Uncertainty in Bidirectional Transformation @SATToSE2014

Dipartimento di Ingegneria e Scienze dell'Informazione e Matematica

Università degli Studi dell'Aquila

Gianni Rosa

PHD Student @ Computer Science Department,
University of L'Aquila, Italy

Join work with Romina Eramo, Alfonso Pierantonio

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

INTRODUCTION

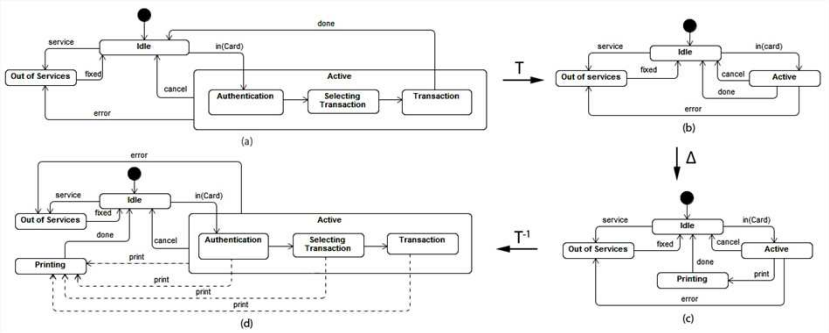
Introduction

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

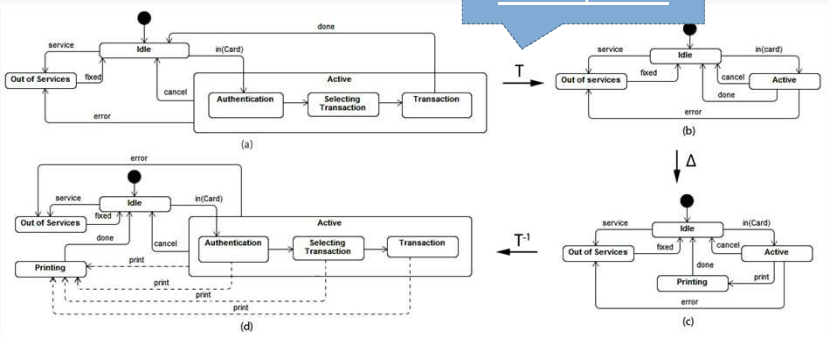
The problem of non-determinism in bidirectional transformations



Introduction

The problem of non-determinism in bidirectional transformations

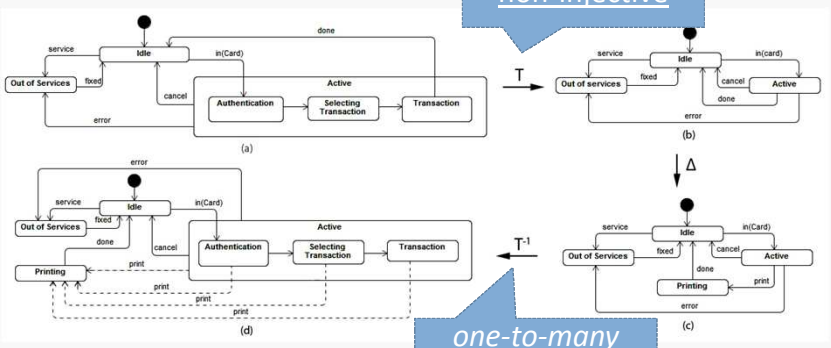
non-injective



Introduction

The problem of non-determinism in bidirectional transformations

non-injective

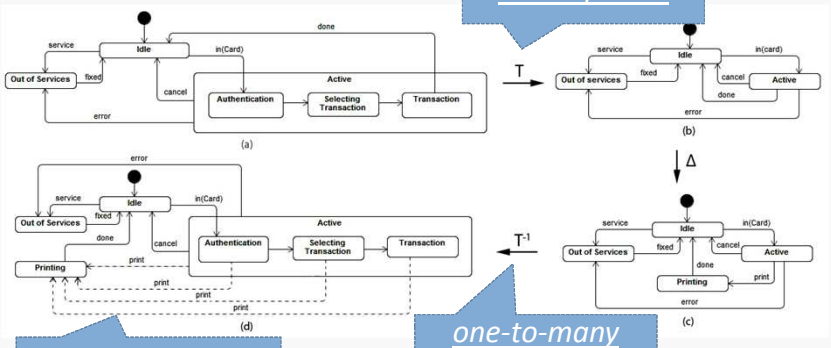


one-to-many mapping

Introduction

The problem of non-determinism in bidirectional transformations

non-injective



multiple solutions

one-to-many mapping

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

Introduction

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

Uncertainty

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

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

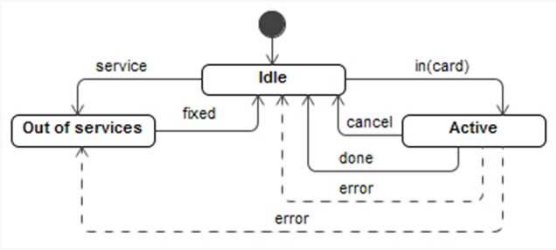
In both cases, design decisions are delayed requiring the manual intervention of the modeler.

Uncertainty In User Models

Designers are often unsure about the information to specify in models because of uncertainty and incompleteness of requirements.

Uncertainty In User Models

Designers are often unsure about the information to specify in models because of uncertainty and incompleteness of requirements.



UNCERTAINTY IN BIDIRECTIONAL TRANSFORMATION

Dipartimento di Ingegneria e Scienze dell'Informazione e Matematica
Università degli Studi dell'Aquila

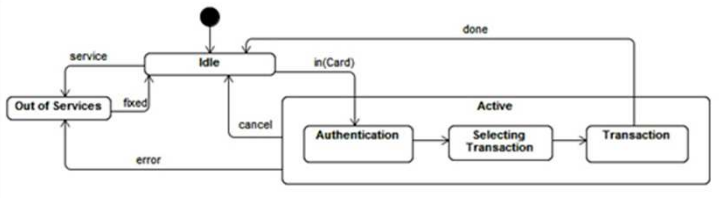
Uncertainty In Generated Models

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.

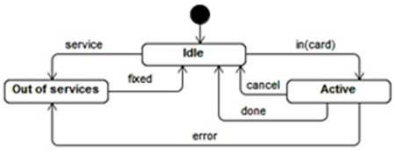
Uncertainty In Generated Models

In reference to the previous example:



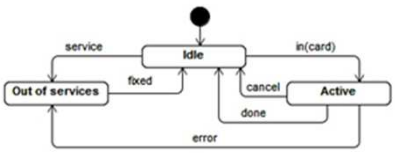
Left Hand Side Hierarchical State Machine

Uncertainty In Generated Models

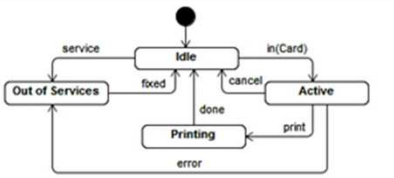


Right Hand Side State Machine

Uncertainty In Generated Models

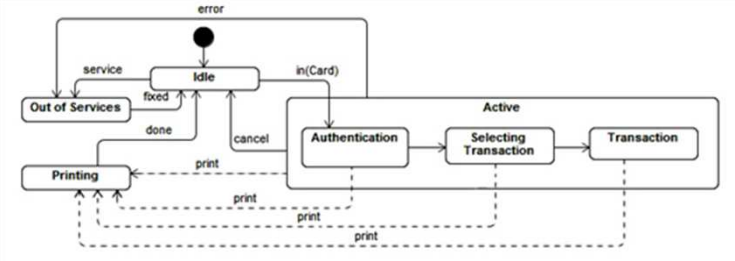


Right Hand Side State Machine



Modified Right Hand Side State Machine

Uncertainty In Generated Models



Propagating changes in Left Hand Side Hierarchical State Machine: more of then one model occurs (multiplicity of solutions)

UNCERTAIN TRANSFORMATIONS

Uncertain Transformation
JTL

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 uncertanty transformation

```
1transformation hsm2sm(source : HSM, target : SM) {
2 ...
3top relation Transition2Transition {
4  enforce domain source sourceTrans: HSM::Transition{
5   owningStateMachine = sourceSM: HSM::StateMachine { },
6  };
7  enforce domain target targetTrans: SM::Transition{
8   owningStateMachine = targetSM: SM::StateMachine { },
9  };
10 when {...}
11 where {...}
12 }
13 relation TransitionSource2TransitionSource {
14  enforce domain source sourceTrans: HSM::Transition {
15   source = sourceState : HSM::State { }
16  };
17  enforce domain target targetTrans: SM::Transition {
18   source = targetState : SM::State { }
19  };
20  when {
21   State2State(sourceState, targetState) and
22   sourceState.owningCompositeState.oclIsUndefined();
23  }
24 }
25 relation TransitionSourceComposite2TransitionSource {
26  enforce domain source sourceTrans: HSM::Transition {
27   source = sourceState : HSM::CompositeState { }
28  };
29  enforce domain target targetTrans: SM::Transition {
30   source = targetState : SM::State { }
31  };
32  when {
33   CompositeState2State(sourceState, targetState);
34  }
35 } ...
```

A fragment of the HSM2SM transformation in JTL

Uncertain Transformation

Specifying uncertanty transformation

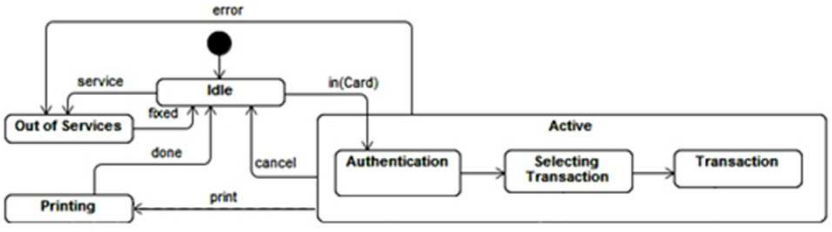
```
13 relation TransitionSource2TransitionSource {
14  enforce domain source sourceTrans: HSM::Transition {
15   source - sourceState : HSM::State { }
16  };
17  enforce domain target targetTrans: SM::Transition {
18   source - targetState : SM::State { }
19  };
20  when {
21   State2State(sourceState, targetState) and
22   sourceState.owningCompositeState.oclIsUndefined();
23  }
24 }
25 relation TransitionSourceComposite2TransitionSource {
26  enforce domain source sourceTrans: HSM::Transition {
27   source - sourceState : HSM::CompositeState { }
28  };
29  enforce domain target targetTrans: SM::Transition {
30   source - targetState : SM::State { }
31  };
```

Non injective

Different source and same target Elements

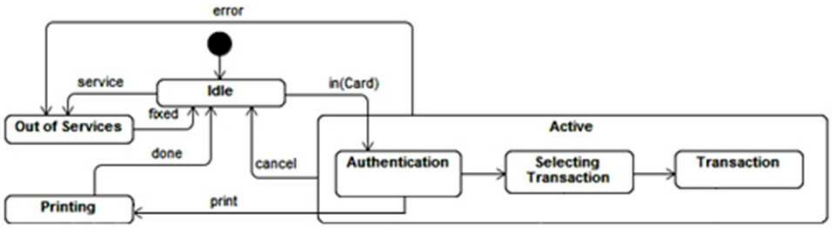
Uncertain Transformation

Generating concrete alternatives



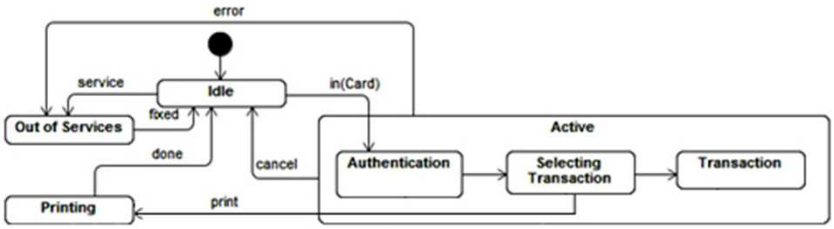
Target HSM (a)

Uncertain Transformation
Generating concrete alternatives



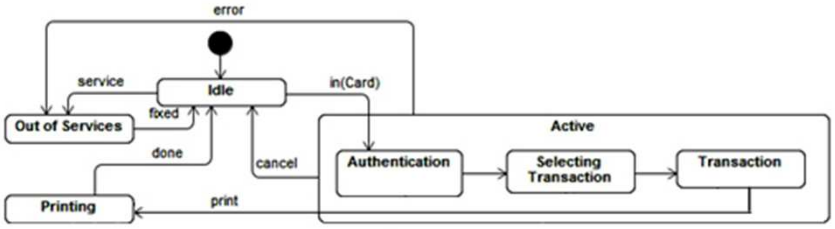
Target HSM (b)

Uncertain Transformation
Generating concrete alternatives



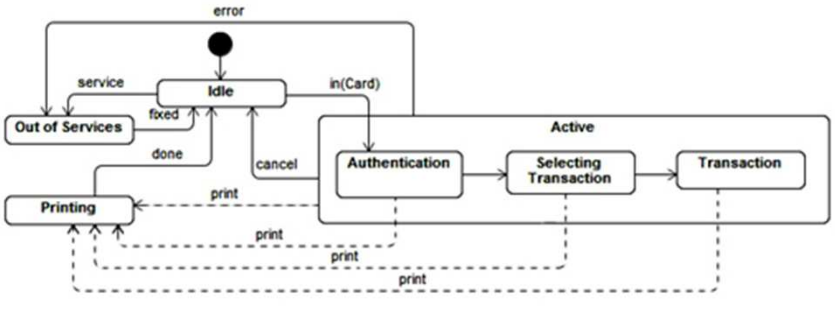
Target HSM (c)

Uncertain Transformation
Generating concrete alternatives



Target HSM (d)

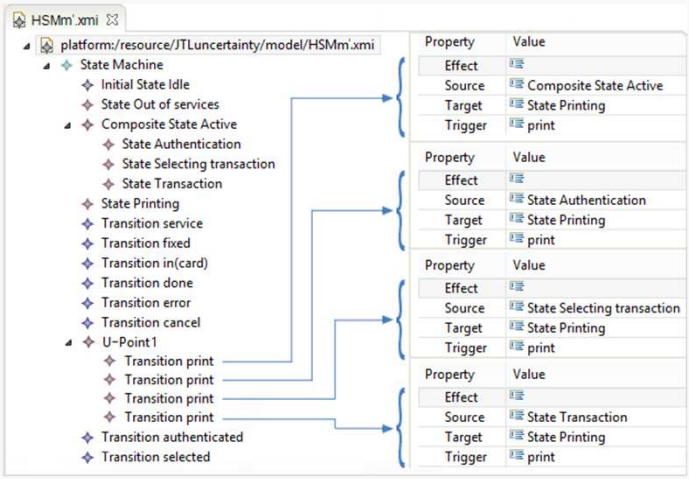
Uncertain Transformation
Generating concrete alternatives



Multiple Solutions in HSM

Uncertain Transformation

Factorizing alternatives



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
- G. Callow and R. Kalawsky. A Satisficing Bi-Directional Model Transformation Engine using Mixed Integer Linear Programming. Journal of Object Technology, 12(1):1: 1–43, 2013.
- A. Cicchetti, D. Di Ruscio, R. Eramo, and A. Pierantonio. JTL: a bidirectional and change propagating transformation language. In Procs. of SLE 2010, LNCS 6563, pages 183–202. Springer, 2011.
- K. Czarnecki, J. N. Foster, Z. Hu, R. Lämmel, A. Schürr, and J. F. Terwilliger. Bidirectional Transformations: A Cross-Discipline Perspective - GRACE meeting notes, state of the art, and outlook. In Procs. of ICMT2009, volume 5563 of LNCS, pages 260–283. Springer, 2009.
- M. Famelis, R. Salay, and M. Chechik. Partial models: Towards modeling and reasoning with uncertainty. In ICSE, pages 573–583, 2012.
- M. Famelis, R. Salay, A. D. Sandro, and M. Chechik. Transformation of models containing uncertainty. In MoDELS, pages 673–689, 2013.
- R. Salay, M. Chechik, J. Horkoff, and A. D. Sandro. Managing requirements uncertainty with partial models. Requir. Eng., 18(2):107–128, 2013.

Dipartimento di Ingegneria e Scienze dell'Informazione e Matematica

Università degli Studi dell'Aquila