MARS: Metamodel Recovery from Multi-Tiered Models Using Grammar Inference

Qichao Liu, Faizan Javed, Marjan Mernik, Barrett R. Bryant, Jeff Gray, Alan Sprague and Dejan Hrnčič

University of Alabama at Birmingham, Regions Financial Corp., University of Maribor

{qichao, bryant, gray, sprague}@cis.uab.edu, faizan.javed@regions.com, {marjan.mernik, dejan.hrncic}@uni-mb.si

http://www.cis.uab.edu/softcom/GrammarInference/

1. Background

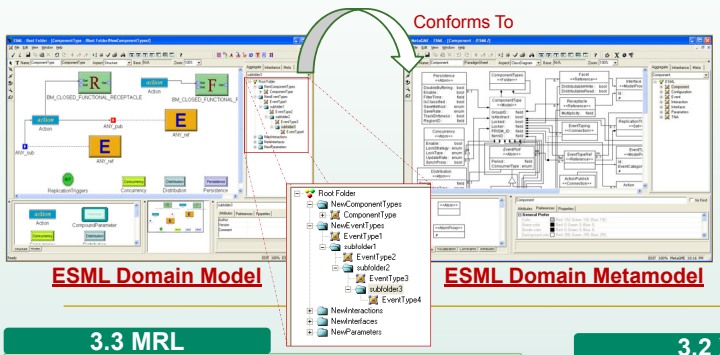
❖A **Model** is an abstraction of phenomena in the real world.

❖A **Metamodel** represents a schema that defines the syntax of a model like a grammar defines a programming language. A model conforms to its metamodel.

❖**GME (Generic Modeling Environment)** is a modeling tool that allows users to define a domain-specific visual modeling language. http://www.isis.vanderbilt.edu/Projects/gme/

❖A **Multi-Tiered Domain** represents large models and enables users to capture multiple viewpoints of the system.

❖**ESML (Embedded System Modeling Language)** is a multi-tiered domain with 7 different viewpoints. In its model instances each viewpoint is established as a separate folder (a model organization concept in GME). Each folder may have subfolders to form a hierarchy.

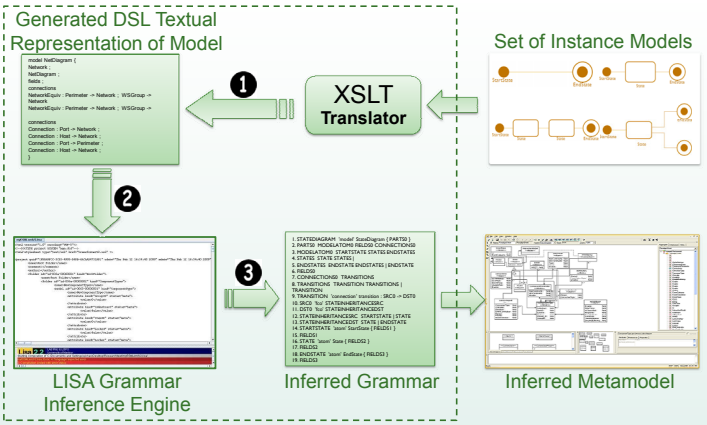


2. Problem Definition

Metamodels define the syntax of models and are needed to load model instances into a modeling tool (e.g., GME). If a metamodel undergoes frequent evolution, then previous model instances may become orphaned from the new definition. If the metamodel get lost, we cannot load and view existing model instances. **MARS** (**MetAmodel Recovery System** using grammar inference) was developed to solve this problem.

3. Overview of MARS

Metamodel Inference Process



3.3 MRL

❖**MRL** (Model Representation Language) is a DSL to bridge the gap between the different representations and capture the essence of the instance models.

❖In order to capture the folder information appearing in the above figure, the complete folder path from "RootFolder" to every item is recorded in the following MRL program.

```
… …
model RootFolder::NewEventTypes::EventType1 {
  fields …  connections …
}

model RootFolder::NewEventTypes::subfolder1::EventType2 {
  fields …  connections …
}

model RootFolder::NewEventTypes::subfolder1::subfolder2::
EventType3 {  fields …  connections …
}

model RootFolder::NewEventTypes::subfolder1::subfolder2::
subfolder3:: EventType4 {  fields …  connections …
}
… …
```

3.2 XSLT Translator

❖**XSLT** (Extensible Stylesheet Language Transformation Language) uses the XML Path Language to locate specific nodes in an XML document.

❖There is a mismatch between the XML representation of models and the syntax expected by the grammar inference tool. The XSLT Translator transforms the model in XML file into a DSL (Domain–Specific Language) and filters the accidental complexities of the XML representation of a model.

❖The XSLT Translator can be downloaded from:

http://www.cis.uab.edu/softcom/MARS/

3.1 ESML Model in XML

❖Modeling tools export a model as an **XML** file (e.g., GME) .

```
01<folder id="id-006a-0001" kind="RootFolder">
02  <name>Root Folder</name>
03      … …
04  <folder id="id-006a-0004" kind="EventTypes">
05    <name>NewEventTypes</name>
06    <folder id="id-006a-0005" kind="EventTypes">
07      <name>subfolder1</name>
08      <folder id="id-006a-0009" kind="EventTypes">
09        <name>subfolder2</name>
10        <folder id="id-006a-000a" kind="EventTypes">
11          <name>subfolder3</name>
12          <model id="id-0065-000b" kind=""EventType>
13            <name>EventType3</name>
14          </model>
15        </folder>  //end "subfolder3"
16        <model id="id-0065-000a" kind="EventType">
17          <name>EventType2</name>
18        </model>
19      </folder>  //end "subfolder2"
20      <model id="id-0065-0003" kind="EventType">
21        <name>EventType1</name>
22      </model>
23    </folder>  //end "subfolder1"
24    <model id="id-0065-0004" kind="EventType">
25      <name>EventType</name>
26    </model>
27  </folder>  //end "NewEventTypes"
28      … …
29</folder>  //end "Root Folder"
```

3.4 LISA Grammar Inference Engine

❖ The grammar inference process is performed within the **LISA** language description environment.

❖LISA was chosen for this project because of its benefits in designing DSLs and since it has been used successfully in our evolutionary-based context-free grammar (CFG) inference engine.

3.5 Inferred Grammar & Inferred Metamodel

Inferred Grammar is generated as the final result of the inference process and concurrently it is transformed into the **Inferred Metamodel.** With the folder information of each element recorded in MRL, the inferred metamodel can recover all viewpoints used in the instance models and place each inferred element under the correct viewpoint where they came from when they were first created.

4. Conclusion

❖This system allows metamodels to be inferred from a set of model instances, including those with multi-tiered domains, such as ESML.

❖ We are currently investigating the scalability of the approach to even larger metamodels.