Introduction

Uncertainty

Uncertain transformation

Conclusions and future work

Roadmap

• Introduction
• Uncertainty
  • In user models
  • In generated models
• Uncertain transformation
  • An overview on JTL
  • Specifying uncertain transformations in JTL
  • Generating concrete alternatives
  • Factorizing alternatives
• Conclusions and future work
In Model-Driven Engineering (MDE), the potential advantages of using bidirectional transformations are largely recognized.

Despite its relevance, bidirectional languages have rarely produced anticipated benefits as demonstrated by the lack of a leading language comparable, for instance, to ATL.
Introduction

The problem of non-determinism in bidirectional transformations

non-injective
Introduction

The problem of non-determinism in bidirectional transformations

- non-injective
- one-to-many mapping
- multiple solutions
Introduction

Transformations may be non-bijective: given a source model, there could be more than one target model which correctly related to the source.

Transformations may be not total: only the relevant concepts of the source models are mapped toward the corresponding target elements.

The solutions for this specific problem can be given at

– transformation-centric: the transformation specify how to resolve the ambiguity;

– language-centric: the language is endowed with appropriate semantics to allow transformations to yield all the valid solutions.
 Uncertainty

Uncertainty

Not So Sure
POSS-IBLY?
Maybe
**Uncertainty** means that «rather than having a single model, we actually have a set of possible models, and we are not sure which is the correct one».

**Uncertainty** occurs when

- The designer does not have the complete, consistent and accurate information required to make a decision during software development.
- In model transformations

In both cases, design decisions are delayed requiring the manual intervention of the modeler.
Designers are often unsure about the information to specify in models because of uncertainty and incompleteness of requirements.
Uncertainty in generated models could arise when reversing a non-injective transformation.

It is non unfrequent that the implementor is (partly or completely) unaware that details are missing at design-time and this will be manifest only when the transformation will be executed.

In reference to the previous example:

Left Hand Side Hierarchical State Machine
Uncertainty in Generated Models

Right Hand Side State Machine

Modified Right Hand Side State Machine
Uncertainty in Generated Models

Propagating changes in Left Hand Side Hierarchical State Machine: more than one model occurs (multiplicity of solutions)
JTL is a constraint-based model transformation language specifically tailored to support bidirectionality. It adopts a QVT-R like syntax and allows a declarative specification of relationship between MOF models.

The semantics is given in terms of Answer Set Programming (ASP), which is a form of declarative programming oriented towards difficult (primarily NP-hard) search problems and based on the stable model (answer set) semantics of logic programming.
Uncertain Transformation
Specifying uncertainty transformation

Non injective
Different source and same target
Elements

Uncertain Transformation
Generating concrete alternatives

Target HSM (a)
**Uncertain Transformation**

Generating concrete alternatives

Target HSM (b)

Target HSM (c)
uncertain transformation
Generating concrete alternatives

Target HSM (d)

uncertain transformation
Generating concrete alternatives

Multiple Solutions in HSM
Uncertainty in Bidirectional Transformation

Factorizing alternatives

Table:

<table>
<thead>
<tr>
<th>Property</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Effect</td>
<td></td>
</tr>
<tr>
<td>Source</td>
<td>Composite State Active</td>
</tr>
<tr>
<td>Target</td>
<td>State Pointing</td>
</tr>
<tr>
<td>Trigger</td>
<td>print</td>
</tr>
<tr>
<td>Property</td>
<td></td>
</tr>
<tr>
<td>Effect</td>
<td></td>
</tr>
<tr>
<td>Source</td>
<td>State Authentication</td>
</tr>
<tr>
<td>Target</td>
<td>State Pointing</td>
</tr>
<tr>
<td>Trigger</td>
<td>print</td>
</tr>
<tr>
<td>Property</td>
<td></td>
</tr>
<tr>
<td>Effect</td>
<td></td>
</tr>
<tr>
<td>Source</td>
<td>State Selecting transaction</td>
</tr>
<tr>
<td>Target</td>
<td>State Pointing</td>
</tr>
<tr>
<td>Trigger</td>
<td>print</td>
</tr>
<tr>
<td>Property</td>
<td></td>
</tr>
<tr>
<td>Effect</td>
<td></td>
</tr>
<tr>
<td>Source</td>
<td>State Transaction</td>
</tr>
<tr>
<td>Target</td>
<td>State Pointing</td>
</tr>
<tr>
<td>Trigger</td>
<td>print</td>
</tr>
</tbody>
</table>

Conclusions and Future Work
Conclusions

We discussed the problem of non-determinism in bidirectional transformations. In many respects, the multiplicity of solutions in non-bijective transformations is an aspect which has been largely neglected.

Our proposal is based on JTL, a language which is able to generate more than one model at once, whose transformation engine has been modified to accommodate uncertainty.

References

- JTL http://jtl.di.univaq.it
- Romina Eramo, Alfonso Pierantonio, Gianni Rosa Uncertainty in Bidirectional Transformation, in: 36th International Conference on Software Engineering (MISE), Hyderabad, India
Dipartimento di Ingegneria e Scienze dell’Informazione e Matematica

Università degli Studi dell’Aquila

UNCERTAINTY IN BIDIRECTIONAL TRANSFORMATION