

# PLEASE – Panel Discussion

Benjamin Hummel

Technische Universität München  
Munich, Germany

## Current Work: Index-Based Clone Detection

Create an index of a software system's code, that allows

- Efficient lookup of clones for single files
- Efficient update of the index after changes to the code

Applications:

- Large-scale clone detection
  - off-memory, distributed, etc.
- Faster response times
  - get clones within seconds after changing the system
- Incremental clone detection
  - Which clones were added/removed by a change to the system

## Current Work: Model Clone Detection

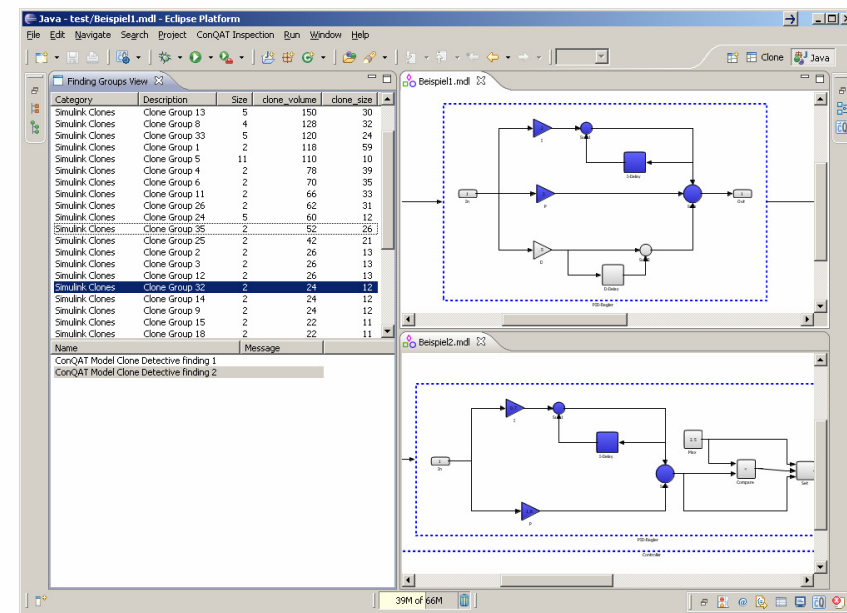
- Current clone detection research mostly based on source code
- In model-based development, the model is the primary artifact

### Results so far:

- Clone detection algorithm for data-flow models
- Application to MATLAB/Simulink models
- Metrics to sort clones by relevance

### Ongoing work:

- Stronger normalization for identifying type-4 clones
- Extension to state-based models



The screenshot displays the Eclipse IDE interface with the ConQAT Model Clone Detective plugin. The 'Finding Groups View' on the left shows a table of detected clones:

Category	Description	Size	clone_volume	clone_size
Simulink Clones	Clone Group 13	5	150	30
Simulink Clones	Clone Group 8	4	128	32
Simulink Clones	Clone Group 33	5	120	24
Simulink Clones	Clone Group 1	2	118	59
Simulink Clones	Clone Group 5	11	110	10
Simulink Clones	Clone Group 4	2	78	39
Simulink Clones	Clone Group 6	2	70	35
Simulink Clones	Clone Group 11	2	66	33
Simulink Clones	Clone Group 26	2	62	31
Simulink Clones	Clone Group 24	5	60	12
Simulink Clones	Clone Group 35	2	52	26
Simulink Clones	Clone Group 25	2	42	21
Simulink Clones	Clone Group 2	2	26	13
Simulink Clones	Clone Group 3	2	26	13
Simulink Clones	Clone Group 12	2	26	13
<b>Simulink Clones</b>	<b>Clone Group 32</b>	<b>2</b>	<b>24</b>	<b>12</b>
Simulink Clones	Clone Group 14	2	24	12
Simulink Clones	Clone Group 9	2	24	12
Simulink Clones	Clone Group 15	2	22	11
Simulink Clones	Clone Group 18	2	22	11

The main editor shows two Simulink block diagrams. The top diagram, 'Beispiel1.mdl', and the bottom diagram, 'Beispiel2.mdl', both feature a dashed blue box highlighting a specific sub-diagram, illustrating the detected clones. The bottom diagram shows a variation of the top one with additional blocks and connections.

## „Controversial Topics“

- Product lines defined/organized on model level or code level?
- Clone detection only useful for reverse engineering of existing product lines or also useful for forward engineering?
- Maybe difference detection more relevant than clone detection in PL context?