Building Graphical Editors using GMF Technology

Model Driven Software Development
Today Workshop

Software Engineering 07

Hamburg, *29th of March 2007*

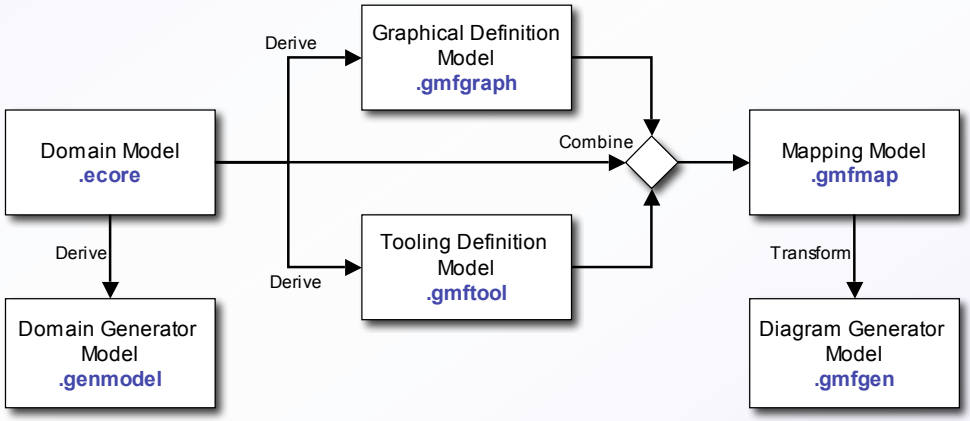
Helge Klimek - Helge.Klimek /a/ TU-Harburg.de

Building Graphical Editors using GMF Technology

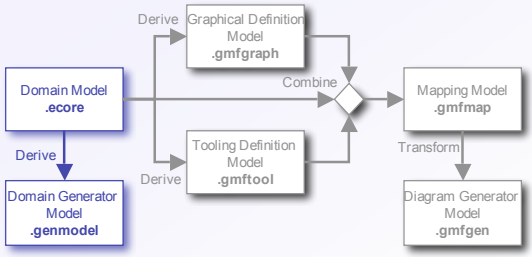
- A practical introduction

- 20 Minute walk through
  - Building a graphical editor
    - Metamodel, tooling definition, graphical model, mapping definition,...
  - Adding constraints

GMF tool chain (1/4)

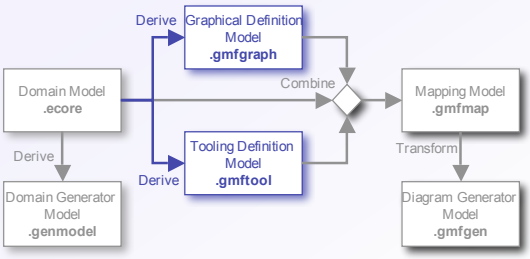


GMF Dashboard, in Eclipse:
Window -> Show View -> Other
-> General -> GMF Dashboard



GMF tool chain (2/4)

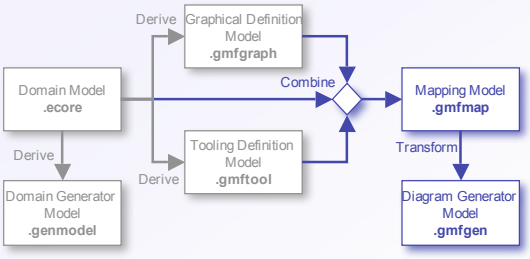
- Domain Model (EMF based)
  - Aka. metamodel in MDSD
  - Defines abstract syntax
  - Constraints
    - Structural constraints used to model static semantics
- Domain Generator Model
  - Decorator model around the metamodel
    - Holds informations about e.g. package base names, templates to use, ...



GMF tool chain

(3/4)

- Graphical Definition Model
  - Model defining graphical notation: shapes, decorations, labels and connections
  - Defines concrete syntax
- Tooling Definition Model
  - Model for the editor‘s palette
    - E.g. Icons for and grouping of buttons



GMF tool chain

(4/4)

- Mapping Model
  - Binds graphical definition and tooling with the meta model
  - Defines how graphical elements are mapped to a model instance with respect to the meta model
- Diagram Generator Model
  - Decorator model around the mapping model
    - Holds informations such as package base names, enabling of validation, - decorators, ...

How about an example?

- *Model editor for some sort of diagram?*
- Something related to the SE07?
- Or to Hamburg?
- Maybe to the University of Hamburg?
- Famous (german) computer scientist?

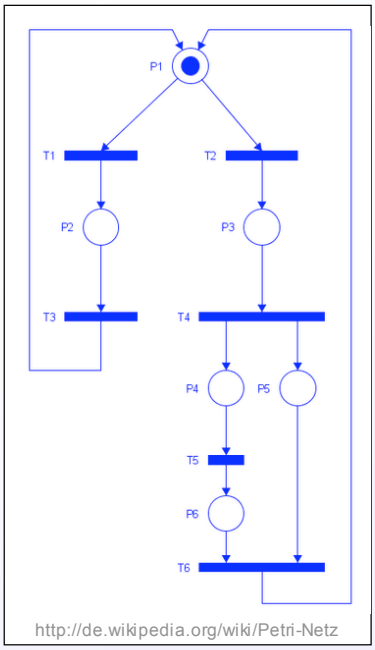
- About 35 years ago, Carl Adam Petri, invented the **Petri Nets** in his Ph.D. thesis
  - Honoraryprofessor in Hamburg since 1988
  - Officially retired in 1991
  - Has still an office in the University of Hamburg

http://en.wikipedia.org/wiki/Petri_net

Example Domain: Petri Nets

(1/2)

- A petri net consists of
  - Places (circles)
  - Transitions (rectangles)
- They are interconnected by directed arcs
- No direct connections between two places or two transitions.

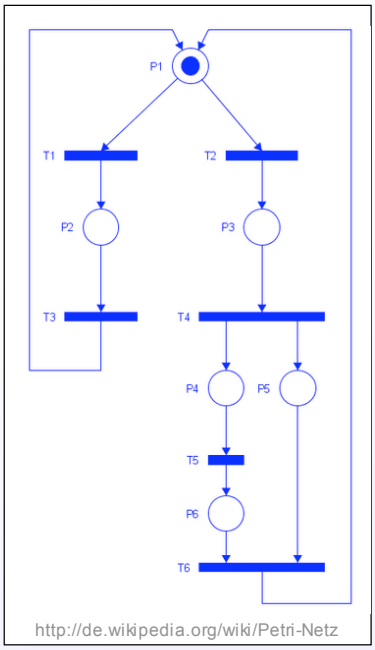


http://de.wikipedia.org/wiki/Petri-Netz

Example Domain: Petri Nets

(2/2)

- Places have a capacity, specifying how many token they can hold
  - If none: 1 or infinite is assumed
- Edges have a weight, representing costs
  - If none: 1 is assumed



http://de.wikipedia.org/wiki/Petri-Netz

References

1. Markus Voelter, Bernd Kolb, Sven Efftinge, and Arno Haase, From Front End To Code - MDSD in Practice: *http://www.eclipse.org/articles/Article-FromFrontendToCode-MDSDInPractice/article.html*
2. GMF Tutorial Part 1: *http://wiki.eclipse.org/index.php/GMF_Tutorial*
3. GMF Tutorial Part 2: *http://wiki.eclipse.org/index.php/GMF_Tutorial_Part_2*
4. GMF Tutorial Part 3: *http://wiki.eclipse.org/index.php/GMF_Tutorial_Part_3*
5. GMF Tutorial Part 4: *http://wiki.eclipse.org/index.php/GMF_Tutorial_Part_4*
6. GMF BPMN Tutorial: *http://wiki.eclipse.org/index.php/GMF_Tutorial_BPMN*
7. GMF Constraints: *http://wiki.eclipse.org/index.php/GMF_Constraints*
8. Madanagopal, Domain Model Integrity for GMF with EMFT OCL: *https://bugs.eclipse.org/bugs/show_bug.cgi?id=163613*