

Extending a UML and OCL Tool for Three level-modeling: Application towards Reflective Constraints and Model Quality Assessment

Khanh-Hoang Doan and Martin Gogolla, *University of Bremen*

Introduction

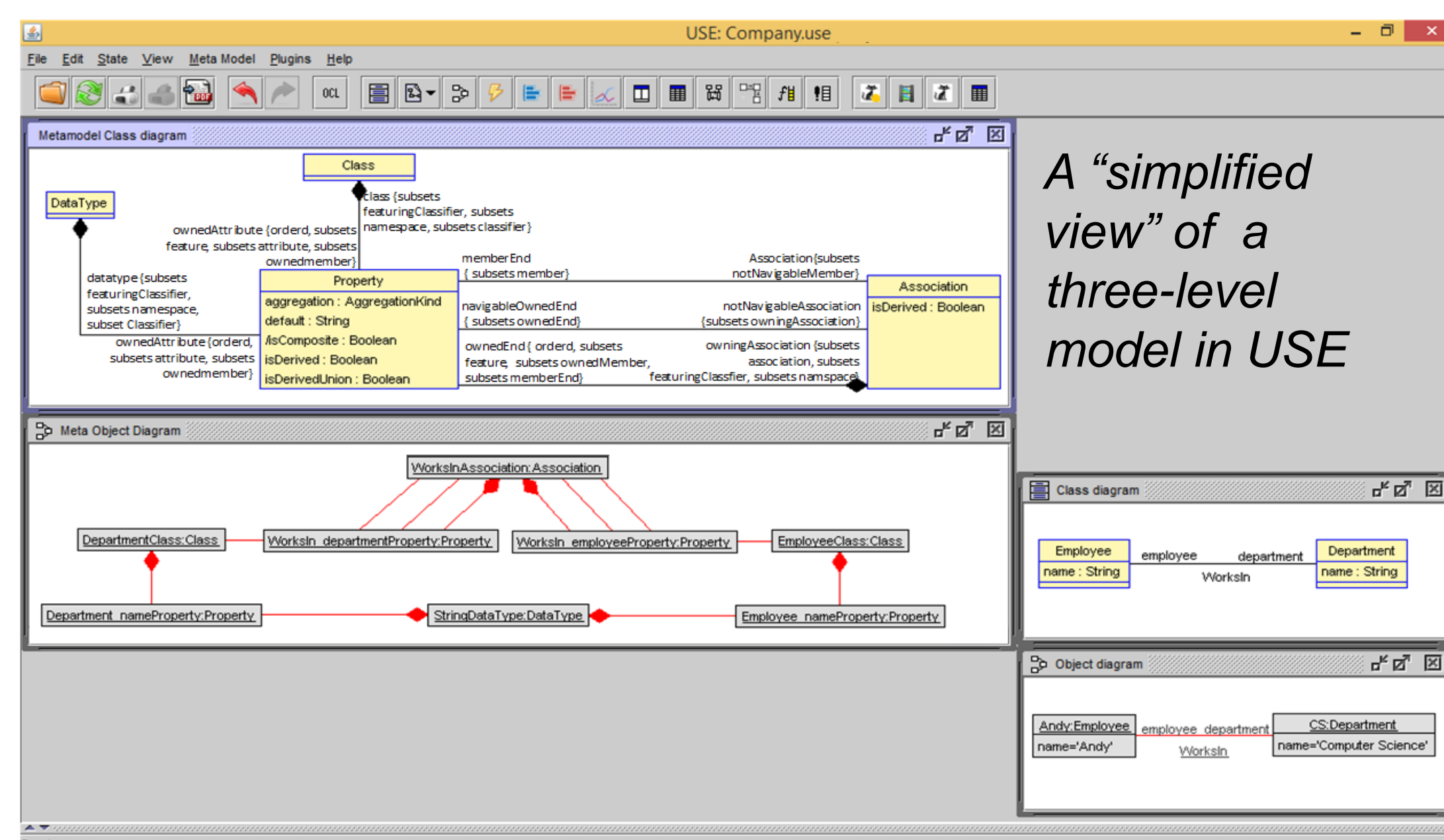
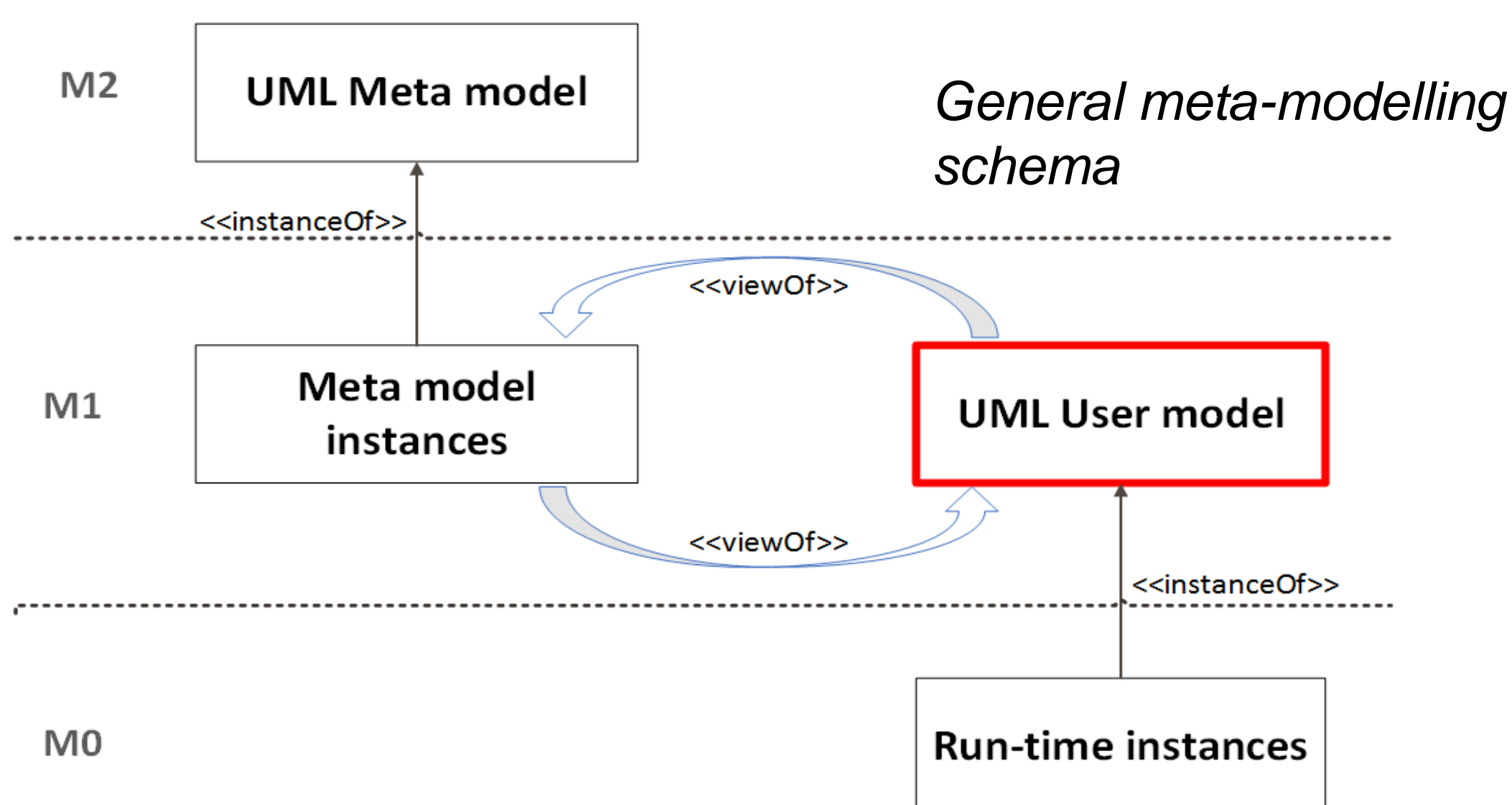
- Standard OCL does not support reflective constraints (considering the current model as data).
- Model analysis can benefit from reflective constraints, e.g., model quality assessment.
- Reflective constraints can be achieved by supporting access to the Meta model (M2 level) from the M1 model.

Our objectives:

- Upgrade tool USE to three-level modeling (including a meta level).
- Allow the developer to write reflective constraints in USE.
- Model quality assessment with tool USE.

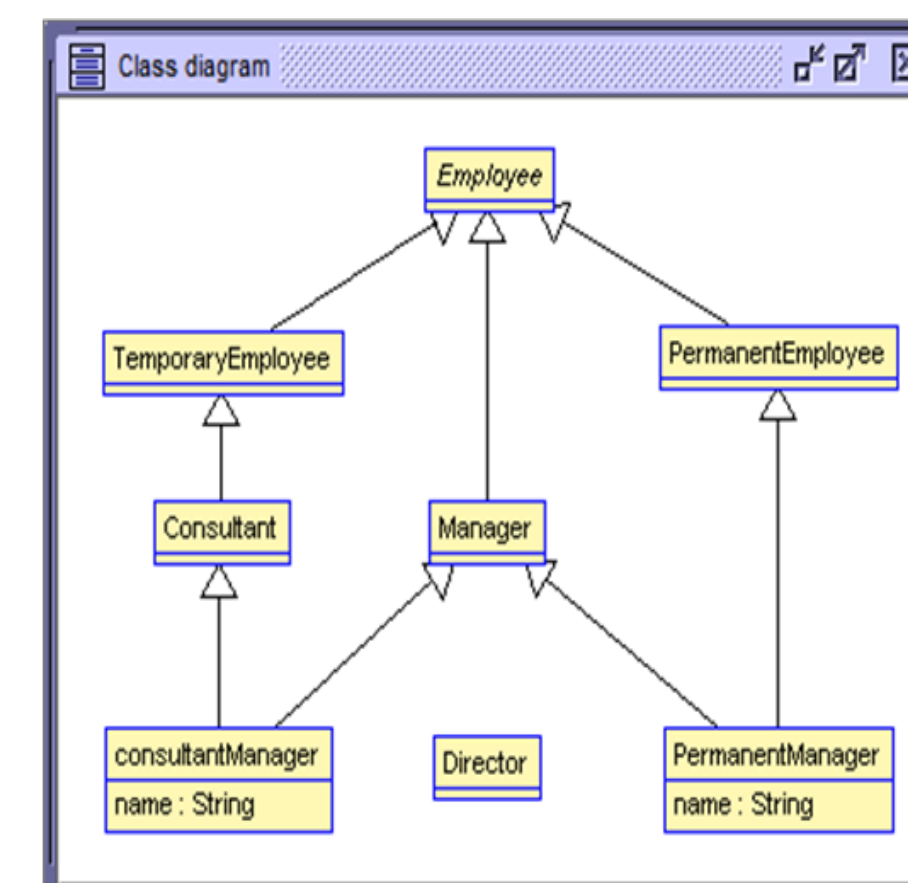
Meta-modeling in Tool USE

- Add the meta-model (the UML 2.4 superstructure from OMG) at the topmost level.
- This M2-Model contains (currently) 63 classes and 99 associations.
- An auto-generated meta-model instance of the user model is added to the M₁ level.
- Provide a “simplified view” of the meta-model corresponding to the input user model.

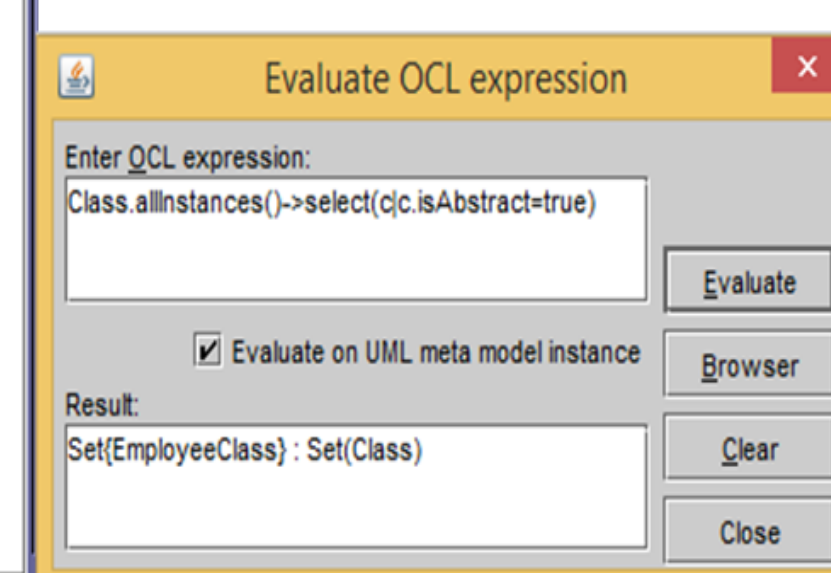


Reflective Constraints in USE

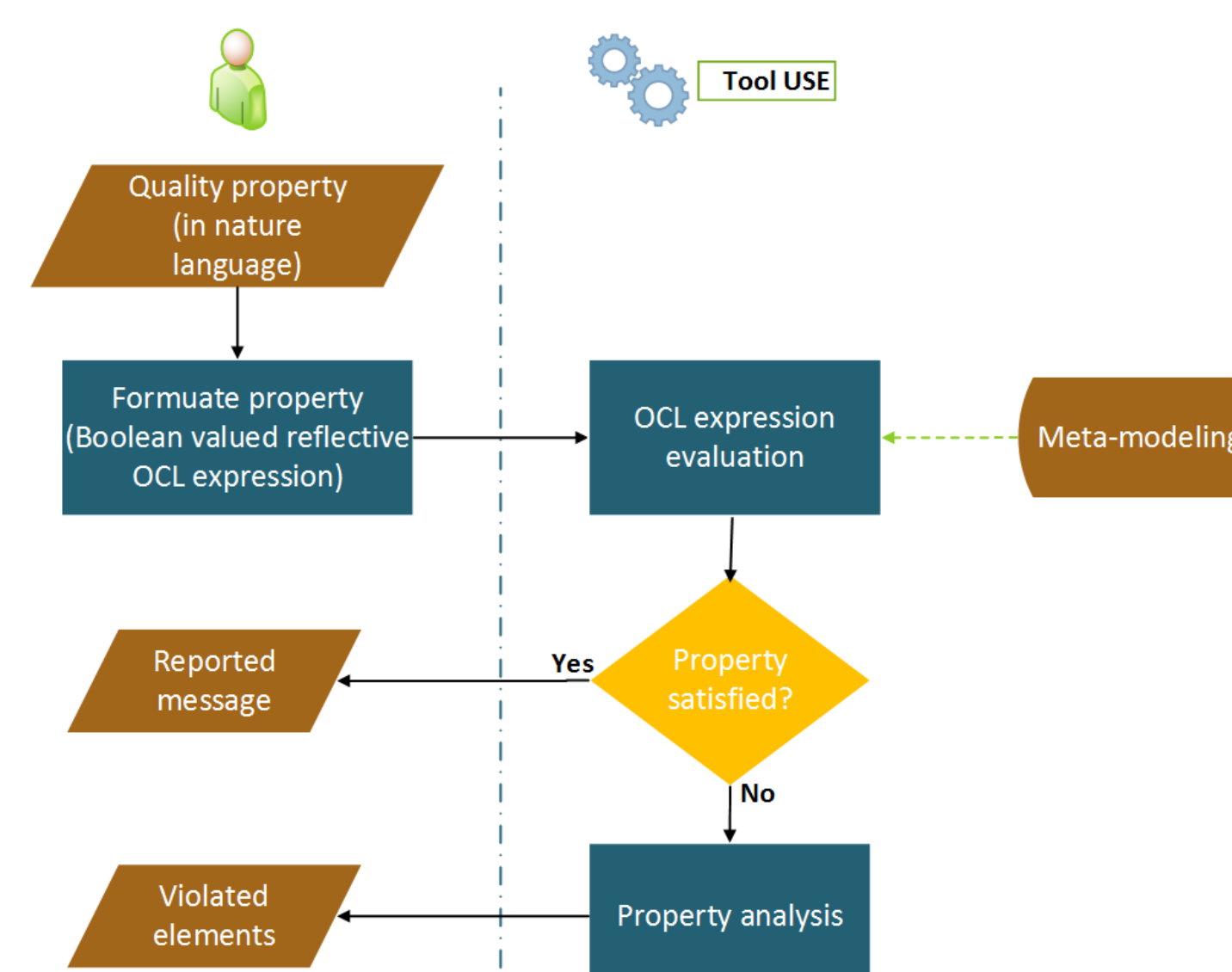
- USE now newly supports access to the meta-model.
- By adding an auto-generated meta-model instance of the user model in the M₁ level -> the model turns into data -> modelers can write OCL expressions to query the model itself.



Example: *find all abstract classes in an Employee hierarchy model.*

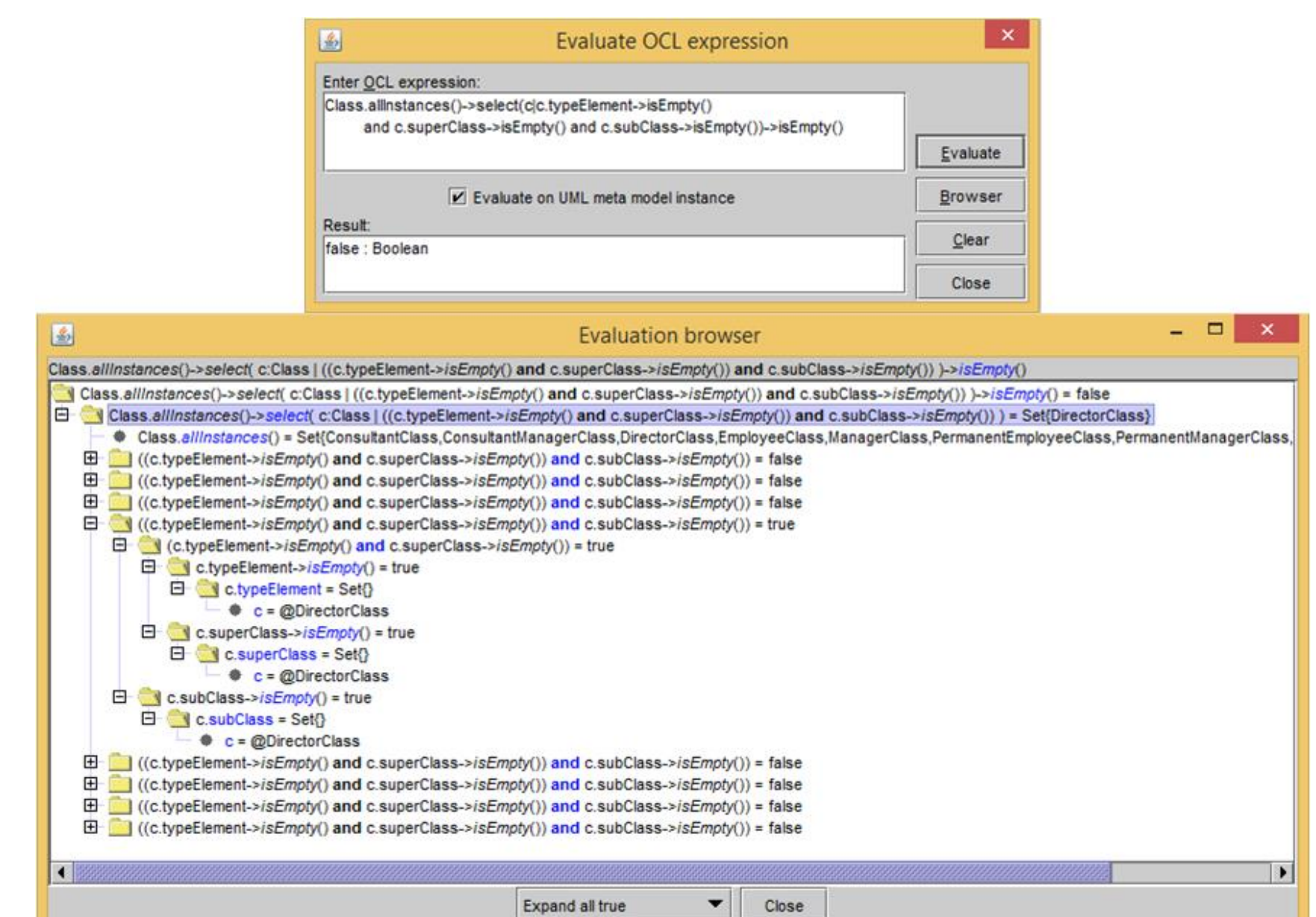


Application: Model Quality Assessment



- Model quality assessment helps modelers to detect errors or mistakes on their model -> fix in their bugs and improve the model.
- Assessment properties might include: Design, Naming conventions, Metrics, etc.

Quality property:
whether there are isolated classes in the Employee hierarchy model



Conclusion

- Tool USE has been upgraded to three-level modeling.
- Reflective queries can be written within the upgraded tool.
- Application of model quality assessment is illustrated with a small example. Dealing with larger models can be done in a similar way.