SUPPORTING INCREMENTAL CHANGES IN LARGE MODELS

Jannik Laval, Simon Denier, Stéphane Ducasse
RMod Team - INRIA

Andy Kellens
Software Languages Lab - VUB

ROADMAP

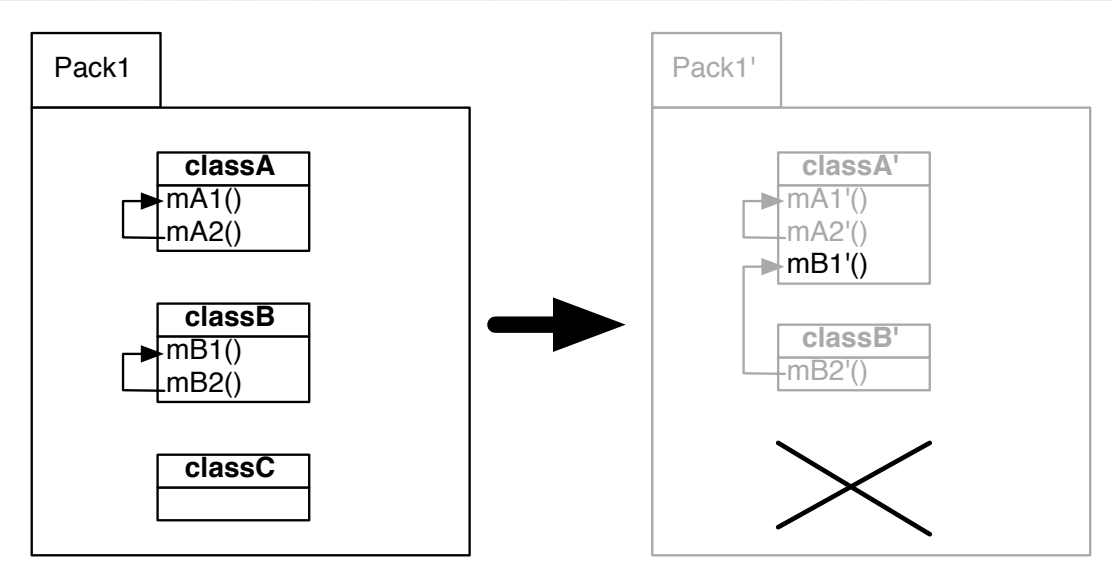
- Problem
- Different naive approaches
  - copy
  - delta
  - lookup
- Our proposal

PROBLEM

- How to predict impact of changes ?
- How to predict and compare different futures ?
- In large models.

EXAMPLE

- move mB1() in classA.
- remove classC.

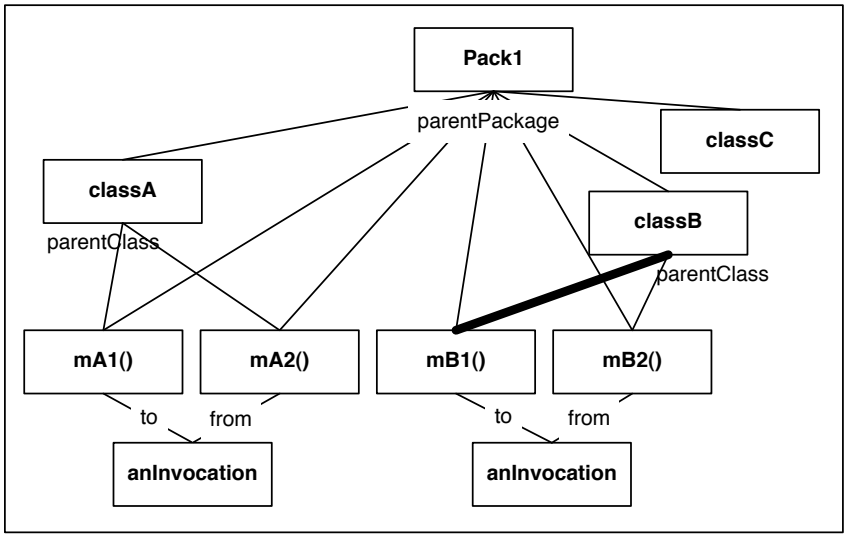


DIFFERENT NAIVE APPROACHES

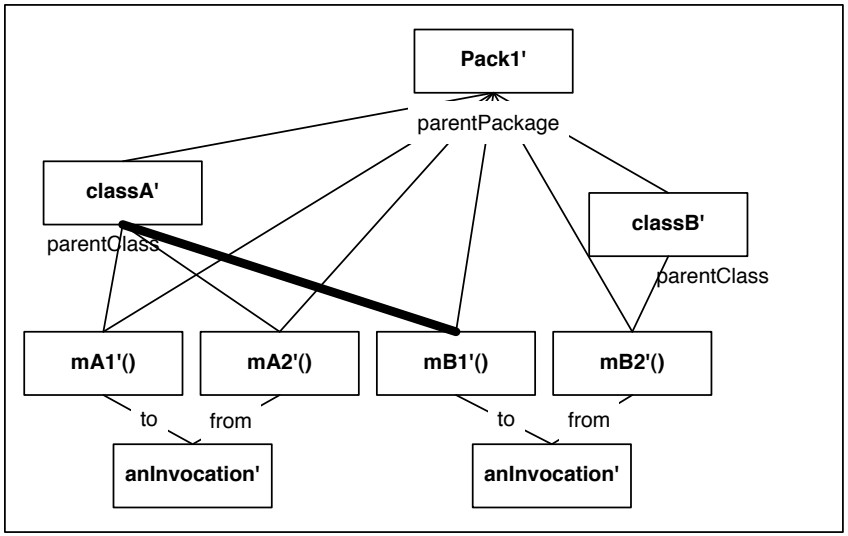
- Copy
- Delta
- LookUp

COPY

Model 1



Model 2

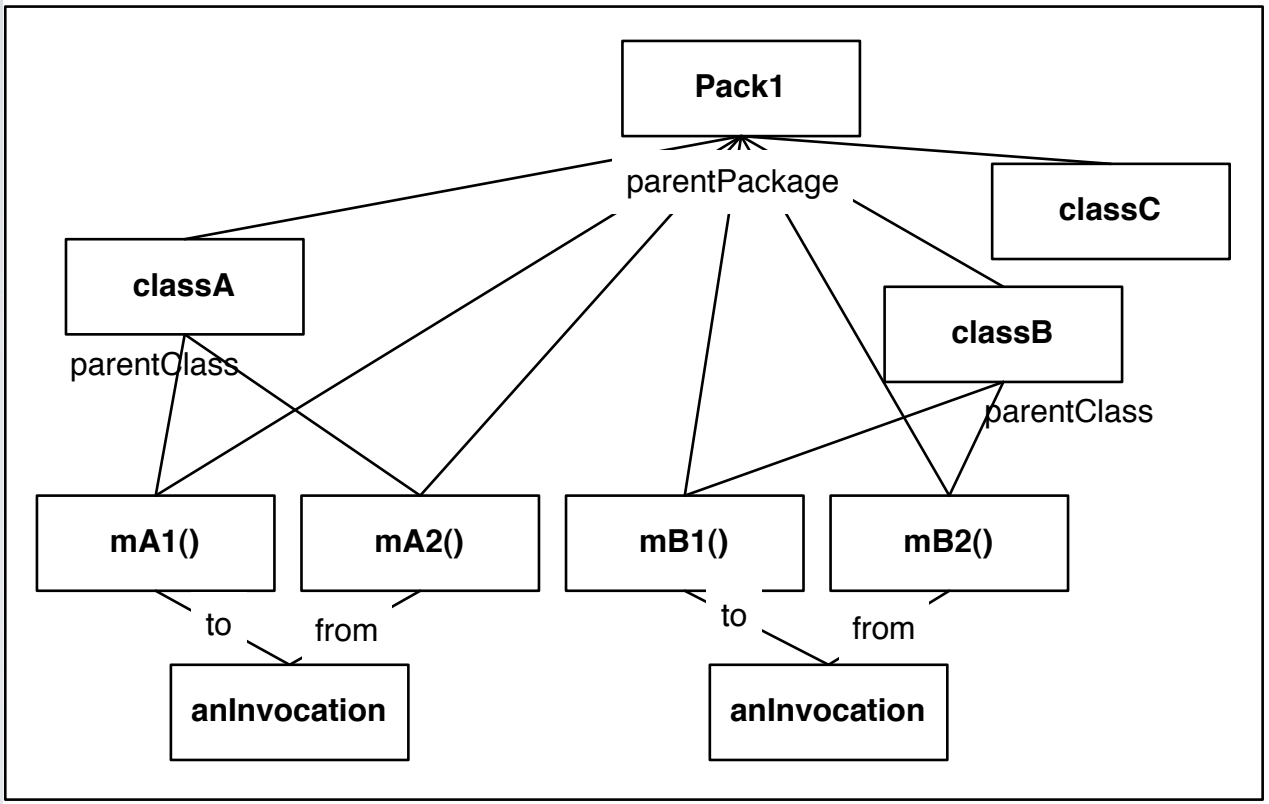


COPY

- Pros
  - each version is complete
  - easy to use and to transform
- Cons
  - high creation time
  - high memory usage

DELTA

Model 1



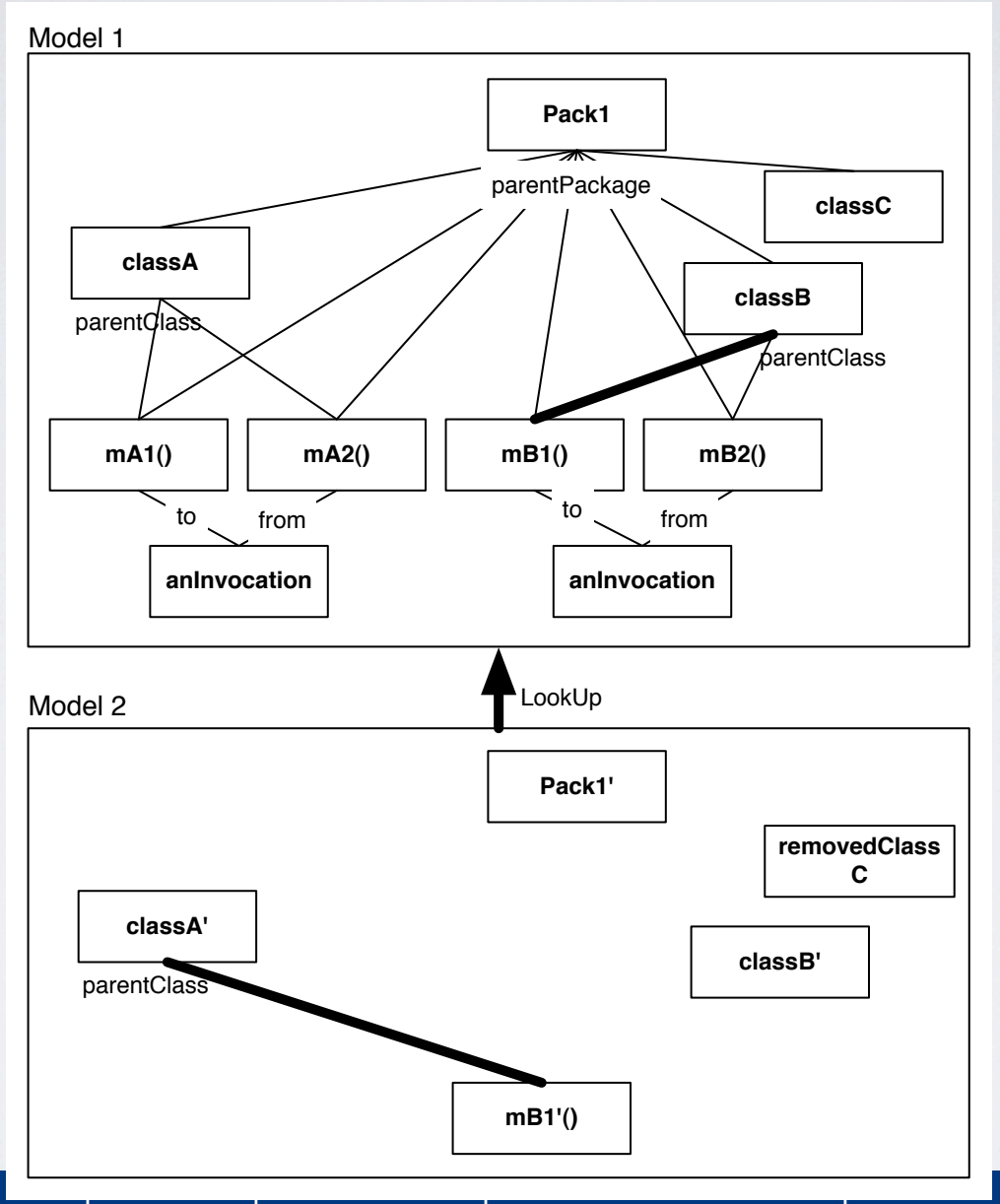
Model 2

- removeLinkBetween: classB and: mB1()
- createLinkBetween: classA and: mB1()
- removeClass: classC

DELTA

- Pros
  - each version is represented by its changes
  - fast to create a version
- Cons
  - slow to access a version

LOOKUP

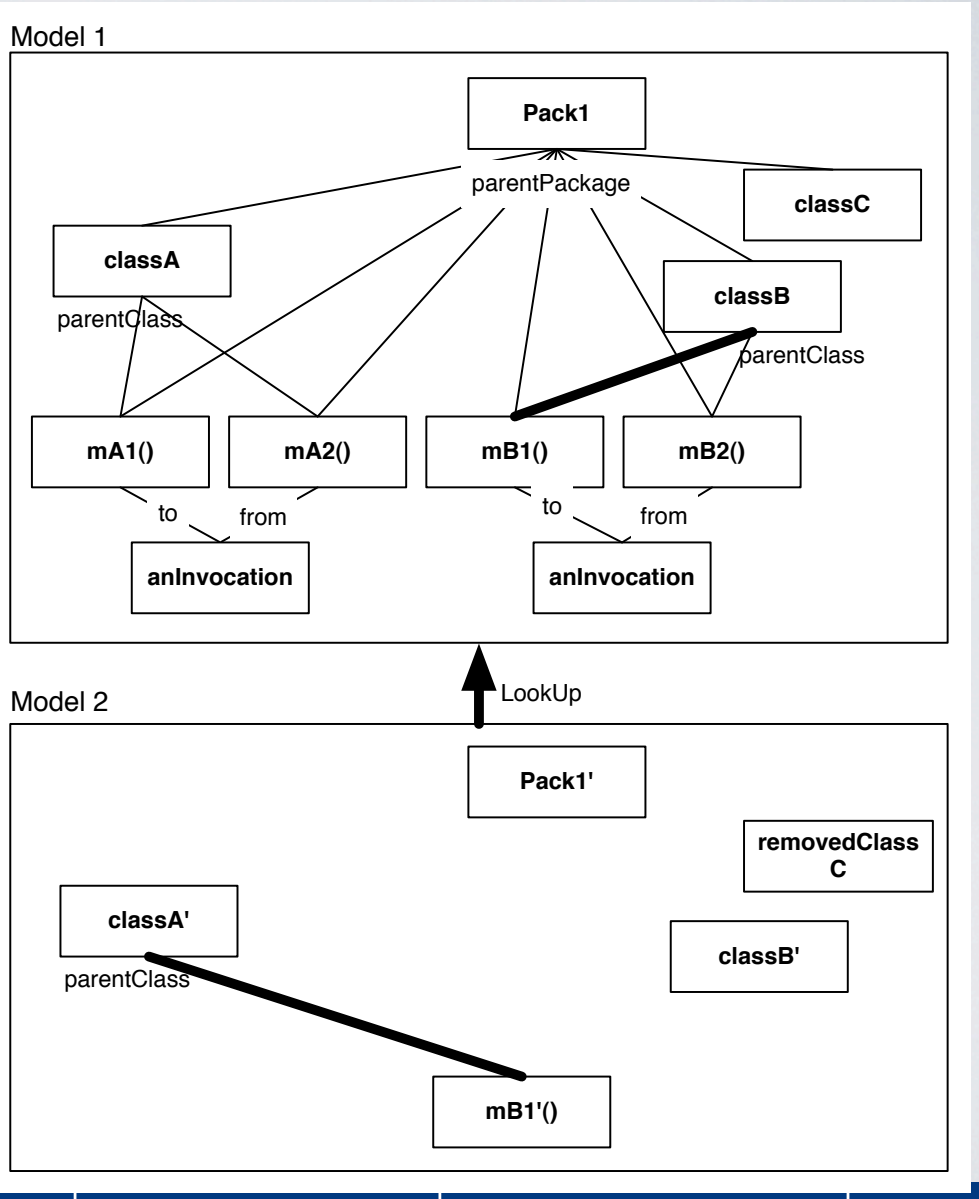

Model 1
Pack1
parentPackage
classC
classA
classB
parentClass
parentClass
mA1()
mA2()
mB1()
mB2()
to
from
to
from
anInvocation
anInvocation
LookUp
Model 2
Pack1'
removedClass
C
classA'
classB'
parentClass
mB1'()

LOOKUP

- Pros
  - only changed entities are copied
- Cons
  - time for lookUp
  - what about removed elements ?

LOOKUP

- a problem:
  - mB2() parentClass ?

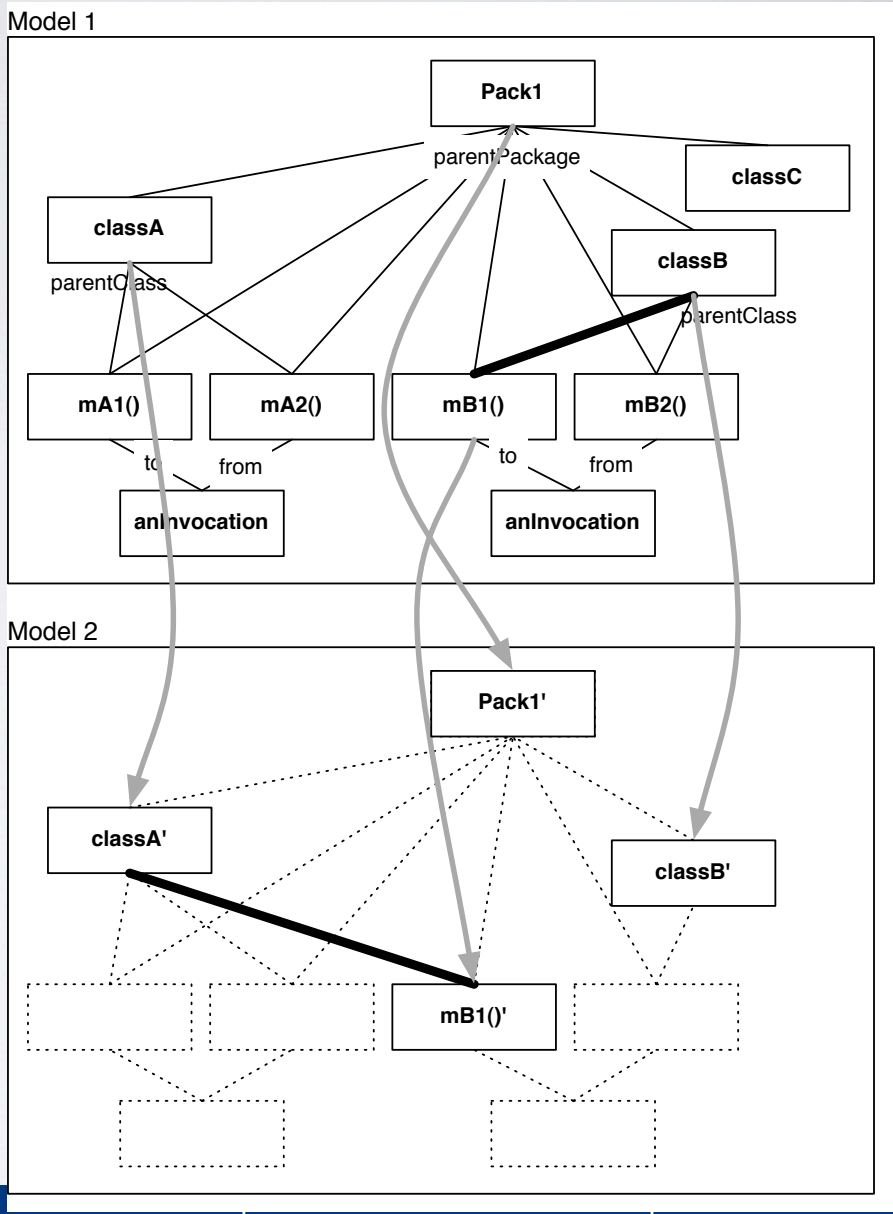


PERFORMANCES

Approach	cost of creation	cost of access	memory cost
copy	++++	I	++++
delta	+	++++	+
lookUp	+	++++	+

ORION APPROACH

- pointers approach
- access to updated elements
  - by an ID
  - mB2() parentClass ?



ORION APPROACH

- Pros
  - fast creation and low memory usage: only changed elements
  - fast access: with the ID or pointers
- Cons
  - a model depends on its predecessors

PERFORMANCES

Approach	cost of creation	cost of access	memory cost
copy	++++	I	++++
delta	+	++++	+
lookUp	+	++++	+
Orion	+	+	+

ITS FUTURE

- Use it to make manual and automatic prediction changes
- Comparing different changes
- Coupling with analysis tools (Famix)

THANKS

SUPPORTING INCREMENTAL CHANGES IN LARGE MODELS

Jannik Laval, Simon Denier, Stéphane Ducasse
RMod Team - INRIA

Andy Kellens
Software Languages Lab - VUB