A Compositional Approach to Bidirectional Model Transformation

ICSE’09 New Ideas and Emerging Results

Soichiro HIDAKA   Zhenjiang HU   Hiroyuki KATO
National Institute of Informatics, Japan

Keisuke NAKANO
The University of Electro-Communications, Japan

Introduction

In bidirectional model transformation, modifications propagate from source models to target models as well as from target to source. Although bidirectional model transformation plays an important role in model-driven software development, lack of clear semantics of composition is one of open problems.

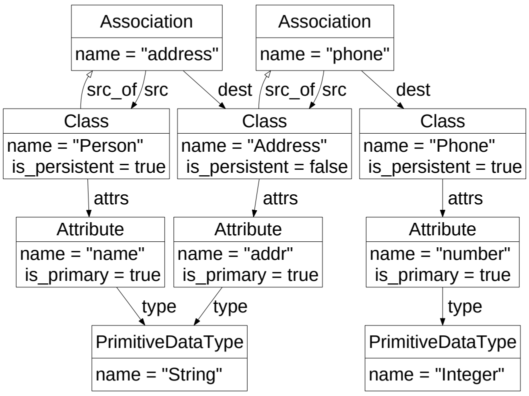
Proposed Approach and Results

Compositional graph transformation language UnQL is extended for bidirectional model transformation by

- Editing primitives (replace, delete, extend) [1]
- Bidirectional interpretation of each graph constructors and combinators [2]

Models as Edge-labeled Graphs

Models are internally represented by edge-labeled graphs.



Association
name = "address"

Association
name = "phone"

src_of / src   dest   src_of / src   dest

Class
name = "Person"
is_persistent = true

Class
name = "Address"
is_persistent = false

Class
name = "Phone"
is_persistent = true

attrs   attrs   attrs

Attribute
name = "name"
is_primary = true

Attribute
name = "addr"
is_primary = true

Attribute
name = "number"
is_primary = true

type   type   type

PrimitiveDataType
name = "String"

PrimitiveDataType
name = "Integer"

Class diagram: an example of models to be transformed

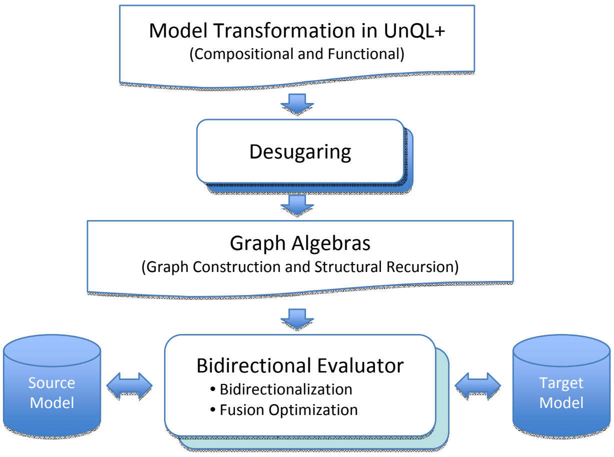
Model Transformations in UnQL+

Transformation to prefix every name of the class by “class_” can be expressed in UnQL+ as

replace {$*name* : {}} by {("*class_*"^$*name*):{}} where
{_*.*Class.name.String*:{$*name*:{}}} in $*classDB*

Bidirectional Evaluator and its property

Every UnQL+ program is translated[1] into UnCAL in which fixed number of constructors and combinators are combined to form a bigger transformation.



Model Transformation in UnQL+
(Compositional and Functional)

Desugaring

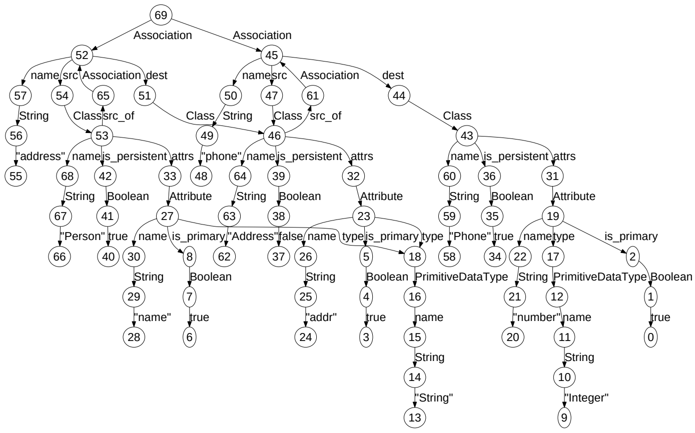
Graph Algebras
(Graph Construction and Structural Recursion)

Source Model

Bidirectional Evaluator
- Bidirectionalization
- Fusion Optimization

Target Model

A compositional Framework for Bidirectional Model Transformation



Internal representation of the class diagram that is transformed by our system

$E ::= \{\} \mid \{L : E\} \mid E \cup E$	(* tree constructors *)
$\mid \&x := E \mid \&y \mid () \mid E \oplus E \mid E \mathbin{@} E$	(* graph constructors *)
$\mid cycle(E)$	(* graph with cycles *)
$\mid Var$	(* variables *)
$\mid \text{let } Var = E \text{ in } E$	(* sequential composition *)
$\mid \text{if } B \text{ then } E \text{ else } E$	(* conditional *)
$\mid rec(\lambda(LabelVar, Var).E)(E)$	(* structural recursion *)

Syntax of UnCAL graph algebra

Formal semantics: union (∪) example

Two transformations are executed componentwise and combined.

$$\frac{\rho \xrightarrow{e_1} g_1 \quad \rho \xrightarrow{e_2} g_2 \quad g_1 \cup g_2 \Rightarrow g}{\rho \xrightarrow{e_1 \cup e_2} g}\,(\mathrm{FWD})$$

Modified target (g’) are decomposed and the resultant components are fed to backward transformation.

$$\frac{g' \Rightarrow_\rho (g'_1, g'_2) \quad \rho'_1 \xleftarrow[\rho]{e_1} g'_1 \quad \rho'_2 \xleftarrow[\rho]{e_2} g'_2}{\rho'_1 \uplus_\rho \rho'_2 \xleftarrow[\rho]{e_1 \cup e_2} g'}\,(\mathrm{BWD})$$

Well behavedness

No change on the target g should give no change on the source (environment) ρ.

$$\frac{\rho \xrightarrow{e} g}{\rho \xleftarrow[\rho]{e} g}\;[\mathrm{GetPut}]$$

Another forward transformation from the modified source ρ’ produces g’ again.

$$\frac{\rho' \xleftarrow[\rho]{e} g'}{\rho' \xrightarrow{e} g'}\;[\mathrm{PutGet}]$$

Impact and Future Work

- Demonstrate that functional approach is helpful to give bidirectional semantics in a formal and concise way
  - Demonstration available at http://www.biglab.org/
- Compare/combine with rule based approach

[1] S. Hidaka, Z. Hu, H. Kato, K. Nakano, Towards Compositional Approach to Model Transformation for Software Development, SAC 2009: 468-475, Mar. 2009.
[2] S. Hidaka, Z. Hu, H. Kato, and K. Nakano. An Algebraic approach to bidirectional model transformations. Technical Report GRACE-TR08-02, GRACE Center, National Institute of Informatics, Sept. 2008.

Soichiro HIDAKA   **Email : hidaka@nii.ac.jp**   **Project URL: http://www.biglab.org/**