



# *Traceability@Runtime*

Cyril Faucher,  
with the help of François Tanguy  
and the idea of Didier Vojtisek and Franck Fleurey

IRISA Lab / INRIA Rennes, France  
Triskell Group

Kermeta Day - April 2nd, 2009



# Outline

- Motivation
- Need traceability in model transformation
- Implementation of the Traceability pattern
- Factorization of the Traceability pattern
- Conclusion



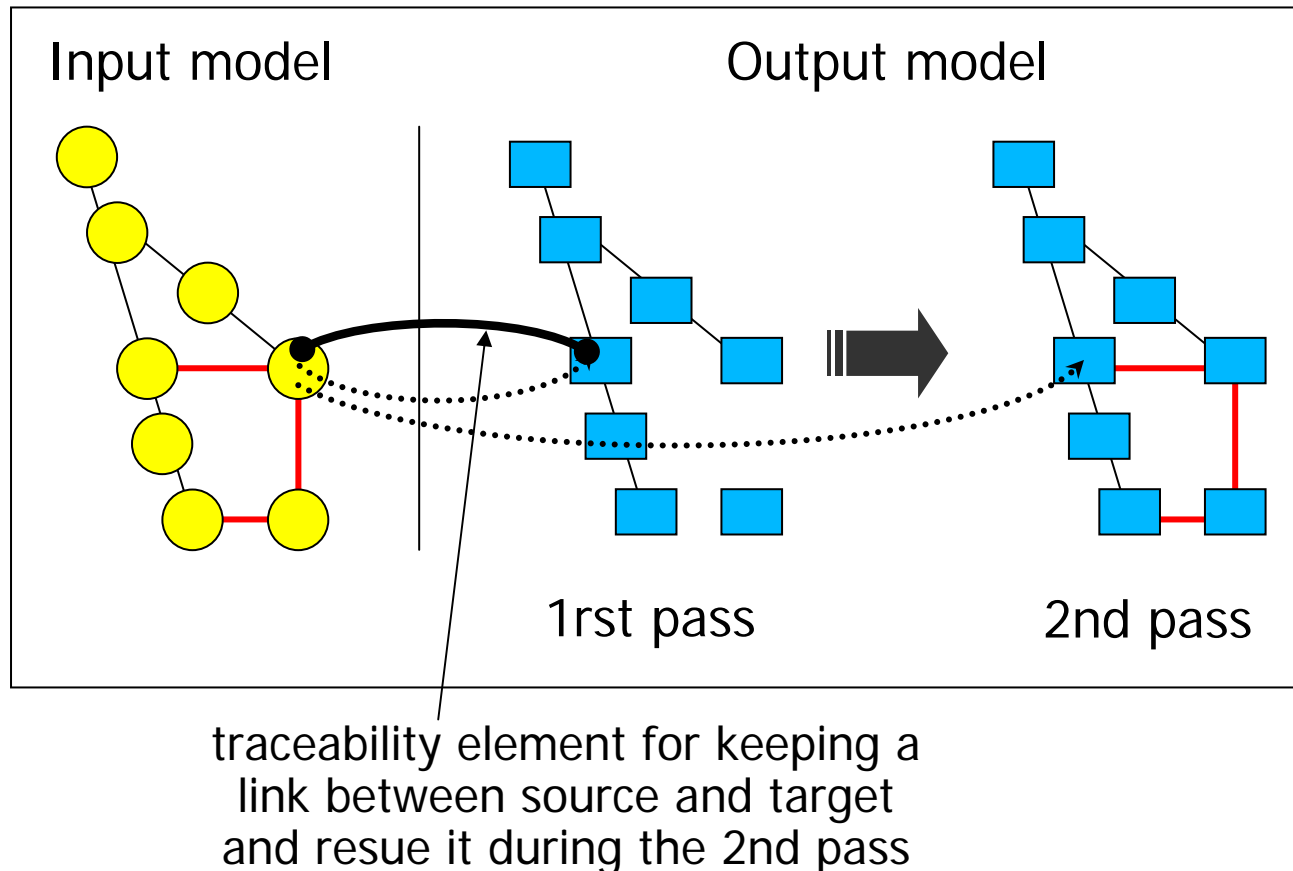
# Motivation

- Traceability is a key point in a model transformation impl.
- Traceability@Runtime
  - that means traceability in memory
  - a traceability pattern is weaved into a metamodel
- How to add traceability with the Kermeta Aspect ability
- How to factorize the traceability pattern



# Need traceability in Model Transformation

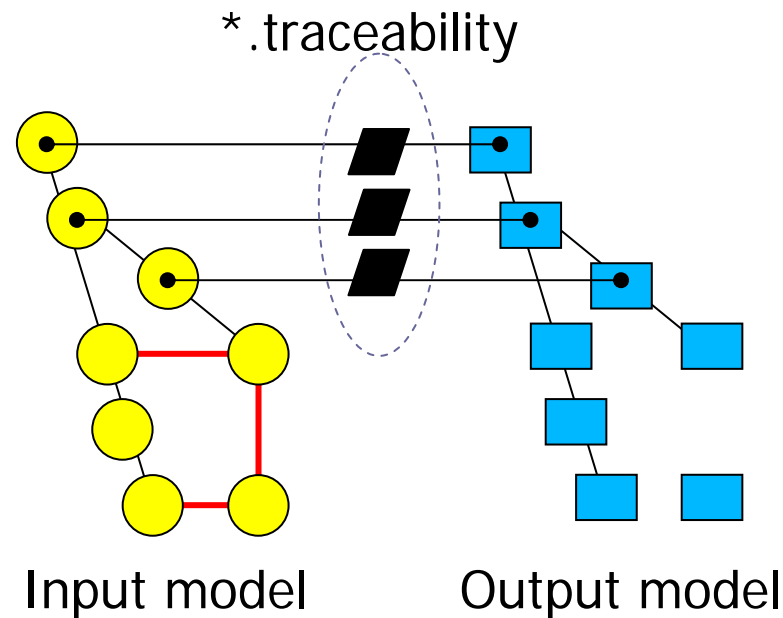
- Model transformation of graphs, often in 2 passes
  - 1<sup>st</sup>: model elements creation
  - 2<sup>nd</sup>: links creation





# Implementation of Traceability

- Traceability implementation in a classical way
  - in Java: `Hashtable<Object, Object>`
  - in Kermeta: `Hashtable<Object, Object>`
- With a specific model (*cf Traceability MDK*)





# Traceability@Runtime implementation

- With Aspect ability
  - Add/weave a property “trace” in the ClassDefinition of the input metamodel

from the input metamodel

```
aspect class MyClassMM1 {  
    reference trace : MyClassMM2  
}
```

from the output metamodel

- Retrieve the trace:  
instance\_myClassMM1.trace //returns a MyClassMM2



# Factorization of the Traceability@Runtime pattern

- Combination of Aspect + Inheritance + Genericity, example from the Kermeta compiler

```
package kermeta::language::structure;

require kermeta
require "http://www.eclipse.org/emf/2002/Ecore"

using kermeta::language::structure
using ecore

class Traceability<X> {
    reference ecoreModelElement : X
}

aspect class ClassDefinition inherits Traceability<EClass> {
    reference ecoreDataType : EDataType[0..1]
    reference subClassDefinitions : ClassDefinition [0..*]
}
```



# Conclusion

- + none search phase, the trace is directly accessible as a Property, trace is typed
- - all the traces are in memory, not easy to free the memory





# QUESTION ?