UNIVERSITY OF AGDER

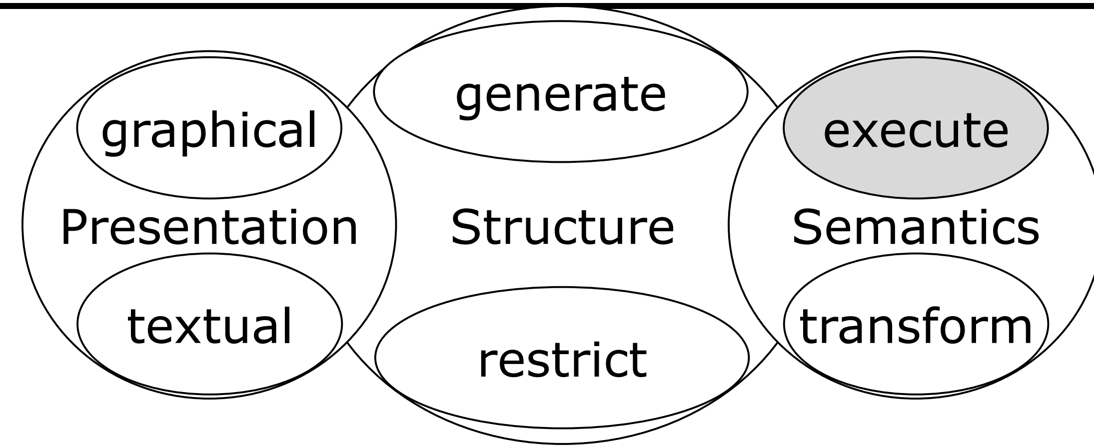
Multi-level Language Descriptions

Andreas Prinz

Introduction
Tools
Runtime states
Instantiation
Summary

Defining language semantics

OMG Level
M3 = meta-language
M2 = language
M1 = model
M0 = instances



- In meta-modelling, semantics is often given by transformation.
- We want to describe execution semantics with two parts: (1) runtime states and (2) runtime state changes.
- We assume that the language structure (meta-model) is given.
- OMG levels are absolute with instantiation between the levels.

- Only execution semantics crosses 2 levels.

Relative architecture

Relative Level	Example 1	Example 2	Example 3
n+2 (language)	MOF	UML	MOF
n+1 (model)	UML	UML model	MOF
n (instances)	UML model	UML objects	UML

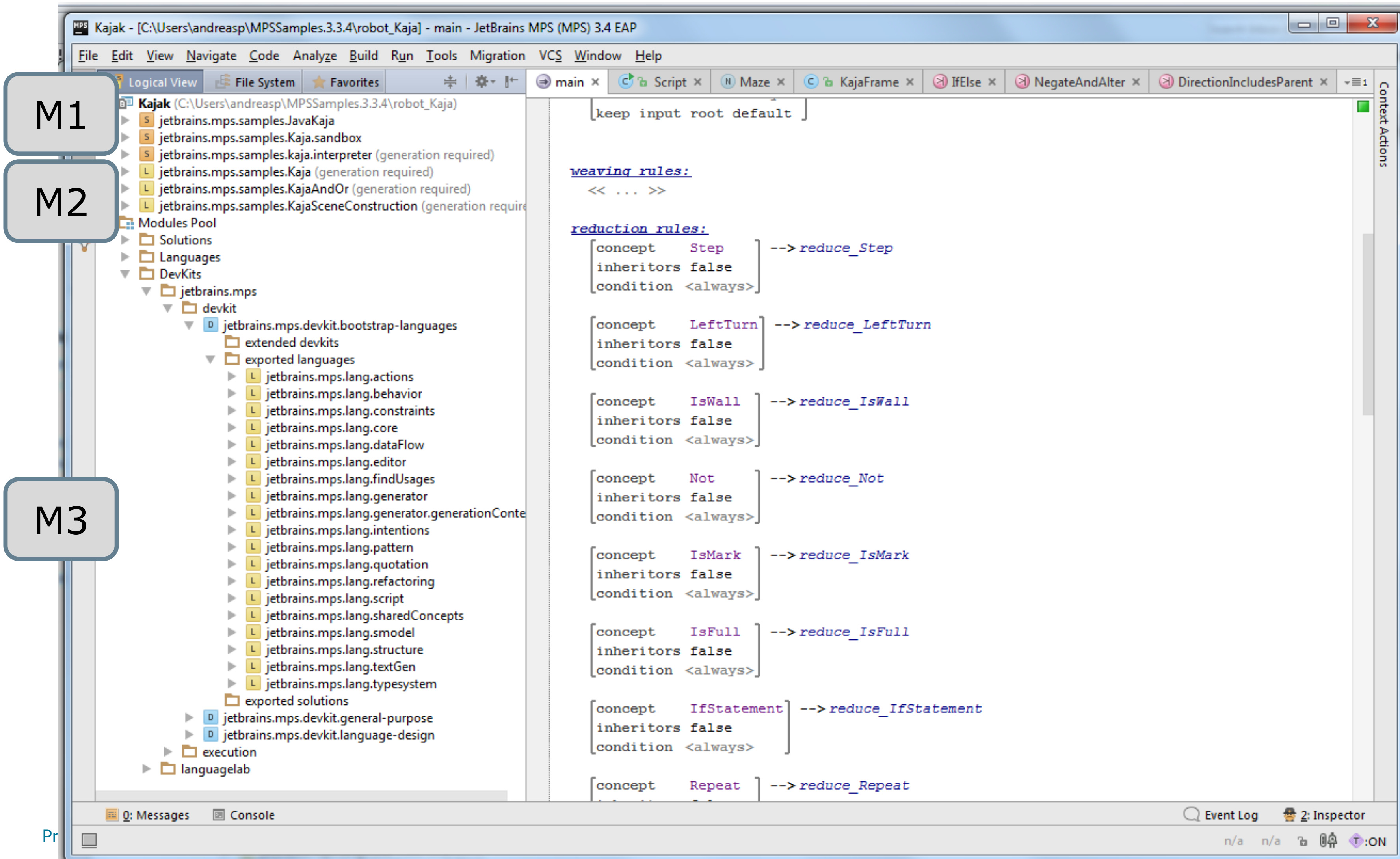
Levels are placed relatively.

Between two adjacent levels there is instantiation.

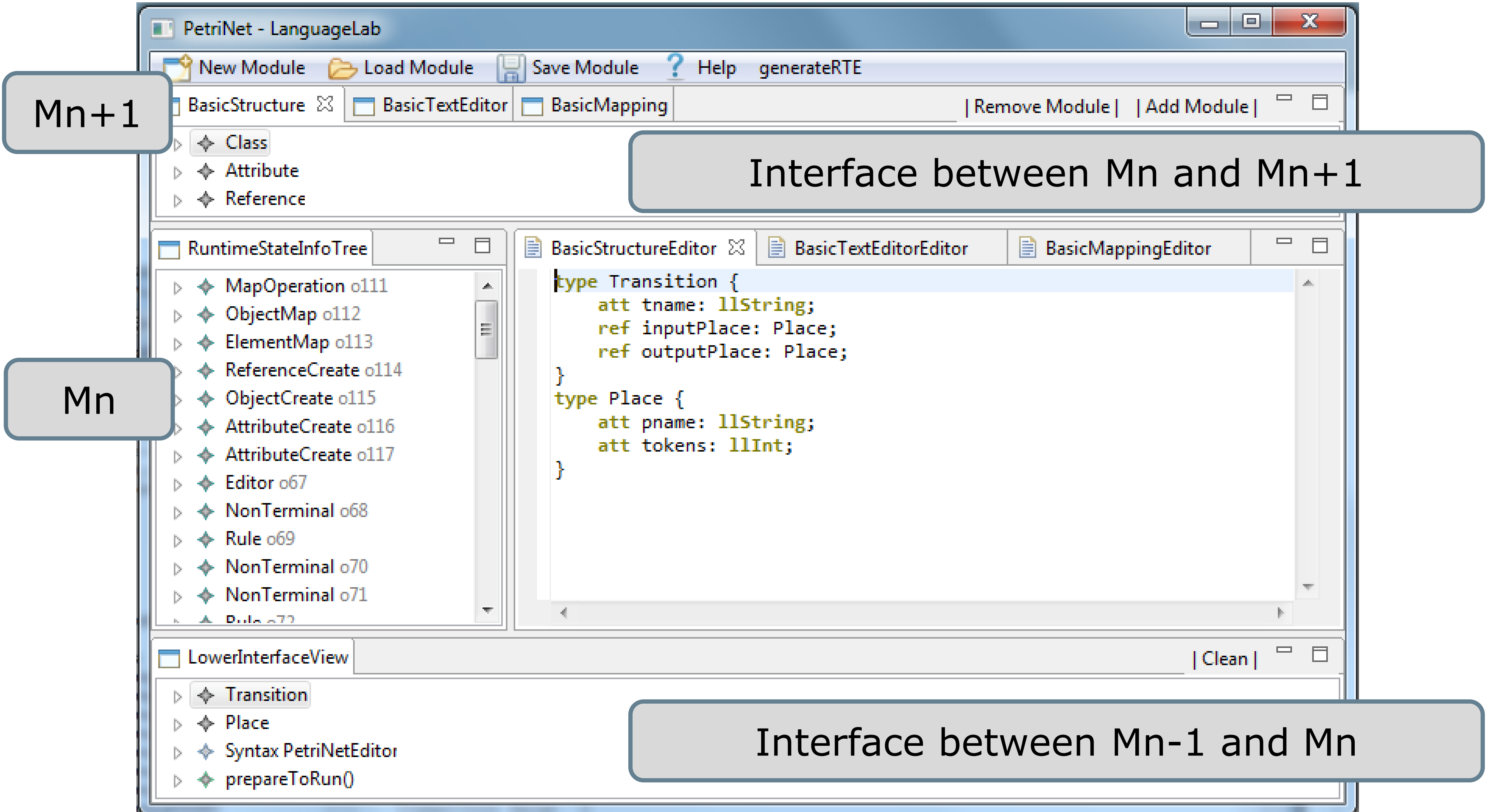
Instantiation semantics describes how to go from n+1 to n.

Execution semantics also describes going from n+1 to n.

MPS (conceptually absolute)

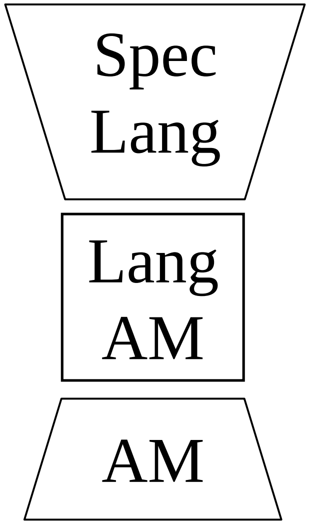


LanguageLab (relative)



Underlying abstract machine

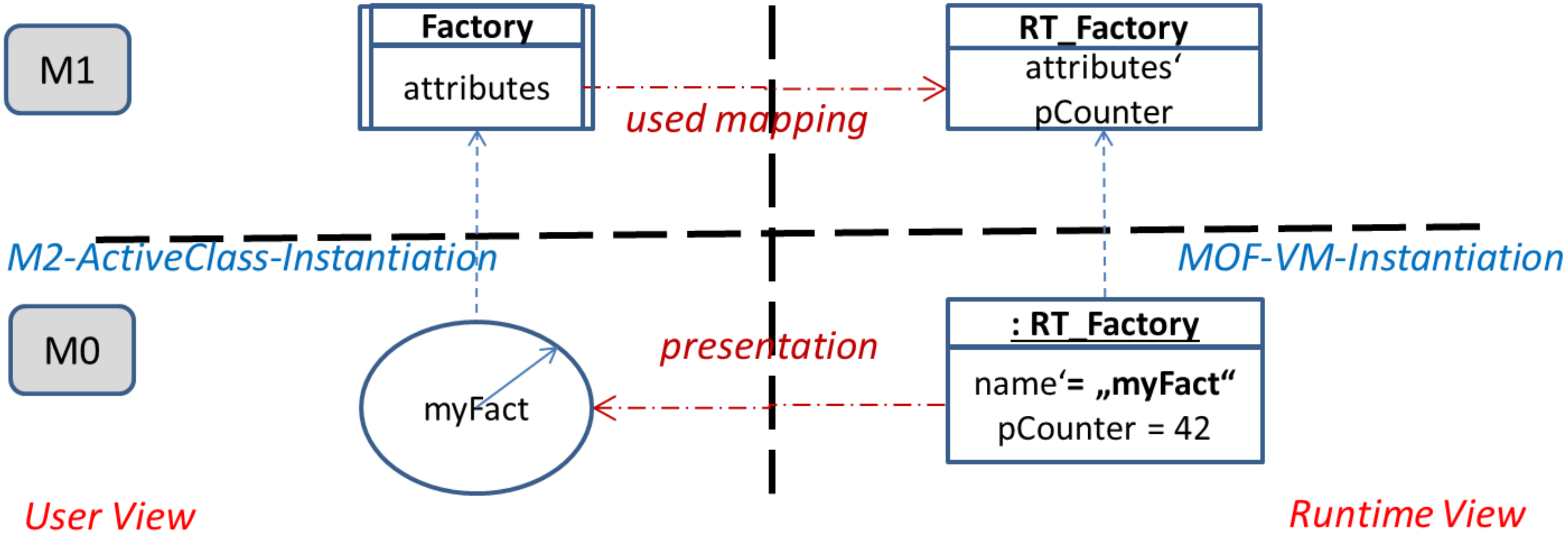
- There is always an underlying (abstract) machine AM.
- What does the AM provide?
  - Functions, parameters
  - Instantiation
- There are runtime states for the underlying machine as well.
- We use a special object-oriented underlying machine: MOF-VM.



What are runtime states (RTE)?

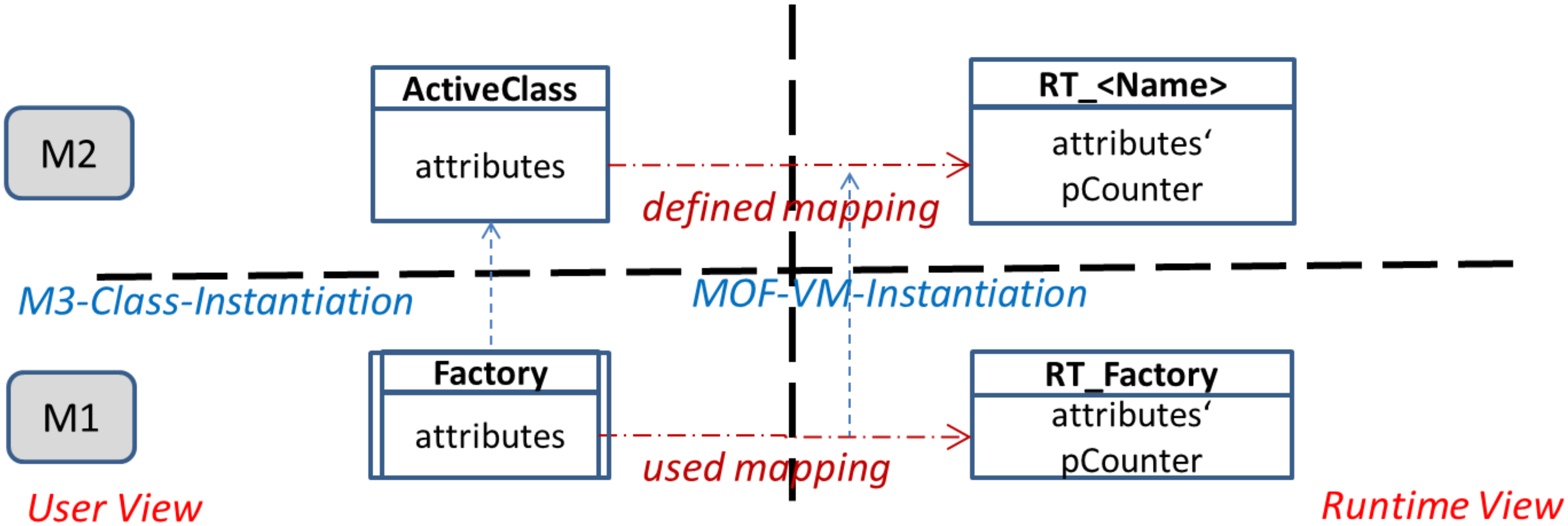
- *Read-only program* included in the RTE
- *Global elements:* independent of the specific program
- *Local elements:* related to language concepts but independent of the instance
  - *None-elements* are not existing at runtime (1:0).
  - *One-elements* are existing at runtime (1:1)
  - *Many-elements* are existing at runtime (1:n)
- *Dependent elements:* related to language concepts and dependent of the instance

Language instantiation



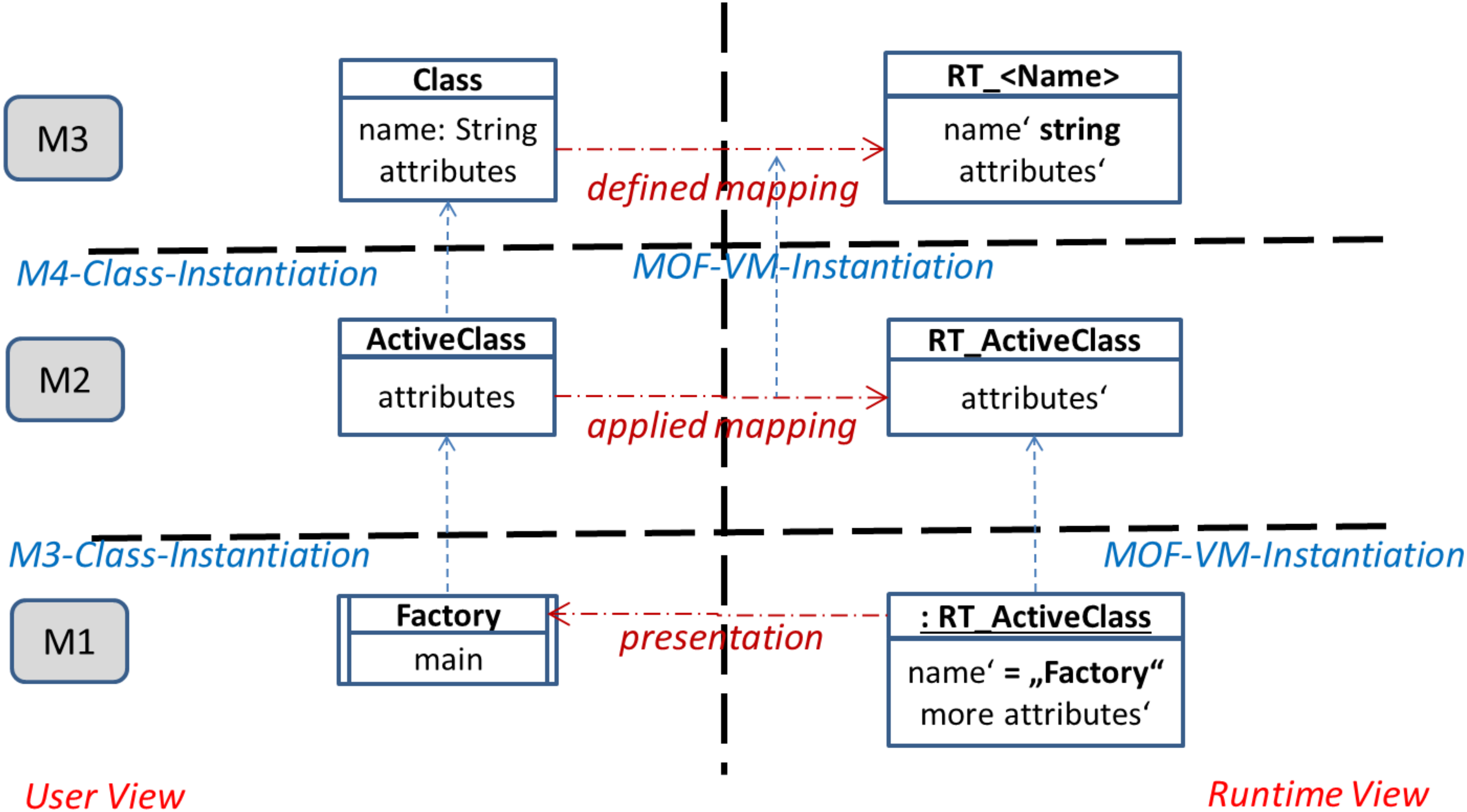
- Language instantiation (linguistic) is based on the underlying machine instantiation (runtime = MOF-VM).

Defining instantiation

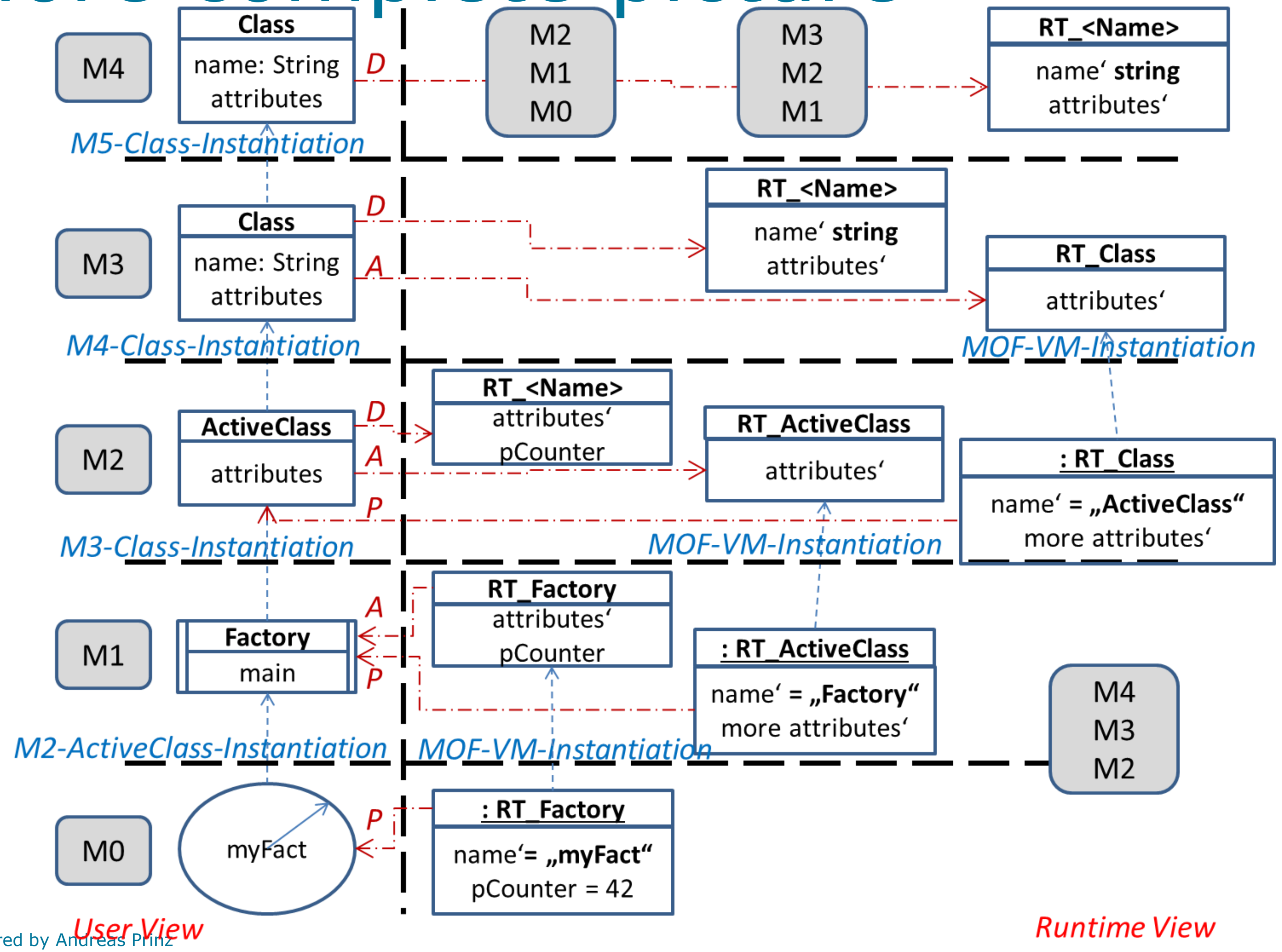


- Instantiation is defined by mapping the language to the underlying machine.
- The mapping is applied on the level below.

Instantiation for metalanguages



More complete picture



Summary

- Language execution semantics needs instantiation semantics and dynamic semantics.
- Instantiation semantics (RTE) is based on an underlying machine (MOF-VM) instantiation.
- Several possible kinds of instantiation relations between specification and RTE were identified.
- They are specified for a language as a mapping between specification and MOF-VM, which is defined at language level and used at specification level.
- There are three kinds of instantiation: linguistic, ontological, and runtime instantiation.

Instances on several levels