

Model-based analysis using the Escalator tool

Mass Soldal Lund

Model-based system analysis workshop

21 May 2008



Overview



- **Model Driven Development (MDD)**
- **Model-based analysis**
 - Stepwise refinement
 - Model-based testing
- **Escalator tool**
- **Demo**
- **Summary**



Model Driven Development

- **Important trend in system development**
- **Software development with models as the driving artefacts**
 - Model transformation
 - Code generation
- **Means to handle the complexity of large systems**
- **Facilitates abstraction**
- **Formalization**
- **Automation**
- **Documentation**

Correct by construction?

- **Douglas C. Schmidt in *IEEE Computer special issue on model driven software development*, February 2006:**
 -  The ability to synthesize artefacts from models helps ensure the consistency between application implementations and analysis information associated with functional and QoS requirements captured by models.
 -  This automated transformation process is often referred to as “correct-by-construction,” as opposed to conventional handcrafted “construct-by-correction” software development processes that are tedious and error prone.



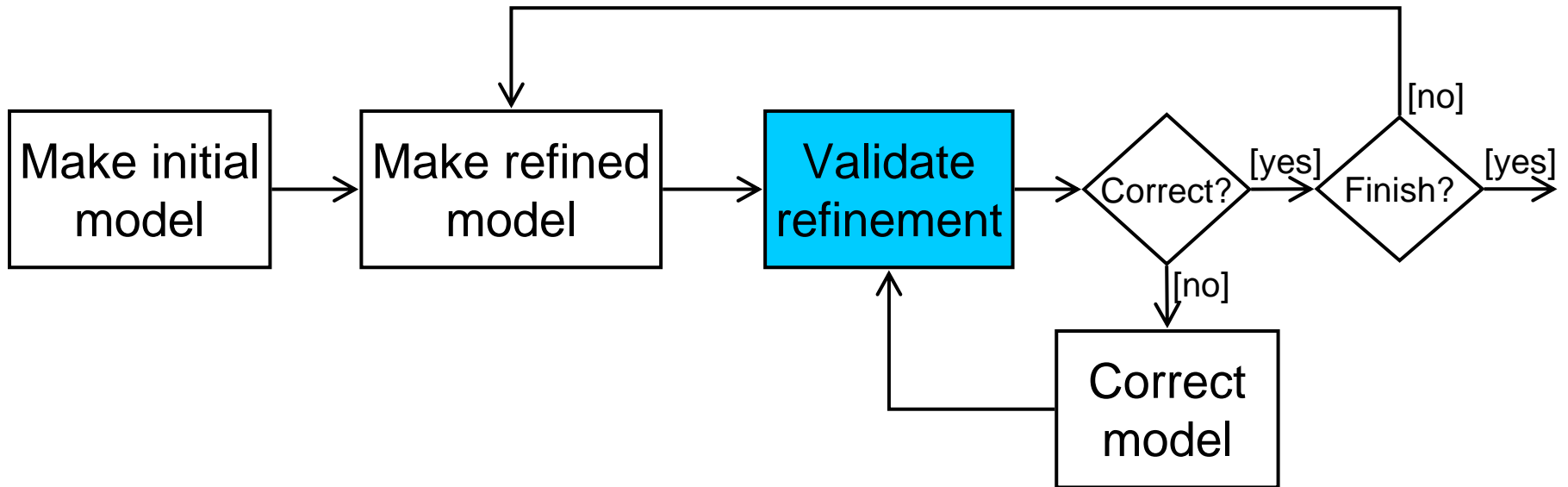
Model-based analysis

- **MDD *does not* ensure correct systems**
- **MDD presupposes the *correctness of models***
- **Making (and refining) the models is still a human activity**
 - and still tedious and error prone
- **Model-based analysis is necessary to ensure good and correct models**
- **In this presentation we look two methods of model-based analysis**
 - Stepwise refinement
 - Model-based testing
 - Supporting tool

- **Stepwise refinement is a form of iterative development**
 - The model/specification is brought closer to implementation for each iteration
- **Each step should be validated to ensure correct refinements**
 - Made possible by formalization of refinement relation
- **Validation should be combination of pragmatics and automated analysis**
 - Refinement verification



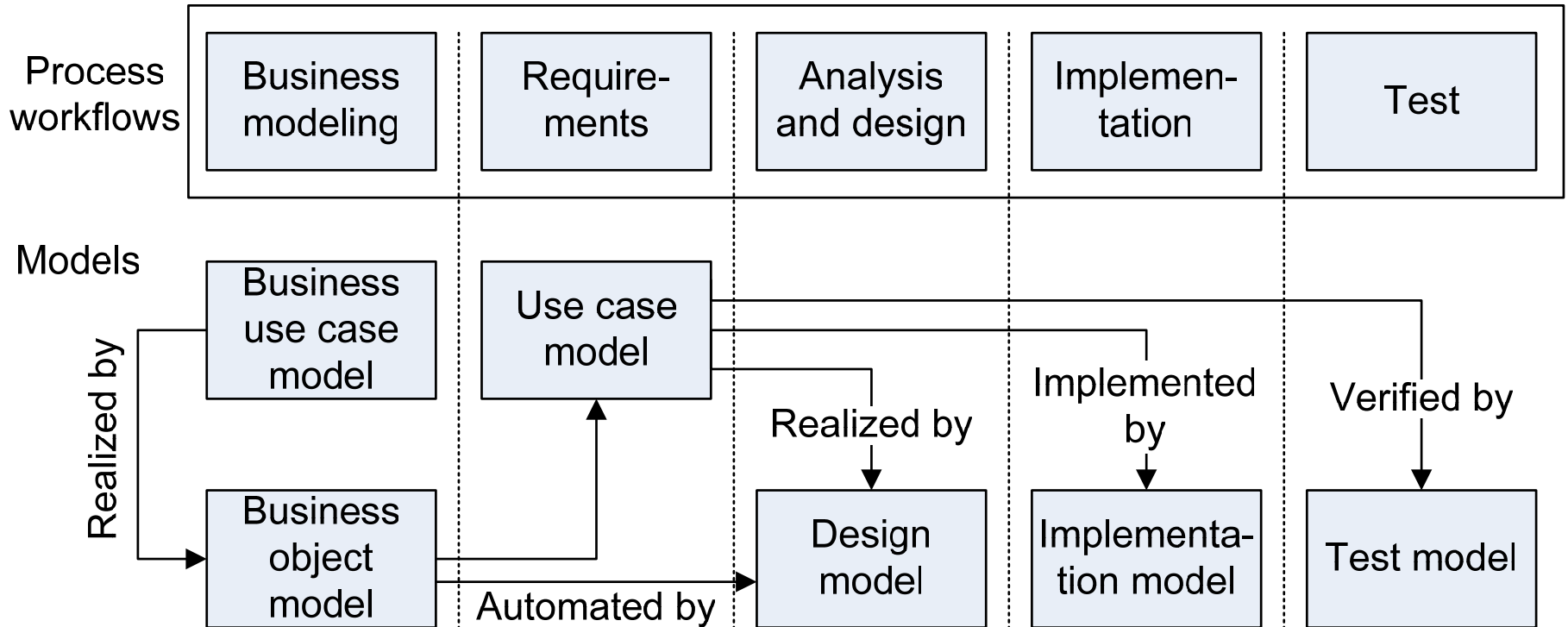
Stepwise refinement



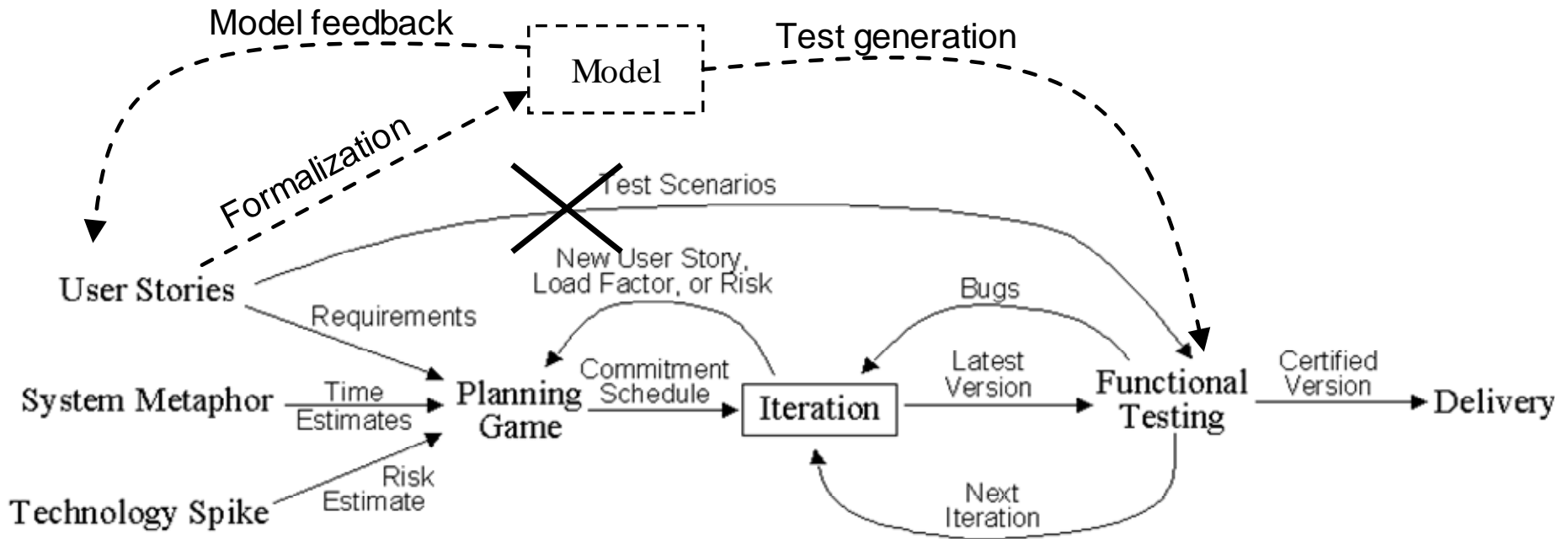
- **Test cases are derived from a model of the system**
 - The automation of the design of black-box tests
 - Instead of writing tests based on requirements documentation, the model are used for automatic generation of tests
- **The test cases are a model**
 - Abstract tests
 - Same abstraction level and semantics as the system model
- **Not necessarily “heavy weight”**
 - Can be used in scenario based development when the model is scenario centric



Model-based testing in RUP



Model-based testing in agile methods





Model-based testing one step further

- Tests are traditionally executed against implementations
- With executable models, models can act as implementations
- Models can be tested throughout the development
- Methods and techniques from testing theory and pragmatics can be applied to ensure quality of models
- Facilitates *refinement testing*



Escalator tool

- **Tool supporting model-based analysis**
 - Refinement verification
 - Model-based testing/Refinement testing
- **Input models and test models are sequence diagrams**
- **Formal foundation**
 - Refinement theory
 - Testing theory
- **Evaluated in medium sized case study**
- **Implemented as Eclipse plugin**
- **Prototype, but under development**

<http://heim.ifi.uio.no/~massl/escalator/escalatordemo.html>



Summary

- **Model-based analysis is a necessary part of Model Driven Development (MDD)**
 - Ensure quality of models
 - Validate refinement steps

- **Model-based analysis**
 - Should be based on both formal and pragmatic methods and techniques
 - Should be supported by automated tools

- **Escalator tool**
 - Refinement verification
 - Model-based testing/Refinement testing