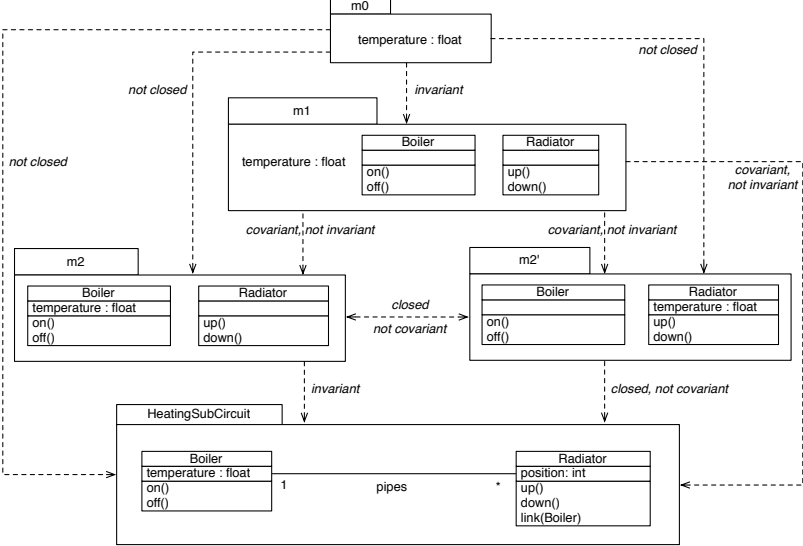


- Dans [4, 5], une formalisation algébrique de sous-modèle avec ses relations et propriétés :

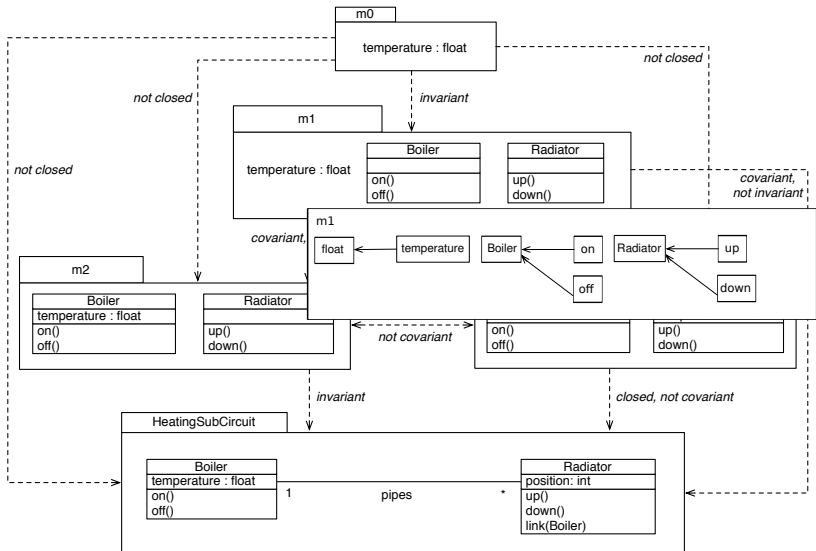
## Definition

- ▶ A model  $m$  is a couple  $(\tilde{m}, \sqsubseteq_m)$  such as:
  - $\tilde{m} \subseteq ModelElements$  is the set of model elements of  $m$
  - $\sqsubseteq_m: \tilde{m} \times \tilde{m}$  is a partial ordering relationship, local to  $\tilde{m}$ , which translates the dependency constraints asserted by  $m$  on its model elements.
- ▶ Fonction totale  $meta : ModelElements \rightarrow ModelElements$   
 $x \mapsto X$  such as  $X$  is the metaconstruct that  $x$  instantiates
- Représentation uniforme des modèles, méta-modèles bien formés ou pas.
- Relations imbriquées de sous-modèles :  
*included*  $\prec$  *closed*  $\prec$  *covariant*  $\prec$  *invariant*
- Tests de conformité : *member*  $\prec$  *well\_formed*
- Formalisation d'opérateurs (*closure*, *extract*, *proper*)
- Propriétés établies telles que la transitivité de la relation d'invariance.

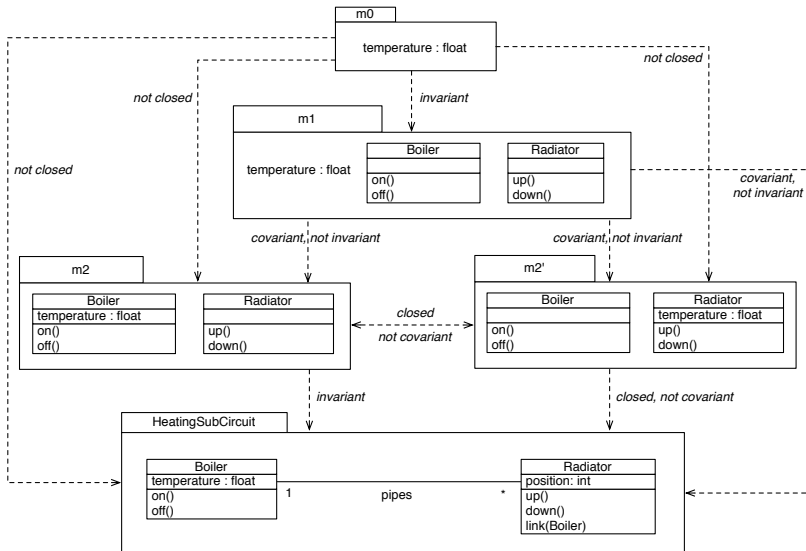
# Les relations de sous-modèles (2/3)



# Les relations de sous-modèles (2/3)

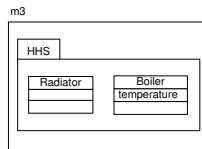
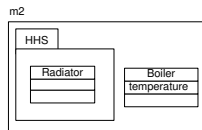
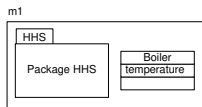


# Les relations de sous-modèles (2/3)



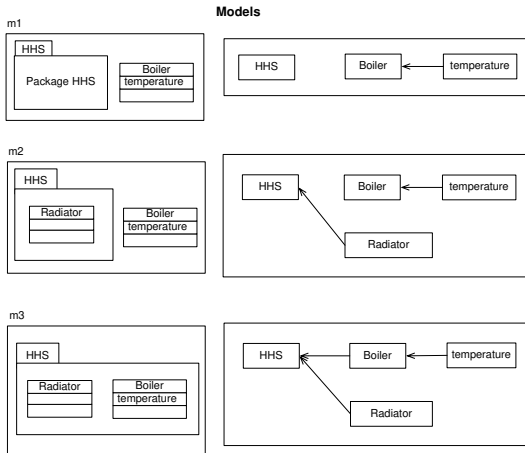
# Les relations de sous-méta-modèles (2/3)

## Models

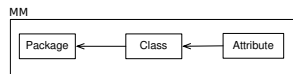


## Proper metamodels

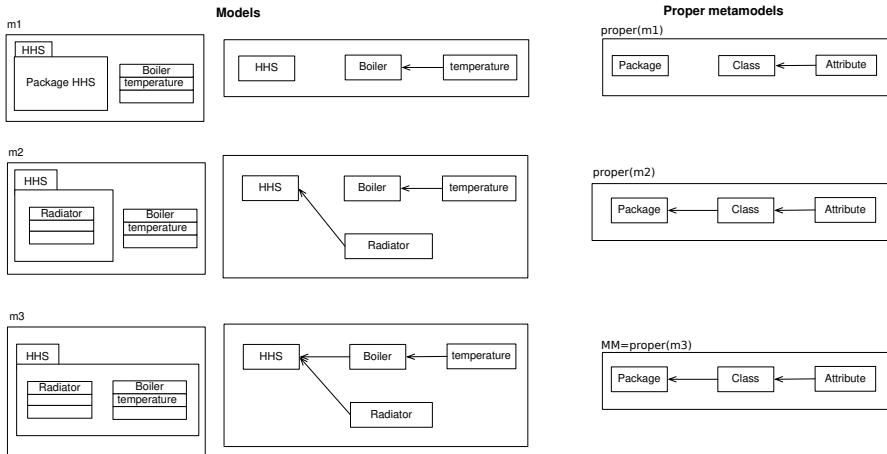
# Les relations de sous-méta-modèles (2/3)



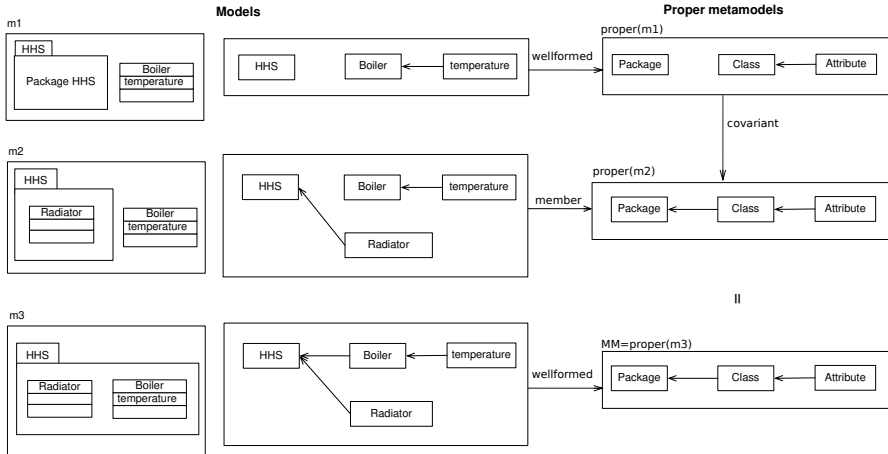
## Proper metamodels



# Les relations de sous-méta-modèles (2/3)

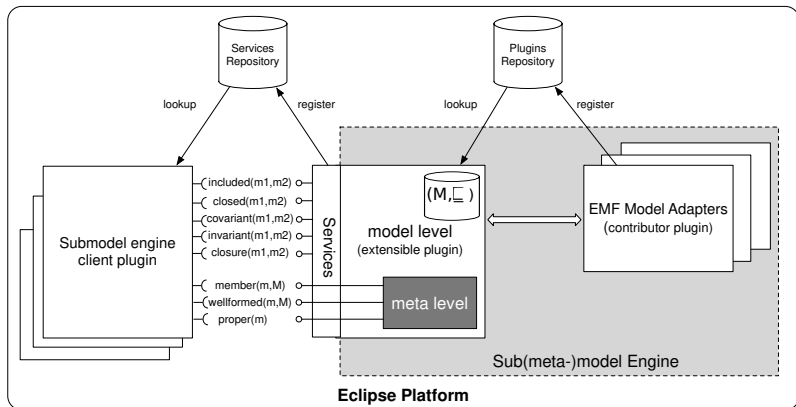


# Les relations de sous-méta-modèles (2/3)





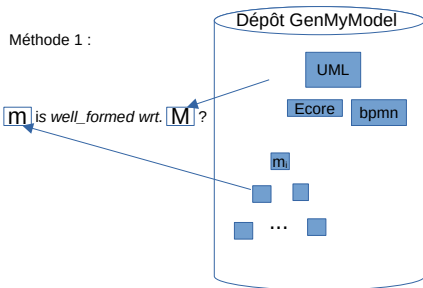
- Une architecture extensible de plugins Eclipse :



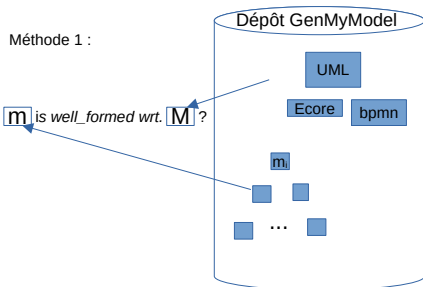
- Dans [5], une étude quantitative sur plus de 3000 modèles du dépôt "GenMyModel".

- Dans [5], une étude quantitative sur plus de 3000 modèles du dépôt "GenMyModel".

- Dans [5], une étude quantitative sur plus de 3000 modèles du dépôt "GenMyModel".



- Dans [5], une étude quantitative sur plus de 3000 modèles du dépôt "GenMyModel".



Type Modèle	Nb	Méth. 1 (ms)
Class	1791	67.15
UseCase	500	37.57
Sequence	397	8.10
Activity	336	31.69
BPMN	29	16.10
Ecore	16	4.25

### Stated Property 8 of [5]

Let  $M$  a metamodel,  $m$  a model,  $m$  is a well formed model of  $proper(m)$  and  $proper(m)$  is an invariant submodel of  $M \Leftrightarrow m$  is a well formed model of  $M$ .

### Stated Property 8 of [5]

Let  $M$  a metamodel,  $m$  a model,  $m$  is a well formed model of  $proper(m)$  and  $proper(m)$  is an invariant submodel of  $M \Leftrightarrow m$  is a well formed model of  $M$ .

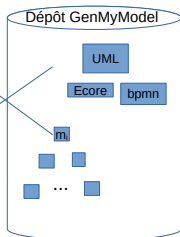
## Stated Property 8 of [5]

Let  $M$  a metamodel,  $m$  a model,  $m$  is a well formed model of  $proper(m)$  and  $proper(m)$  is an invariant submodel of  $M \Leftrightarrow m$  is a well formed model of  $M$ .

Méthode 2 :

$pmm = proper(m)$

Prop. :  
 $m$  is well\_formed wrt.  $M$  iff  
 $pmm$  is invariant in  $M$   
and  
 $m$  is well\_formed wrt.  $pmm$





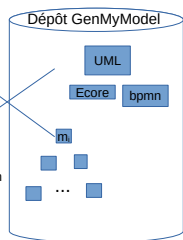
### Stated Property 8 of [5]

Let  $M$  a metamodel,  $m$  a model,  $m$  is a well formed model of  $proper(m)$  and  $proper(m)$  is an invariant submodel of  $M \Leftrightarrow m$  is a well formed model of  $M$ .

Méthode 2 :

$pmm = proper(m)$

Prop. :  
 $m$  is well\_formed wrt.  $M$  iff  
 $pmm$  is invariant in  $M$   
 and  
 $m$  is well\_formed wrt.  $pmm$



Type Modèle	Nb	Méth. 1 (ms)	Méth. 2 (ms)	Gain (%)
Class	1791	67.15	28.7	57.26
UseCase	500	37.57	9.08	75.83
Sequence	397	8.10	3.64	54.99
Activity	336	31.69	7.57	76.11
BPMN	29	16.10	8.10	49.68
Ecore	16	4.25	3.38	20.59

- Vers de meilleures performances par capitalisation de sous-(méta-)modèles et hiérarchies de sous-modèles invariants

- Vers de meilleures performances par capitalisation de sous-(méta-)modèles et hiérarchies de sous-modèles invariants

- Vers de meilleures performances par capitalisation de sous-(méta-)modèles et hiérarchies de sous-modèles invariants

- Vers de meilleures performances par capitalisation de sous-(méta-)modèles et hiérarchies de sous-modèles invariants

### Stated Property 9 of [5]

Let  $M$  and  $SM$  two metamodels,  $m$  a well formed w.r.t.  $SM$ ,  $SM$  is an invariant submetamodel of  $M \Rightarrow m$  is a well formed model of  $M$ .

- Vers de meilleures performances par capitalisation de sous-(méta-)modèles et hiérarchies de sous-modèles invariants

### Stated Property 9 of [5]

Let  $M$  and  $SM$  two metamodels,  $m$  a well formed w.r.t.  $SM$ ,  $SM$  is an invariant submetamodel of  $M \Rightarrow m$  is a well formed model of  $M$ .

### Stated Property 15 of [5]

Let  $m$  a well formed w.r.t  $M$ ,  $sm$  is an invariant submodel of  $m \Rightarrow sm$  is well formed w.r.t  $M$ .

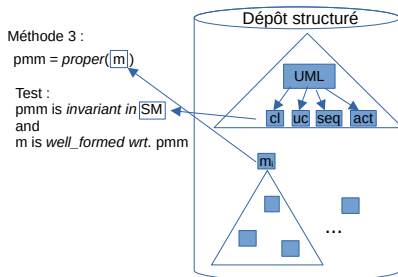
- Vers de meilleures performances par capitalisation de sous-(méta-)modèles et hiérarchies de sous-modèles invariants

### Stated Property 9 of [5]

Let  $M$  and  $SM$  two metamodellers,  $m$  a well formed w.r.t.  $SM$ ,  $SM$  is an invariant submetamodel of  $M \Rightarrow m$  is a well formed model of  $M$ .

### Stated Property 15 of [5]

Let  $m$  a well formed w.r.t.  $M$ ,  $sm$  is an invariant submodel of  $m \Rightarrow sm$  is well formed w.r.t.  $M$ .



- Vers de meilleures performances par capitalisation de sous-(méta-)modèles et hiérarchies de sous-modèles invariants

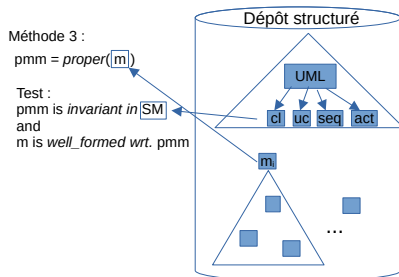
### Stated Property 9 of [5]

Let  $M$  and  $SM$  two metamodellers,  $m$  a well formed w.r.t.  $SM$ ,  $SM$  is an invariant submetamodel of  $M \Rightarrow m$  is a well formed model of  $M$ .

### Stated Property 15 of [5]

Let  $m$  a well formed w.r.t.  $M$ ,  $sm$  is an invariant submodel of  $m \Rightarrow sm$  is well formed w.r.t.  $M$ .

- [5] introduit également des fonctions de recherche de sous-modèles et mécanismes d'indexation





- Présence importante des sous-modèles dans les diverses pratiques IDM.

- Présence importante des sous-modèles dans les diverses pratiques IDM.
- Structuration de l'espace des modèles

- Présence importante des sous-modèles dans les diverses pratiques IDM.
- Structuration de l'espace des modèles
- La formulation algébrique et ses propriétés établies des relations d'inclusion de sous-modèles favorisent une gestion efficace et uniforme des dépôts de modèles

- Présence importante des sous-modèles dans les diverses pratiques IDM.
- Structuration de l'espace des modèles
- La formulation algébrique et ses propriétés établies des relations d'inclusion de sous-modèles favorisent une gestion efficace et uniforme des dépôts de modèles
- Il convient de poursuivre l'étude des opérateurs issus des pratiques MDE afin d'établir des propriétés pour une meilleure efficacité.

- [1] M. Alanen and I. Porres. “Difference and Union of Models”. In: *Proceedings of 6th International Conference on the Unified Modeling Language, Modeling Languages and Applications (UML'03)*. Vol. 2863. LNCS. Springer, 2003, pp. 2–17.
- [2] M. Allon et al. “Isolating and Reusing Template Instances in UML”. In: *Proceeding of 12th European Conference on Modelling Foundations and Applications, ECMFA, Austria*. 2016.
- [3] Hugo Bruneliere et al. “MoDisco: a Model Driven Reverse Engineering Framework”. In: *Information and Software Technology* 56.8 (Aug. 2014), pp. 1012–1032. DOI: 10.1016/j.infsof.2014.04.007. URL: <https://hal.inria.fr/hal-00972632>.
- [4] B. Carré, G. Vanwormhoudt, and O. Caron. “From subsets of model elements to submodels: A characterization of submodels and their properties”. In: *Software & Systems Modeling (Ed. Springer)* 14.2 (2015), pp. 861–887.

- [5] B. Carré, G. Vanwormhoudt, and O. Caron. “On submodels and submetamodels with their relation: A uniform formalization through inclusion properties”. In: *Software & Systems Modeling (Ed. Springer)* 17.4 (2018), pp. 1105–1137.
- [6] Jürgen Dingel, Zinovy Diskin, and Alanna Zito. “Understanding and improving UML package merge”. In: *Software and System Modeling* 7.4 (2008), pp. 443–467. DOI: 10.1007/s10270-007-0073-9. URL: <https://doi.org/10.1007/s10270-007-0073-9>.
- [7] John Edward Hutchinson et al. “Empirical assessment of MDE in industry”. In: *Proceedings of the 33rd International Conference on Software Engineering, ICSE 2011, Waikiki, Honolulu , HI, USA, May 21-28, 2011*. Ed. by Richard N. Taylor, Harald C. Gall, and Nenad Medvidovic. ACM, 2011, pp. 471–480. DOI: 10.1145/1985793.1985858. URL: <https://doi.org/10.1145/1985793.1985858>.

- [8] D.S Kolovos et al. “A research roadmap towards achieving scalability in model driven engineering”. In: *Proceedings of the Workshop on Scalability in Model Driven Engineering*. Budapest, Hungary: ACM, June 2013, 2:1–2:10. DOI: 10.1145/2487766.2487768.
- [9] Parastoo Mohagheghi and Vegard Dehlen. “Where Is the Proof? - A Review of Experiences from Applying MDE in Industry”. In: *Model Driven Architecture - Foundations and Applications, 4th European Conference, ECMDA-FA 2008, Berlin, Germany, June 9-13, 2008. Proceedings*. Ed. by Ina Schieferdecker and Alan Hartman. Vol. 5095. Lecture Notes in Computer Science. Springer, 2008, pp. 432–443. DOI: 10.1007/978-3-540-69100-6\\_31. URL: [https://doi.org/10.1007/978-3-540-69100-6%5C\\_31](https://doi.org/10.1007/978-3-540-69100-6%5C_31).
- [10] P. Sriplakich, X. Blanc, and M.P. Gervais. “Collaborative software engineering on large-scale models: requirements and experience in ModelBus”. In: *Proceedings of the 2008 ACM symposium on Applied computing*. ACM, 2008, pp. 674–681.

- [11] G. Taentzer et al. “A fundamental approach to model versioning based on graph modifications: from theory to implementation”. English. In: *Software and Systems Modeling* (2012), pp. 1–34. ISSN: 1619-1366. DOI: 10.1007/s10270-012-0248-x. URL: <http://dx.doi.org/10.1007/s10270-012-0248-x>.
- [12] G. Vanwormhoudt, O. Caron, and B. Carré. “Aspectual templates in UML - Enhancing the semantics of UML templates in OCL”. In: *Software and System Modeling (Ed. Springer)* 16.2 (2017), pp. 469–497. URL: <https://doi.org/10.1007/s10270-015-0463-3>.
- [13] Gilles Vanwormhoudt et al. “Template based model engineering in UML”. In: *MoDELS '20: ACM/IEEE 23rd International Conference on Model Driven Engineering Languages and Systems, Virtual Event, Canada, 18-23 October, 2020*. Ed. by Eugene Syriani et al. ACM, 2020, pp. 47–56. DOI: 10.1145/3365438.3410988. URL: <https://doi.org/10.1145/3365438.3410988>.