

1. Einleitung

Für unser Projekt ist JUnit Verwendung zum größten Teil nicht relevant, weil JUnit nicht ausgeführt werden kann, da die meisten umgebenden Objekte und Variablen in unserem Projekt nur zur Laufzeit erzeugt werden und sich einzelne Klassen nicht aus dem Gesamtablauf des GMF-Editors extrahieren lassen. Die Algorithmen, die von unseren Klassen verwendet werden, sind größtenteils ausgelagert und ihre Rückgabewerte lassen sich schwer überprüfen. Beispiel: Der Übergabewert diagram zu PluginEcoreNeu.ProjektDateienErzeugen() gehorcht komplexen Gesetzen, die mit JUnit schwer zu simulieren sind.

Viel besser ist es da, die konkreten Anwendungsfälle zu überprüfen und zu testen, ob die erzeugten Dateien korrekt implementiert sind. Dafür haben wir folgende Kriterien verwendet.

2.0. Testkriterien

Erläuterungen:

'Beschriftung': Standardbeschriftung aus dem Label des Elements

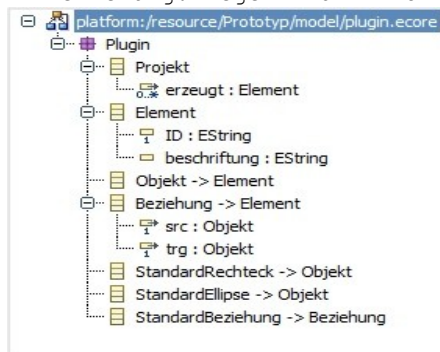
'ID': ID aus dem Label des Elements

'Form': Rectangle, Ellipse, Rounded Rectangle, Polyline Connection

2.1. bei jeder Datei: Lesbarkeit durch den jeweiligen Editor

2.2. ecore

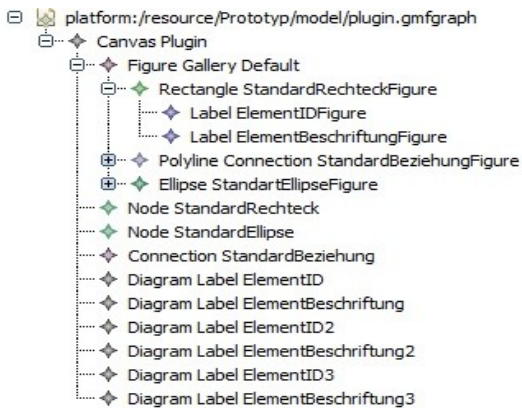
- Wurzelement "Projekt" mit containment-Referenz auf alle Elemente
- alle erzeugten eClasses müssen von "Element" erben
- Klassenname muss aus Element-ID abgeleitet werden.
- Standardbeschriftung muss als "Beschriftung"-Attribut und mit richtigem default-Wert in die eClass übernommen werden
- Beziehung: Src und Trg-Referenz muss richtig gesetzt sein
- Beziehung: Regeln für Anzahl und Art der Referenzen müssen korrekt übernommen werden



2.3. gmfigraph

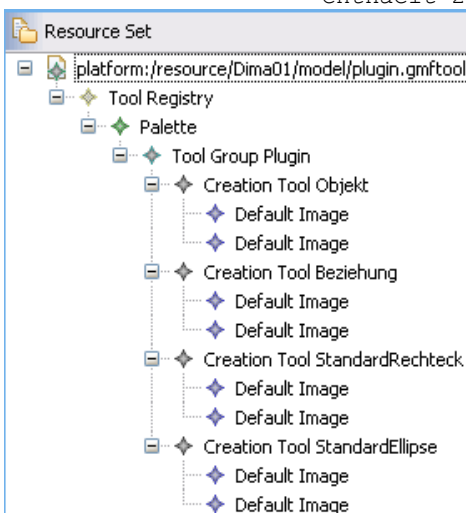
Objekt muss wie folgt erzeugt werden:

- in Figure Gallery „Default“:
 - Child Bezeichnung: "'Form' 'ID'Figure"
 - Child für "'Form' 'ID'Figure" soll ein Label sein: "Label 'ID'BeschriftungFigure"
- Node soll heißen: "Node 'ID'" und muss auf entsprechende "'Form' 'ID'Figure" zeigen
- Connection soll heißen: „Connection 'ID'“ und muss auf entsprechende Polyline Connection verweisen.
- "DiagramLabel 'ID'" der auf die entsprechende "Label 'ID'BeschriftungFigure" zeigt.



2.4. Gmftool

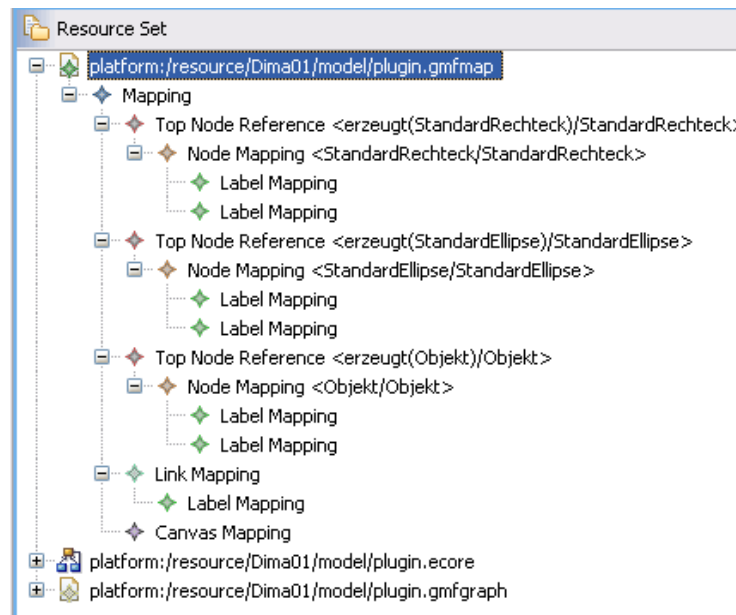
- 1 Wurzelement "Tool Registry" enthält
 - 1 Element "Palette" enthält
 - 1 Element Tool Group "Plugin" enthält:
 - 1 Creation Tool "'ID'" für jedes Objekt und jede Beziehung:
 - Eigenschaft: Description = "Erzeuge 'Beschriftung'"
 - Eigenschaft: Title = "'Beschriftung'"
 - enthält 2 "Default Image"



2.1. gmfmap

- 1 Wurzelement "Mapping" enthält
 - 1 "Top Node Reference" für jedes Objekt mit folgenden Eigenschaften:
 - Containment Feature = "EReference erzeugt"
 - enthält Node Mapping mit folgenden Eigenschaften:
 - Element = EClass "'ID'"
 - Diagram Node = Node "'ID'"

- Tool = Creation Tool ""ID""
- enthaelt: 1 Label Mapping mit folg.E.:
 - Diagram Label = Diagram Label ""ID'Beschriftung"
 - Features = EAttribute "beschriftung"
- 1 "Link Mapping" für jede Beziehung mit folgenden Eigenschaften:
 - Containment Feature = "EReference erzeugt"
 - Element = EClass ""ID""
 - Source Feature = EReference "src"
 - Source Feature = EReference "trg"
 - Diagram Node = Node ""ID""
 - Tool = Creation Tool ""ID""
 - enthaelt: 1 Label Mapping mit folg.E.:
 - Diagram Label = Diagram Label ""ID'Beschriftung"
 - Features = EAttribute "beschriftung"
- 1 Canvas Mapping mit folgenden Eigenschaften:
 - Domain Model = EPackage Plugin
 - Element = EClass "Projekt"
 - Palette = "Palette"
 - Diagram Canvas = Canvas "Plugin"



3. Testbericht

Es soll für jeden Testbericht ein eigenes Dokument pro Projekt erzeugt werden, in dem folgende Vorlage abgearbeitet wird.

Wenn auf eine Frage nein geantwortet wurde, dann muss dahinter ein Eintrag erfolgen, was in welcher Klasse falsch eingetragen wurde. Beispiel: /lesbar?=nein, Top Node Reference von StandardRechteck kann nicht geöffnet werden

/lesbar? fragt, ob das Dokument richtig vom zugehörigen Editor gelesen werden kann – also, jedes Element kann mit seinen Kindern angezeigt werden.

/Namen? fragt, ob die Namenskonventionen richtig eingehalten wurden.

/Referenzen? fragt, ob alle Referenzen richtig gesetzt und implementiert wurden.

/Vererbung? fragt, ob alle Elemente von der Superklasse Element erben.

/Verwandschaft? fragt, ob alle Verwandschaftsverhältnisse richtig sind.

/Werte? fragt, ob alle Werte richtig gesetzt wurden: Kardinalität, Farbe und Beschriftung im

ecore – nicht beachten bei Release 1

/Mapping? fragt, ob alle Mappingeinträge richtig vorgenommen wurden.

Projektdatei: ecore

/lesbar?=ja/nein

/Namen?=ja/nein

/Referenzen?=ja/nein

/Vererbung?= ja/nein

/Werte?=ja/nein

Projektdatei: gmfgraph

/lesbar?=ja/nein

/Namen?=ja/nein

/Referenzen?=ja/nein

/Verwandschaft?= ja/nein

Projektdatei: gmftool

/lesbar?=ja/nein

/Namen?=ja/nein

/Referenzen?=ja/nein

/Verwandschaft?= ja/nein

Projektdatei: gmfmap

/lesbar?=ja/nein

/Namen?=ja/nein

/Referenzen?=ja/nein

/Verwandschaft?= ja/nein

/Mapping?=ja/nein