**MOTIVATION**

Why executing models?
- Due to the Model-Driven Development (MDD) paradigm models become the main artifact of the software development process
- Executing models is a means for testing and validating them
- Defects can be identified in an early stage of the development process

Why executing UML models?
- OMG’s MDD approach suggests the usage of UML for modeling
- UML is the de facto standard for modeling software systems

**PROBLEM STATEMENT**

- The semantics of UML is neither precisely nor completely specified
- Without a clear semantics models can be interpreted and executed in different ways
- OMG released the new fUML standard in February 2011 that specifies the semantics of a subset of UML precisely

**OBJECTIVE**

- Implementation of a prototypical model interpreter for executing and debugging UML activity diagrams like specified in the fUML standard
- Evaluation of the applicability of the fUML standard for building tools that enable the execution of UML models

**APPROACH**

- Literature survey on execution semantics of UML
- Implementation of the prototypical model interpreter as Eclipse plug-in based on the Eclipse Modeling Framework (EMF)
- Evaluation of the prototype and of the fUML standard

**fUML STANDARD**

- "Semantics of a Foundational Subset for Executable UML Models"
- Version 1.0 released in February 2011
- Provides a selection of a foundational subset of the UML 2 metamodel
- Defines the precise execution semantics of that subset with Java code
- Supports a subset of class diagrams and activity diagrams

**RESULT**

Prototypical model interpreter implemented as Eclipse plug-in
- Editors for class diagrams, activity diagrams and diagrams of the expected output of an activity
- Execution and debugging of UML activity diagrams that model the manipulation of objects and links in a system

**CONCLUSION**

- The fUML standard is applicable for implementing tools that support the execution of UML activity diagrams
- But high efforts are necessary in order to develop user-friendly and efficiently usable tools for executing UML models

**Execute Activity**

**Actual Output**

**Expected Output**

**Compare Outputs**

**Debugging functionality**

- Step-wise execution of an activity and setting of breakpoints
- Visualization of debugging process, execution trace, runtime objects