

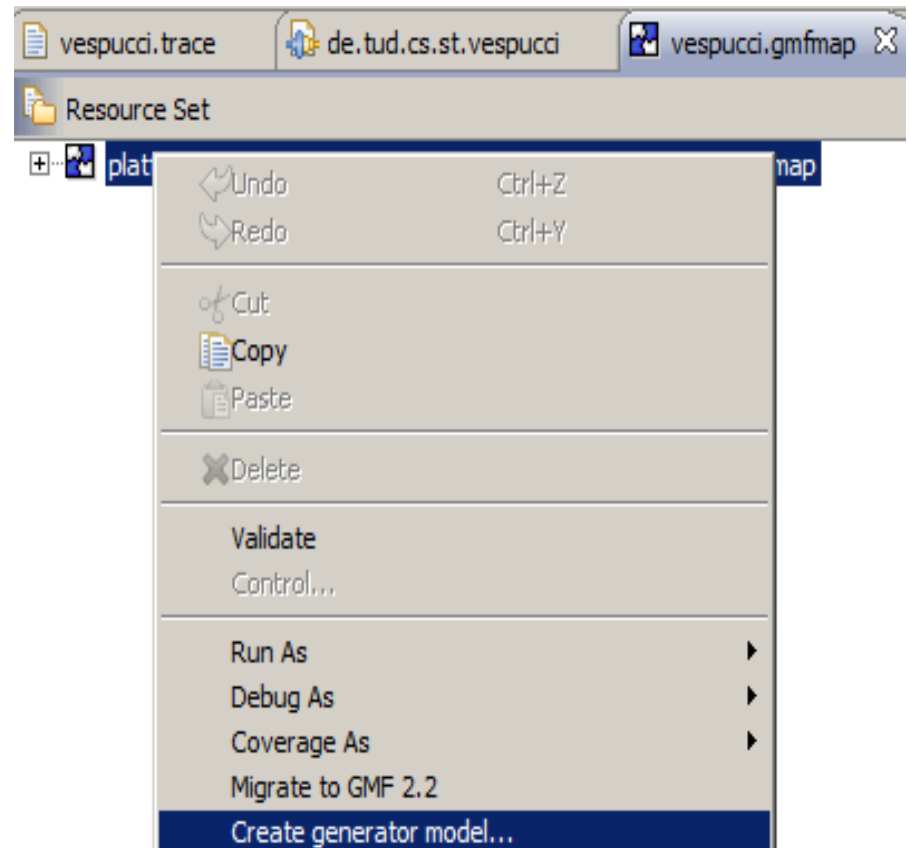
Funktion des „Traces“ (*<Projektname>.trace*) eines GMF- Projekts

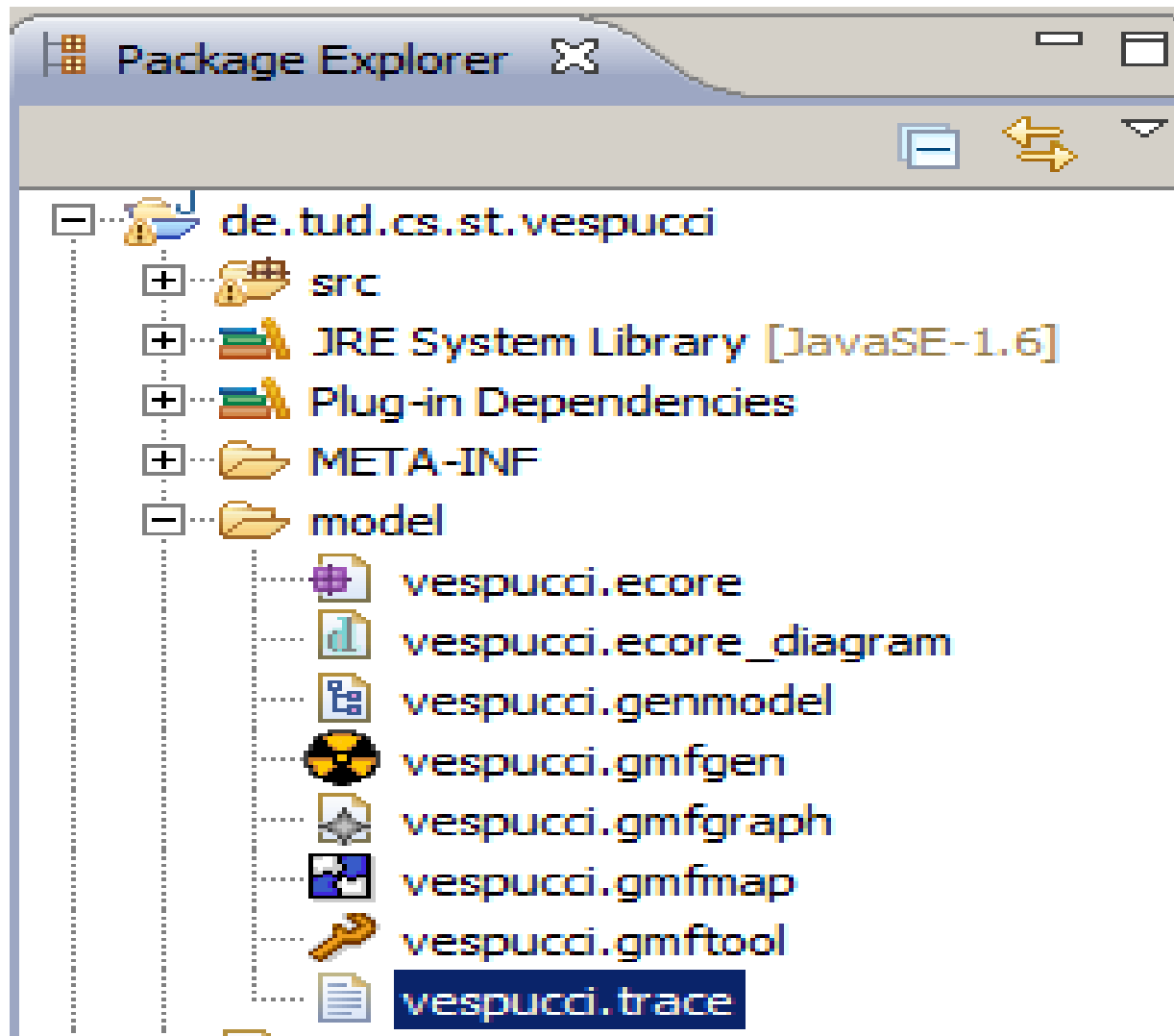
Theo Kischka, Dominic Scheurer, Thomas Schulz, Alexander Weitzmann

1. Wie und wo wird der Trace erzeugt?
2. Wie ist der Trace aufgebaut?
3. Wie kann man den Trace einsetzen?

1. Erzeugung der Trace-Datei

- Plugin:
org.eclipse.gmf.bridge.trace (im GMF-SDK mitgeliefert)
- Wird beim Erzeugen des Generator-Models automatisch im selben Ordner erzeugt
- (Durch Auswählen von „**Create generator model**“ im Popupmenu bei Rechtsklick auf das Mapping-Model)





2. Aufbau des Traces


- XML-Format
- Beschreibt wie grafische Repräsentationen des Domänenmodells aus ihren GEF-“Bauteilen“ (Compartments, Nodes, Links, ...) zusammengesetzt sind
- Jedes Bauteil erhält eine ID

Teile des Domänenmodells

```
<?xml version="1.0" encoding="UTF-8"?>
<trace:TraceModel xmi:version="2.0"
  xmlns:xmi="http://www.omg.org/XMI"
  xmlns:trace="http://www.eclipse.org/gmf/2006/Trace">
  <nodeTraces
    visualID="2001"
    queryText="(let _eClass :ecore::EClass = modelFace!
Facet.containmentMetaFeature.ecoreFea!
  <nodeTraces
    visualID="2002"
    queryText="(let _eClass :ecore::EClass = modelFace!
et.containmentMetaFeature.ecoreFeatur
  <childNodeTraces
    visualID="3001"
    queryText="(let _eClass :ecore::EClass = modelFace!
Facet.containmentMetaFeature.ecoreFea!
  <childNodeTraces
    visualID="3003"
    queryText="(let _eClass :ecore::EClass = modelFace!
et.containmentMetaFeature.ecoreFeatur
  <linkTraces
    visualID="4001"
    queryText="modelFacet.ocIsKindOf(gmfgen::TypeLinkI
r' and _eClass.name = 'InAndOut')">
  <linkTraces
    visualID="4002"
    queryText="modelFacet.ocIsKindOf(gmfgen::TypeLinkI
r' and _eClass.name = 'Expected')">
  <linkTraces
    visualID="4003"
    queryText="modelFacet.ocIsKindOf(gmfgen::TypeLinkI
r' and _eClass.name = 'Outgoing')">
  <linkTraces
    visualID="4004"
    queryText="modelFacet.ocIsKindOf(gmfgen::TypeLinkI
r' and _eClass.name = 'NotAllowed')">
  <linkTraces
    visualID="4005"
    queryText="modelFacet.ocIsKindOf(gmfgen::TypeLinkI
r' and _eClass.name = 'Incoming')">
</trace:TraceModel>
```

Beschreibung durch GEF-Komponenten

- Auszug: Beschreibung von Ensemble



```
<?xml version="1.0" encoding="UTF-8" standalone="yes" type="text/xml" />
<trace:TraceModel xmi:version="1.0" xmlns:xmi="http://www.omg.org/xmi/1.0" xmlns:trace="http://www.eclipse.org/trace/1.0" />
  <nodeTraces
    visualID="2001"
    queryText="(let _eCla
  <nodeLabelTraces
    visualID="5001"
    queryText="true"/>
  <nodeLabelTraces
    visualID="5008"
    queryText="true"/>
  <compartmentTraces
    visualID="7001"
    queryText="true"/>
  </nodeTraces>
```

3. Funktion des Traces

- Protokolliert visuelle ID
- ID's werden für das Generator-Modell übernommen -> Änderungen der ID's in der Trace-Datei wirken sich bei Neugenerierung auf das Generator-Modell aus
- Problem: Für jedes Diagramm basierend auf einem einzigen Domänenmodell werden jeweils eigene Instanzen erzeugt (schlecht wenn Synchronization gewünscht)
- Durch Manipulation der ID's in der Trace-Datei und den Generatormodellen kann gemeinsame Verwendung von Instanzen erzwungen werden

Quellen

„Shared Editing Domain“

<http://code.google.com/p/gmftools/wiki/SharedEditingDomain>

<http://wiki.eclipse.org/GMF/Tips>