	ATL Transformation Catalogue of Model Transformations	Author Baudry Julien Eric Simon Jul.baudry <at> gmail.com Simon.eric2 <at> gmail.com
	Documentation	Aug 7th 2006

1.	THE CATALOGUE OF MODELS TRANSFORMATIONS	1
1.1.	REFINEMENTS	1
1.2.	QUALITY IMPROVEMENT TRANSFORMATIONS.....	1
2.	METAMODELS.....	2
3.	REFERENCES	3

1. The catalogue of models transformations

All transformations have been extracted from the catalogue of models transformations (cf. Reference). This document presents a set of model transformations on UML class and state machine models. We have only implemented the model transformations on UML class. These transformations can be classified on two categories:

1.1. Refinements

These transformations are used to refine a model towards an implementation. For example, PIM to PSM transformations in the Model-driven Architecture. They typically remove certain constructs or structures, such as multiple inheritances, from a model, and represent them instead by constructs, which are available in the implementation platform.


- Replacing inheritance by association
- Removal of many-many associations
- Removal of association classes
- Weakening preconditions or strengthening postconditions
- Replace association by attribute
- Introduce primary key
- Replace association by foreign key

1.2. Quality improvement transformations

The transformations in this category aim to improve the structure of a model, making it conform more closely to normal uses of UML notation, or improving its precision or its flexibility for extension and adaptation.

Two general categories of quality improvement transformation are the removal of redundancy and the factoring out/decomposition of elements.

The 'introduce superclass' transformation is one example of the removal of redundancies from a model. Normally redundancies should be eliminated – they complicate a model unnecessarily and, if implemented, lead to extra work and possibilities for a program flaws. Other examples of such transformations are the removal of redundant inheritances and associations.

	ATL Transformation Catalogue of Model Transformations	Author Baudry Julien Eric Simon Jul.baudry <at> gmail.com Simon.eric2 <at> gmail.com
	Documentation	Aug 7th 2006

- Remove redundant association
- Remove redundant inheritance
- Remove redundant classes
- Merge partial classes
- Introduce interface(s) for supplier class(es)
- Disaggregation
- Raise Supplier abstraction level
- Make partial role total (a)
- Make partial role total (b)

2. Metamodels

These transformations are models transformations, that means the input metamodel and the output metamodel are the same. First of all, we use the metamodel KM3 as much as possible. But some transformations need concepts that KM3 cannot represents, for example association class. These ones are:

- Weakening preconditions or strengthening postconditions
- Replace association by attribute
- Introduce interface (s) for supplier class (es)
- Remove redundant classes
- Removal of association classes

We needed a more complete metamodel. We use the UML2 metamodel available on the UML2 website. Here is a short description of this project:


“The UML2 project (an Eclipse Tools sub-project) is an EMF-based implementation of the UML™ 2.x metamodel for the Eclipse platform. The objectives of this project are to provide a useable implementation of the metamodel to support the development of modelling tools, a common XMI schema to facilitate interchange of semantic models, test cases as a means of validating the specification, and validation rules as a means of defining and enforcing levels of compliance.”

For each transformation, we create 2 different transformations: with context and without context. The transformation “without context” means the transformation only handle the minimum element needed by the transformation. The transformation “with context” deals with all elements of the metamodel (but only with the KM3 metamodel, because the UML2 metamodel is too huge). All the transformations have ANT script to perform injection, transformation and extraction.

For these transformations, we use these different tools

- Eclipse 3.1,
- ATL Engine 1.0.7 with ATLtoASMCompiler 1.10
- (org.atl.eclipse.engine\src\org\atl\ eclipse\ engine\resources\ATLtoASMCompiler.asm)
- EMF 2.1.0,
- UML 1.1.1 (2.0 version need Eclipse 3.2, but not compatible with current ATL version).

Be careful, that means theses transformation do not work with the bundle (2006/05/01).

	<p style="text-align: center;">ATL Transformation</p> <p style="text-align: center;">Catalogue of Model Transformations</p>	<p style="text-align: center;">Author</p> <p style="text-align: center;">Baudry Julien Eric Simon</p> <p style="text-align: center;">Jul.baudry <at> gmail.com Simon.eric2 <at> gmail.com</p>
	<p style="text-align: center;">Documentation</p>	<p style="text-align: center;">Aug 7th 2006</p>

3. References

- [1] Catalogue of Model Transformations
<http://www.dcs.kcl.ac.uk/staff/kcl/tcat.pdf>
- [2] UML2 Project
<http://www.eclipse.org/uml2/>