A Manifesto for Model Merging

Brunet, G., Chechik, M., Easterbrook, S., Nejati, S., Niu, N., Sabetzadeh, M.

Presented by: Lissa Dora
October 10, 2012
Introduction

• **Problem:** Difficulties in comparing model merging approaches (ex: consistent vs inconsistent, homogeneous vs heterogeneous, etc).
  – Various and inconsistent use of vocabularies
  – Different assumptions about model types and its relationship

• **Solution:** A framework for comparing different approaches.
  – By treating merge as *an algebraic operator over models and model relationships*. 
• Global Model Management Operators
• Algebraic Properties
  – Some of the properties we might expect of the model management operators.
• Implementation of the global model management framework in 2 different domains:
  1. Structural model, represented as Entity Relationship Diagram (ERD)
     • Open world semantics is assumed.
  2. Behavioural model, represented as state machines
     • Closed world semantics is typically used.
• 9 Basic Operators for Global Model Management

1. merge  
2. match  
3. diff   
4. split  
5. slice 
6. check-property 
7. is-consistent 
8. patch  
9. propagate

Model Checking  
Applies Transformation
Merge, Match, and Split

- **merge**: model x model x relationship → model
- **match**: model x model → relationship
  - There may be more than just one possible relationship between the models.
- **split**: model → model x model x relationship
  - Produces a *partition* (compatible models)
  - Is the *inverse* of merge.
• **diff** : model x model → transformation
  – Diff is *not* commutative. Its result, “transformation”, will be used by **patch**.

• **slice** : model x criterion → model
• Both operator is used for model checking before merging.

• **check-property** : model x property $\rightarrow$ truth-value
  – Check model based on certain behavioural and/or structural properties.

• **is-consistent** : model x model x relationship $\rightarrow$ truth-value
  – Check models’ consistency based on models relationship.
Patch and Propagate

- **patch**: model × transformation → model

- **propagate**: transformation × model × model × relationship → model
Outline

- Global Model Management Operators
- Algebraic Properties
  - Some of the properties we might expect of the model management operators.
- Implementation of the global model management framework in 2 different domains:
  1. Structural model, represented as Entity Relationship Diagram (ERD)
     - Open world semantics is assumed.
  2. Behavioural model, represented as state machines
     - Closed world semantics is typically used.
Algebraic Properties

1) **Idempotency:**

\[ \text{merge}(m_1, m_1, \text{match}(m_1, m_1)) = m_1 \]

**relationship** \( r \)

2) **Commutativity:**

\[ \text{merge}(m_1, m_2, r) = \text{merge}(m_2, m_1, r) \]

3) **Associativity:**

\[ \text{merge}(\text{merge}(m_1, m_2, r_{1,2}), m_3, r_{(1,2),3}) = \text{merge}(m_1, \text{merge}(m_2, m_3, r_{2,3}), r_{1,(2,3)}) \]

\[ m_1 + m_2 \]

\[ m_2 + m_3 \]
4) **Inverses:**

\[
\text{split}(\text{merge}(m_1, m_2, r)) = (m_1, m_2, r)
\]

\[
\text{merge} = \sim \text{split}
\]

5) **Monotonicity:**

\[
m_1 \preceq m_1' \land m_2 \preceq m_2' \Rightarrow
\]

\[
\text{merge}(m_1, m_2, r) \preceq \text{merge}(m_1', m_2', r')
\]

refinement and trace equivalence

6) **Totality:**

\[
\forall m_1, m_2 \in \text{model} \cdot \text{merge}(m_1, m_2, r) \in \text{model}
\]

The merge operation is well-defined for any pair of models, whether or not they satisfy consistency. It is preferable to repair the inconsistency first.
Outline

- Global Model Management Operators
- Algebraic Properties
  - Some of the properties we might expect of the model management operators.
- Implementation of the global model management framework in 2 different domains:
  1. Structural model, represented as Entity Relationship Diagram (ERD)
     - Open world semantics is assumed.
  2. Behavioural model, represented as state machines
     - Closed world semantics is typically used.
Merging Entity-Relationship Models

- Relationships:
  \[ f : C \to A \quad \rightarrow \quad \text{C1-To-Rob} \]
  \[ g : C \to B \quad \rightarrow \quad \text{C1-To-Sue} \]
Merged ERD Result

- Truth values are drawn from set \{\checkmark, !, \times\}
  \checkmark = conclusively appropriate
  ! = proposed
  \times = conclusively inappropriate

- In this example ! is a default annotation, and there is no \times

(b) The merged model
Merging State Machines

Relationships between $CM_1$ and $CM_2$:

$$\{(s_0, t_0), (s_1, t_1), (s_2, t_2), (s_2, t_3)\}$$
Merged State Machine Result

- Truth values are drawn from set \{t, !, f\}
  - \(t\) = explicitly included
  - \(!\) = unknown
  - \(f\) = explicitly excluded
• The nine basic model operators provides a standard vocabulary for discussing the key ideas in model merging.

• Each operator can interact with other operators.
  – Example: **match** operator generate “relationship” which is used in **merge** operator.

• Relationship plays a central role in model merging.

• The examples provided are homogeneous merges, yet the algebraic approach is believed to apply to heterogeneous merges too.
  – **diff**, **slice**, **patch**, **propagate**